On Large Scale near-Independent Blind Source Separation

by

Dharmani Bhaveshkumar Choithram

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in

Information and Communication Technology

Dhirubhai Ambani Institute of Information and Communication Technology

January 2015

Declaration

This is to certify that

1. the thesis comprises my original work towards the degree of Doctor of Philosophy in Information and Communication Technology at DA-IICT and has not been submitted elsewhere for a degree,
2. due acknowledgment has been made in the text to all other material used.

Dharmani Bhaveshkumar Choithram

Certificate

This is to certify that the thesis work entitled *On Large Scale near-Independent Blind Source Separation* has been carried out by *Dharmani Bhaveshkumar Choithram* (200721001) for the degree of Doctor of Philosophy in Information and Communication Technology at this Institute under my supervision.

Prof. Dr. Suman Kumar Mitra

To Rev. DADAJI
(Rev. Pandurang Vaijanath Aathawale)

Acknowledgment

My complete PhD program has been a pilgrimage for me. A PhD program is like deciding to travel a path that is un-travelled and un-known, atleast less-travelled. It has been a demonstration for me, how GOD helps and directs if one is completely surrendered and ready to fight against every unknown hurdles. From where to begin, how to begin, whom to carry with as a supervisor or PhD guide, how to introduce our own creativity - all the related questions and confusions are hurdles. But, the understanding that GOD is near to me; HE loves me; not just HE is with me, HE works for me - changes the focus from the query or confusion or hurdle to the joy of being able to 'feel' the touch of HIS love. It has been my experience throughout that whenever, whatsoever, howsoever either guidance or caution or appreciation or indifference or any other type of psychological needs or even material needs are there for me; they have been always passed on to me by HIM. May be, when HE sends I have not understood at the same time. Thank you GOD, for constantly loving me, caring me! Not just the bit of creativity; definitely the creative idea is GOD-sent, as not a product of mechanical thought process; every utterance of thought or written word has been GOD-sent. But, then GOD-sent has to be the best. Off-course it has to be best; but not according to what HE can think or write but best according to what my limited qualities, abilities, capacities and hard-work can perceive. The PhD program has been definitely a voyage, because the overall experience has make me understand you better.

Starting form decision of joining for a PhD program, based on the philosophy that material success is not a hurdle but can be an instrument to serve GOD Rev. DADAJI, you have remained with me. GOD loves and loved; but DADAJI, you made me understand that. GOD knows better for me - what is good and what is not. HE gives me everything for my own development - whether it be circumstances or relatives or material needs. Whatever is given to me is a gift or *prasad* from HIM. With that I am supposed to just act on them without complain or condemn or disregard. DADAJI, this is the understanding passed on to me through your overflow of love for me. DADAJI, the *trikal-sandhya* given by you has been regular reminder of the presence of GOD within me and working for me. Thank you DADAJI! And, my gratefulness to you will not just find in words but also in deeds. That is my promise.

The understanding of philosophy has a power, but it is not the whole. It is very difficult to overcome the force of mechanical habits though understanding is there - has been my experience. So, continuously being with that philosophy needs continuous motivation, regular reminder. Rev. DIDIJI, you have been a living example for us to display that philosophy is to live and just to utter. DIDIJI, your simple slogans; such as, 'change of work is a refreshment', 'as much is possible, as much I will do and not I shall do, as much is possible' (exactly in Gujarati - '*Jetlu kariye etlu thay, nahi ke thay etlu kariye*') and others are not just has worked for empowerment of spiritual journey but also for the empowerment of our material life and success. Thank yoy DIDIJI, for being a constant source of inspiration by setting an example for me! Also, thanks to the divine family of DADAJI; who has been continuously with me withstanding my limitations. Specifically, I thank Dr. Jitubhai for his unbreakable faith in my abilities. Also, thank to Hardipbhai, Apurvbhai, Ashwinbhai, Amitbhai and many others for being with me, sharing my emotions.

Though GOD sends; HE sends through those around me, who love me, care me, keep concern for me, capable of and willing to pass on. Thanks to all those who have been instrumental in sending GODs gifts to me. The overall environment at DAIICT has been always encouraging and motivating. The focus on self learning, depth of work than superficial work, scholastic attitude and regular discussions with the faculties and co-mates have imparted a lot; without having formal teaching. Specifically, the discussions with Prof. V. P. Sinha and Sumanda have instilled lot in me. The discussions were ranging from technical to philosophy; from politics to day-by-day simple experiences. On one side, I was offered hard-core concepts of maths and the research area by Pro. S.C. Sahasrabudhe, Prof. Samaresh Chaterajee, Prof. Manish Gupta, Prof. Vijay Chakka and others. On the other side, there were professors like Prof. V.P. Sinha, Prof. Sumanda; who offered an assurance and confidence that research or concepts are not that difficult and I can understand, achieve it. I thank Prof. Hemant Patil for giving a IEEE special issue on Speech Processing to me. That has been first interaction to the research articles. The collection of quality research articles have serve to put before me an ideal research work. I thank Prof. Manish Gupta for passing on a love for non-triviality and being always friendly to me. I thank Prof. M. V. Joshi for being an example of hardwork and perfection for me. I thank my supervisor and thesis guide Prof. Sumanda. Relationship with others at DAIICT may be an accident, but relationship with Sumanda has been definitely GOD-sent. I was always sure of his love, concern and good wishes for me. Many times, I have been too much theoretical and in heavenly fantasy. I understand that the role given to Sumanda was to make me understand reality and bring down me on earth. And, through his pragmatic wisdom he definitely did it for me. Our discussions were full of extending my understanding of the world. Beyond all these, he has been a very humble person to me, allowing me great amount of freedom. He helped me, guided me such that my own originality is not ruined but enhanced by his presence. Being closer to me, he had to withstand all my limitations and he did that without reducing a bit of love for me. Thank you 'Da! A special thank to the members of synopsis committee. From that, Prof. Sanjay Srivastav for his discussion that has made me realize the limitations of numbers and I was inspired for further results through mathematical proofs; instead just numbers. I thank Prof. Ashok Amin, for making me able to see my habitual mistakes in writing patterns. I thank my external jury members, Prof. Aurobindo Routray and Prof. W. Skarbek for their inspiring reviews and guidance that empowered the thesis.

I can not forget to thank members of my tea-club including many; some of them are, Pratik, Ratnik, Guneshwar, Aaditya, Jignesh Bhavasar, Ghanashyam, Purushotthaman, Jignesh Bhatt, Naveen, Aashish, Gitam, Purvi, Rupsa for sharing my happiness, anger and sorrows. I also thank Aaditya, Pratik and Krishnagopal for sharing their math expertise. I thank Jignesh Bhatt, Guneshwar and Milind for sharing their technical expertise.

I thank the help-desk members for always being ready to offer their support. I thank the persons in administration, library, house-keeping, security for their co-operation. Thanks to all my neighbours, dears and nears for their co-operation. Thanks also to those who have always co-operated but due to space limitations I can not name.

Finally, I have not forgotten my family members - my parents, wife, brother, sister, son, nephew, niece and the other members. But, they were not different from me. I always felt them to be sailing through a same boat with me. GOD was one step ahead of me, and they were all together with me. So, just to say that in acknowledgment process also they are with me, and all previous feeling of gratefulness and saying thanks in response is conveyed by all of us together!

Abstract

The thesis addresses Blind Source Separation (BSS) in Large Scale (LS) and near-Independent (nI) sources scenario. The large scale in BSS imply number of unknown sources ranging from 15 to 140, so that the corresponding number of unknowns to be optimized range from 100 to 10000. The real world sources producing either an added spurious local optima or a shift of global optima or both are defined to be near-independent with respect to the used BSS contrast as an optimization criteria. The exponentially increasing solution space with linearly increasing dimensions for optimization and added complexity in the optimization landscape due to the near-independent sources make the Large Scale near-Independent BSS (LSnIBSS) to be a more difficult problem than the BSS. As a solution to the LSnIBSS problem, the thesis derives suitable optimization criteria and a Large Scale Global Optimization (LSGO) technique. Looking Probability Density Function (PDF) as a generalized multivariate differentiable function, there is derived L^2-Norm of Gradient of Function Difference (GFD) as a BSS contrast, where, GFD is the difference between gradient of product of marginal PDFs and gradient of joint PDF. A nonparametric estimation of the derived contrast is achieved through 'least squares' based kernel method in a single stage directly, instead of a two stages indirect estimation method. The contrast estimation is a particular demonstration of a derived more general method for information field analysis through a newly introduced concept of Reference Information Potential (RIP). The performance of kernel methods depend upon the choice of kernel bandwidth parameter. There is derived *Extended Rule-of-thumb* (ExROT) for bandwidth parameter selection in Kernel Density Estimation (KDE). The method is based on Gram-Charlier A-Series expansion as an approximation to the unknown PDF, assuming it being near Gaussian. The ExROT is better, in terms of Integrated Mean Square Error (IMSE) criteria of performance, compare to the Silverman's *Rule-of-thumb* (ROT) for unimodal density estimation with marginal increase in computational cost. The ExROT derived for multivariate density estimation and multivariate gradient of density estimation are applied to the derived BSS contrast. To accommodate near-independent scenario, there is introduced a Search for Rotation based Independent Component Analysis (SRICA) algorithm using, Genetic Algorithm (GA) like, search based global optimization technique. The BSS contrasts in simultaneous mode are proved to be nonseparable optimization functions (functions those can not be optimized componentwise), a difficult class of functions for LSGO. Towards success of GA, the schema concept is further generalized to dependency relation based *Extended Forma* from an existing generalization of equivalence relation based *Forma*. The generalization has an impact on the current debate on whether minimal (binary) alphabet or maximal (float) alphabet of representation for GA success. Taking inspiration from nature, the work in the thesis recommends use of either an intermediate level of alphabet or varying representation throughout the search. The former suggestion is empirically realized through Mendelian GA (MGA) based on the operators exploiting *Extended Forma*. The latter suggestion is empirically realized through newly defined the Gradual search scheme, the Spiral search scheme and others. The concepts are combined together to achieve a GA variant for LSGO of nonseparable functions. The solution is tested on the LSGO test bench functions and applied to the LSnIBSS problem using various contrasts.

Table of Contents

List of Figures

Chapter 1

Introduction

Blind Source Separation (BSS) is an area of research started in 1980's, popularized in 1990's and still expanding. It aims retrieval of unobserved sources from the observed mixtures in the absence of any prior information about the sources or the mixing system. Originated from a neuro-biological problem, now it finds applications in the areas of feature extraction, classification, telecommunications, brain signal processing, audio and speech processing, denoising and so on. The research has been nurtured and advanced by the research communities mainly working in the areas of neural net, signal processing and statistics. The detailed history can be found in (35, 65, 76).

1.1 The BSS Problem

The famous *Cocktail party problem* is a good example to explain the Source Separation problem. Let there be going on a cocktail party with multiple speakers, background music, breaking of a glass, continuous murmuring of a mass etc. as audio sources. Though there are multiple audio sources; for a human being present in that party, it is possible to focus on a particular audio source of choice. Whether a machine could have similar ability to separate the sources of choice from their mixtures? Elaborating more, let the recording devices are placed at some locations. The audio sources combine together to generate a mixture signal at each recording device. Let there be an assumption that the audio sources are linearly and instantaneously combined; where, the coefficients of linearity depend on their mutual geometrical locations, distances, characteristics of the environment and others. Mathematically,

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) + \ldots + a_{1n}s_n(t) \\ &\vdots \\ x_m(t) &= a_{m1}s_1(t) + a_{m2}s_2(t) + \ldots + a_{mn}s_n(t) \end{aligned}$$

where, m is the number of mixture signals; n is the number of sources; t is the time indice; $x_i(t), i = 1 : m$ are the mixture signals; $s_i(t), i = 1 : m$ are the source signals and $a_{ij}, i = 1 : m, j = 1 : n$ are the mixing coefficients. Overall, the generation mechanism of the mixture signals can be explained, using matrix notations, as under:

$$\mathbf{x}(t)_{m\times 1} = \mathbf{A}_{m\times n}\mathbf{s}(t)_{n\times 1} \tag{1.1}$$

where, $\mathbf{x}(t) = (x_1(t), x_2(t), ..., x_m(t))^T \in \mathbb{R}^m$ is an observed mixture random vector;
$\mathbf{s}(t) = (s_1(t), s_2(t), ..., s_n(t))^T \in \mathbb{R}^n$ is a source random vector and $\mathbf{A}$ is a mixing transformation. The *cocktail party problem* aims to separate and obtain back the actual sources ($\mathbf{s}(t)$) from the available mixtures ($\mathbf{x}(t)$). There are many possible approaches to solve this source separation problem. But, the 'blind' assumption implies there is no information available about the audio sources or recording devices; e.g. their mutual location geometry or source distributions or other.

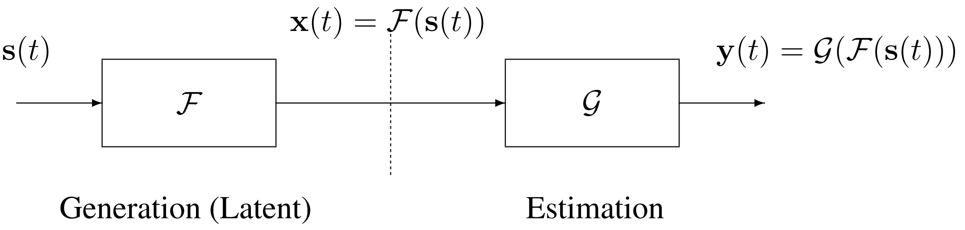
$\mathbf{s}(t)$
$\mathbf{x}(t) = \mathcal{F}(\mathbf{s}(t))$
$\mathbf{y}(t) = \mathcal{G}(\mathcal{F}(\mathbf{s}(t)))$
$\mathcal{F}$
$\mathcal{G}$
Generation (Latent)
Estimation

Figure 1.1: The BSS Model and the BSS Problem

Formally, the *BSS model* explains generation of an observed random vector $\mathbf{x}(t)$ as an unknown transformation $\mathcal{F}$ to an unobserved source vector $\mathbf{s}(t)$. Mathematically,

$$\mathbf{x}(t) = \mathcal{F}(\mathbf{s}(t)) \tag{1.2}$$

where, $\mathcal{F}$ is an m-component invertible mixing transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Formally, the *BSS problem* is to estimate the unknown $\mathbf{s}(t)$ based on some *generic assumption*[*] on the sources. The word *blind* implies that there is no other information available about the

[*] more generalized, in contrast to application *specific assumptions* in semi-BSS problem.

sources or the miximng system, except the applicability of the generic assumption. If **y**(t) is the estimated source vector and $\mathcal{G}$ is the estimated inverse of the mixing function $\mathcal{F}$ then

$$\mathbf{y}(t) = \mathcal{G}(\mathbf{x}(t)) = \mathcal{G}(\mathcal{F}(\mathbf{s}(t))) = \mathcal{H}(\mathbf{s}(t)) \tag{1.3}$$

where, $\mathcal{H}$ is an n-component transformation from $\mathbb{R}^n$ to $\mathbb{R}^n$ and each component h_i, $i = 1, 2, \ldots, n$ is a left indeterminacy transformation giving one-one mapping between the i^{th} estimated source and the i^{th} actual source; i.e. $y_i(t) = h_i(s_i(t))$.

The mixing transformation ($\mathcal{F}$) can be linear or non-linear, instantaneous (without memory) or dynamic (with memory), stationary or time-varying. Similarly, the transformation with condition $m > n$ (i.e., number of mixtures more than the number of sources) implies an *overdetermined* system; with condition $m < n$ (i.e., number of mixtures less than the number of sources) implies an *underdetermined* system; and that with condition $m = n$ (i.e., number of mixtures are same as the number of sources) implies a *determined* system. The thesis, if not specified, assumes linear, instantaneous and *determined* system.

1.1.1 Linear Instantaneous BSS Problem

Assuming the mixing transformation $\mathcal{F}$ to be linear, instantaneous and invertible; similar to those in *cocktail party problem* example; and extending the model for N sample instances, the BSS generation mechanism is explained as under:

$$\mathbf{X}(t)_{n\times N} = \mathbf{A}_{n\times n}\mathbf{S}(t)_{n\times N} \tag{1.4}$$

Accordingly, the linear instantaneous *BSS problem* is to estimate the source random vector $\mathbf{S}(t)_{n\times N}$ back from the available mixture random vector $\mathbf{X}(t)_{n\times N}$ without using any information about $\mathbf{A}$ or $\mathbf{S}(t)$. Mathematically, it is represented as in Equation (1.5) below:

$$\mathbf{y}(t)_{n\times N} = \mathbf{W}_{n\times n}\mathbf{X}(t)_{n\times N} = \mathbf{WAS}(t) = \mathbf{H}_{n\times n}\mathbf{S}(t)_{n\times N} \tag{1.5}$$

where, $\mathbf{Y}(t) = \hat{\mathbf{S}}(t)$ denotes estimated source random vector, $\mathbf{W} = \hat{\mathbf{A}}^{-1}$ denotes estimated inverse of the mixing matrix and $\mathbf{H}$ is the left indeterminacy transformation.

1.2 The BSS Solution

Towards the BSS solution, there are mainly four different approaches based on four different generic assumptions. The assumptions are:

1. Second order uncorrelatedness among the sources
2. Statistically independence and identical distribution among the sources, which leads to Independent Component Analysis (ICA)
3. Sparsity of the sources, which leads to Sparse Component Analysis (SCA)
4. Nonnegativity of the sources, which leads to Non-negative Matrix Factorization (NMF)

Instead of using mere uncorrelatedness depending upon second order statistics, statistical independence considering higher order statistics among the sources is a more stronger property. Sparsity and nonnegativity are relatively new assumptions valid in specific applications. In general, BSS using the generic assumption of independence among the sources has a wider perspective to cover more applications and the topic under the scope of this thesis.

1.2.1 Can Independence Assumption Solve the Linear, Instantaneous BSS Problem?

The fact that independence assumption can lead to BSS solution is proved by Darmois-Skitovich Theorem. The Theorem is independently proved by both Darmois (38) and Skitovitch (117), in the context of factor analysis.

Theorem 1.1 (The Darmois-Skitovich Theorem)**.** *Let y_1 and y_2 be random variables defined as under:*

$$
\begin{aligned}
y_1 &= g_{11}s_1 + g_{12}s_2 + \ldots + g_{1n}s_n \\
y_2 &= g_{21}s_1 + g_{22}s_2 + \ldots + g_{2n}s_n
\end{aligned}
$$

where, s_1, s_2, …, s_n are independent random variables. Then, if y_1 and y_2 are independent, all variables s_k for which $g_{1k}g_{2k} \neq 0$ are Gaussian.

The theorem states that non-Gaussian independent random variables can not be mixed linearly and instantaneously to have independent mixture outcomes. Consequently, with linear and instantaneous (memoryless) transformations both $\mathcal{G}$ and $\mathcal{F}$ in the above equation (1.3), if $\mathrm{s}(t)$ and $\mathbf{y}(t)$ both are independent then $\mathcal{G}(\mathcal{F}(\cdot))$ must correspond to a product of scale and permutation transformations. This proves that independence assumption can lead to separation of sources if not more than one source is Gaussian.

1.2.2 Linear, Instantaneous ICA for BSS

The equation (1.4) reminds us factorization of a data matrix in Component Analysis (CA). The goal of CA is to remove redundancy in the data matrix by change of basis. The conventional Principal Component Analysis (PCA) achieves this goal of redundancy removal by finding directions with maximum variance and mutually uncorrelated. As based on second order statistics, PCA is used to separate Gaussian or wide sense stationary (WSS) random processes. The general framework to solve BSS, based on the assumption of statistical independence among the sources, is inspired by PCA and has been identified as Independent Component Analysis (ICA). It achieves redundancy removal by finding the components, which are statistically the most independent; in a sense that the information in a component direction can not be known by knowing the other components. Other than a BSS technique, as a CA tool, ICA has been reported for many applications as dimensionality reduction, pattern classification, pattern recognition, feature extraction, data compression and others (42). The formal definition of ICA, given by P. Comon in (33) for linear transformations, is as under:

Definition 1.2. The ICA of a random vector $\mathbf{x} = (x_1, x_2, ..., x_m)^T$ with finite covariance $\mathbf{C_X}$ is a pair $\{\mathbf{B}, \mathbf{\Delta}^2\}$ of matrices such that

i. the covariance factorizes into $\mathrm{C}_\mathbf{x} = \mathbf{B\Delta^2B}^*$, where Δ is diagonal real positive and $\mathbf{B}$ is full column rank matrix, * indicates complex conjugate;

ii. the observations can be written as $\mathbf{x} = \mathbf{By}$, where $\mathbf{y}$ is an $n \times 1$ random vector with covariance $\mathbf{\Delta}^2$ and whose components are 'the most independent possible', in the sense of the maximization of a given 'contrast function'[†], as defined in Chapter 2.

The first condition, similar to the PCA definition, identifies $\mathbf{B}$ as a set of eigenvectors. In case of PCA, the second condition on $\mathbf{B}$ is orthogonality. For ICA, the second condition restricts $\mathbf{B}$ to give random variables y_i as independent components (ICs).

As can be noted, the ICA definition assumes $\mathbf{x}$ and $\mathbf{y}$ as random vectors without any time indices. Also, the instantaneous model, if extended for N number of samples, assumes identical distribution of the sources. Over all, ICA model assumes 'independent and identical distribution' (*i.i.d.*) assumption on the sources. Given the assumption is satisfied by a source random process, ICA can be used for BSS. Conventionally, linear ICA is considered equivalent to BSS. More details on the ICA solution and discussion on its' use for BSS is provided in Chapter 4.

[†] Roughly speaking, contrast function is a maximization function based on independence measure, satisfying specific conditions to bring the ICs uniquely.

As a conclusion, the BSS solution for linear, instantaneous mixing system can be obtained by maximizing the independence among $y_i(t)$s with respect to the separation matrix $\mathbf{W}$, as:

$$\mathbf{y}^*(t) = \underset{\mathbf{W}}{\text{argmax}}\ \Phi(\mathbf{y}(t)) \tag{1.6}$$

where, $\Phi(\mathbf{y}(t))$ is the contrast function. The solution demands discussion on the suitable contrast functions as optimization criteria and suitable optimization technique corresponding to that contrast function.

1.3 The Linear ICA Problem and Solution

The Independent Component Analysis (ICA) model explains generation of an observed random vector $\mathbf{x}$, as a linear transformation to another latent (hidden) random vector $\mathbf{s}$. Mathematically, $\mathbf{x} = \mathbf{A}\mathbf{s}$, where $\mathbf{x} = [x_1; x_2; \ldots; x_m]$; $\mathbf{s} = [s_1; s_2; \ldots; s_n]$; x_i, s_i are random variables with values in $\mathcal{R}$; $m = n >= 2$ and $\mathbf{A}$ is full rank. Let there be available N umber of samples of each observed random variable. Assuming an identical distribution, the instantaneous model can be extended for N realizations. Let $\mathbf{X} = [\mathbf{x}_1; \mathbf{x}_2; \ldots; \mathbf{x}_m]$ be the $m \times N$ data or observation matrix and $\mathbf{S}$ be the $n \times N$ component or source matrix. Then,

$$\mathbf{X} = \mathbf{A}\mathbf{S} \tag{1.7}$$

The problem of ICA is to estimate both the unknowns $\mathbf{A}$ and $\mathbf{S}$, with the only assumption of $\mathbf{s}_i$ being mutually the *most independent possible (m.i.p.)* random variables with respect to a given contrast. If $\mathbf{W}$ is the estimated inverse of the mixing matrix $\mathbf{A}$ then the estimated source or component matrix $\mathbf{Y}$ is:

$$\mathbf{Y} = \mathbf{A}^{-1}\mathbf{X} = \mathbf{W}\mathbf{X} = \mathbf{W}\mathbf{A}\mathbf{S} \tag{1.8}$$

The above ICA solution has the following inherent limitations or indeterminacy as discussed in (25, 33, 46):

- As $\mathbf{X} = \mathbf{A}\mathbf{S}$, scaling to any source s_i can be canceled by dividing corresponding column a_i of $\mathbf{A}$. So, both being unknown, the scaling or the variances and the signs of the ICs can not be estimated. Similarly, if $\mathbf{P}$ is the permutation matrix, $\mathbf{X} = (\mathbf{A}\mathbf{P}^{-\mathbf{1}})(\mathbf{P}\mathbf{S})$ i.e. the order of the estimated components can change with the change in the estimated mixing matrix. In short, the estimated sources $\mathbf{Y}$ can be obtained as a scaled and permuted version of actual sources $\mathbf{S}$. To have a unique solution, ICA assumes all components to be equivariant or univariant (33, Section 1.5).

- Gaussian distribution is symmetric. It can not be modified by the mixing vector or any mixing vector produces the same distribution. Accordingly, if there are more than one components with Gaussian distribution, actual mixing vector can not be known from the mixture.

1.3.1 An Orthogonal Approach to ICA Solution

The goal is to obtain ICs ($\mathbf{y}_i$s) from the correlated mixtures ($\mathbf{x}_i$s) of them.

- By definition, statistical independence implies uncorrelatedness (the opposite is true only for Gaussian variable). The uncorrelated components with zero mean imply orthogonality. So, the ICs with zero mean are also mutually orthogonal.
- The zero mean uncorrelated components of the data matrix are mutually orthogonal and zero mean ICs of the data matrix are also mutually orthogonal. There are techniques available to get zero mean uncorrelated components of the data matrix. If we think of a transformation from the former to the latter, then it must be an orthogonal transformation. Because, only an orthogonal transformation can keep the orthogonal components, still orthogonal. To bring uniqueness of the estimated components, the uncorrelated components should be univariant or equivariant. Let a zero mean observed mixture data matrix **X**, be linearly transformed through a whitening matrix **V**, to give a zero mean, univariant, whiten (uncorrelated and equivariant) data matrix **Z**.

$$\begin{aligned} \mathbf{Z} &= \mathbf{VX} && (1.9) \\ \Rightarrow \mathbf{Z} &= \mathbf{VAS} && (1.10) \end{aligned}$$

The goal is to obtain the estimated sources $\mathbf{y}_i$ as zero mean, univariant, ICs. Let us assume **R** is the linear transformation for that. Then,

$$\begin{aligned} \mathbf{Y} &= \mathbf{RZ} && (1.11) \\ \Rightarrow E\{\mathbf{yy}^T\} &= E\{\mathbf{RZZ}^T\mathbf{R}^T\} \\ \Rightarrow \mathbf{I} &= E\{\mathbf{RIR}^T\} && (1.12) \end{aligned}$$

where, **I** is the identity matrix. It proves that **R** needs to be orthogonal. Accordingly,

$$\mathbf{Y} = \mathbf{RZ} = \mathbf{RVAS} = \mathbf{WAS} \quad (1.13)$$

where, $\mathbf{W} = \mathbf{RV}$ is the estimated unmixing matrix.

- Orthogonal matrices with determinant +1 are rotation matrices and those with determinant -1 are reflection matrices. Given a reflection orthogonal matrix with determinant -1, by negating an odd number of columns, a new rotation orthogonal matrix with determinant +1 can be derived. As the estimated sources are allowed to be scaled or reflected version of the actual sources, the orthogonal transformation matrix $\mathbf{R}$ can be a rotational matrix.

- Concluding above - a specific rotation matrix, from the infinite set of all n-dimensional rotation matrices, will be able to transform a set of zero mean whiten or eigen (not univariant but orthogonal) components to ICs. So, the ICA problem reduces to estimating a rotation matrix $\mathbf{R}$ giving *m.i.p.* $\mathbf{y}_i$s.

$$\mathbf{Y}^* = \underset{\mathbf{W}*}{\text{argmax}}\ \Phi(\mathbf{WZ}) \tag{1.14}$$

  where, $\Phi(\mathbf{y})$ or $\Phi(\mathbf{Y})$ is the contrast function based on the dependnece or independence measure of random vector $\mathbf{Y}$.

- The $n \times n$ rotation matrix $\mathbf{R}$ has $d = \frac{n(n-1)}{2}$ entries to be estimated. Other way, $d = \frac{n(n-1)}{2}$ number of 2-d rotations are required to have $\mathbf{R}_{n\times n}$. Overall, the linear ICA or linear BSS problem to separate n number of sources reduces to d-dimensional optimization problem.

1.4 The Large Scale BSS (LSBSS) Problem

It is known and also proved in the Section 1.3.1 that the linear BSS problem with n number of unknown sources is an $d = \frac{n(n-1)}{2}$ dimensional optimization problem. Accordingly, with number of sources $n > 14$ in BSS, the optimization problem has to deal with dimensions $d > 100$. Similarly, $n > 45$ corresponds to $d > 1000$ and $n > 141$ corresponds to $d > 10000$. The optimization research community refers a problem with dimensions $d \in [100, 10000)$ as the Large Scale Global Optimization (LSGO) problem and a problem with dimensions $d > 10000$ as the Big Scale Global Optimization problem. So, the thesis refers 'Large Scale' in LSBSS problem as the BSS in higher dimensions with number of sources ranging from more than 14 to less than 141, i.e. $n \in [15, 140] \Rightarrow d \in [100, 10000)]$. The optimization techniques already face the problem of 'curse of dimensionality'. With the linear increase in number of dimensions (d), the solution space increases exponentially (a^d, some a) and so does the difficulty in optimization. So, the LSGO for real world applications is still challenging and an identified problem (139). It has been a part of competitions at many conferences; such as, IEEE Congress on Evolutionary Computation (IEEE CEC) 2008, 2010, 2013, 2015. Overall, the solution of LSBSS demands multi-disciplinary approach.

1.5 The Large Scale near-Independent BSS (LSnIBSS) Problem

The literary meaning of near-independence is - 'not exact independence'. But, The ICA model allows the components, being separated, as mutually the 'most independent possible' (*m.i.p.*)[‡] with respect to a given contrast function. So, the thesis defines 'near-independent' sources as the sources not being *m.i.p.* with respect to the used contrast function.

The *m.i.p.* sources correspond to the global optima of the optimization landscape. The near-independence among the sources may be exhibited in the following three ways in the optimization landscape of the used contrast:

i. The actual sources correspond to the solution near global optima i.e. there is a shift of the global optima such that the optimal solution do not correspond to the actual sources.

ii. There exists an added one or more local optima, which do not correspond to the actual sources.

iii. There is simultaneously an added local optima, as well as, shifted global optima.

The added local optima makes the optimization landscape difficult to be optimized but the shifted global makes either almost impossible to find the optima corresponding to the actual sources without any additional information or only an approximate solution can be obtained based on the amount of shift. At lower dimensions, a slight shift in global optima may allow atleast an approximate solution. With increasing dimension, cumulative slight shifts in pairwise optima, may cause the actual solution much far than the global optimal. Overall, the LSBSS problem demands special focus on the study of the circumstances causing these adverse optimization landscape and their consequences on *separability*. The thesis identifies the study area as 'near-Independent' BSS (nIBSS). The sources producing either shift of global optima or addition of spurious local optima or both with respect to the used contrast qualify to be near-independent for that contrast. It is to be noted that the near-independence is not a characteristic of sources alone, but it is the characteristic of sources exhibited in the presence of a specific contrast.

Though the term near-independence is new, there already exists local minima and extrema/optima analysis of different contrast functions with respect to various types of sources. Recently, there has been found situations that affect the optimization landscape in case of BSS of real world sources. There exists spurious local optima of information theoretic independence measures for multimodal source distributions. The empirical observations are supported by theoretical extrema analysis in (17, 90, 91, 92, 132, 134). On the other hand, it has been proved that lack of

[‡] whether mention or not, *m.i.p.* implies most independent possible with respect to the given contrast function.

number of samples may bring overlearning phenomena in ICA for kurtosis like independence measures (83, 108, 109). The overlearning results into a shift of global optima. The near-independence terminology makes it possible to study two differently looking problems, under the same roof.

The near-independence condition is not same as the non-independent or non-*i.i.d.* conditions stated in the BSS literature. Usually, non-independent sources imply time dependencies and non-identicle conditions imply non-stationarity. There exists BSS model extensions for non-*i.i.d.* i.e. temporally dependent sources and/or non-stationary sources (41, 70, 73). More precisely, the non-*i.i.d.* is the property of the sources only, while the near independence implies the source model violation with respect to the used contrast function only. Also, non-independence is more stronger than the near-independence, in terms of violating of ICA source model assumption.

Overall, the LSBSS of the more difficult real world near-independent sources give birth to the Large Scale near-Independent BSS (LSnIBSS) problem. The performance degradation in LSnIBSS is either due to the failure of an optimization techniques in the presence of local minima] or due to the shift of global (i.e. an optimization technique is successful in finding the optimal but the optimality does not assure separability) or due to both the former reasons.

1.6 Current State of the Art

The linear BSS algorithms differ based on the used optimization criteria and optimization method.

Conventionally, independence interpretations in terms of minimization of mutual information, maximization of non-gaussianity and their approximations using higher order statistics (cumulants and moments) have served as the major guiding principles to derive the BSS contrasts. There exists ICA techniques using adaptive learning through neural net (11, 63, 64, 71, 77, 129). The nonlinearity used for learning has to be a function of probability density function (PDF) of the components to be estimated. In the absence of this knowledge, family of densities (e.g. super-Gaussian or sub-Gaussian) is used as an approximation to select the nonlinearity. This requires some prior knowledge of densities to be estimated and so violates the blind assumption. There exists algebraic techniques (22, 23, 27, 28, 33) trying to obtain uncorrelatedness of third or fourth order statistics, inspired by the diagonalization techniques for PCA through second order uncorrelatedness. With approximate independence measures, they offer a less precise solution (10). There are also likelihood (12, 24) based signal processing techniques for ICA.

Most of the ICA algorithms use gradient based optimization techniques though exhaustive search based optimization techniques have also been explored (75, 112). The gradient based optimization techniques lack global convergence, required specifically in near-independent BSS with local minima. The exhaustive search methods used for optimization are computationally demanding, specifically in large scale.

Overall, a BSS algorithm using contrast that allows blind estimation, offers precision in separation quality and computationally efficient is still in demand. Further, the large scale and near-independent scenario demands the same BSS algorithm using a global and computationally efficient optimization technique.

The kernel based nonparametric estimation methods (10, 18, 87) are both quite precise and blind but require high computational cost. In search of a BSS contrast with computation reduction, there has been explored alternative definitions of Entropy other than Shanon's. So, the latest trend is to develop an ICA algorithms using kernel estimation of an independence measure that is based on alternative definitions of Entropy, specifically, the quadratic measures of independence offering low computational cost. The Information Theoretic Learning (ITL) - a new research area (96) and article (1) provide many such alternative definitions of independence and quadratic independence measures; for example, generalized β-Class Entropy, Renyi's Entropy, Cross Information Potential (CIP), Euclidean distance (D_{ED}) based and Cauchy-Schwartz distance (D_{CS}) based Quadrature Mutual Information (QMI) (58, 112) and others. It should also be noted that Pham (93) proved that there are risks using Renyi's entropy definition for BSS.

Concluding above, in the midst of existing many other algorithms, there is still a requirement for a linear BSS algorithm that is - blind in nature, based on computationally efficient kernel based nonparametric estimation of contrast and using optimization technique with good global convergence - even in small or medium scale BSS.

The performance degradation of existing ICA algorithms with increase in dimensions is a known fact (10, 75). The large scale in BSS, using independence assumption, has been addressed only in article (18) as per the knowledge of the author. There it is claimed that the ICA technique (NPICA) based on the nonparametric estimation of marginal entropies can seamlessly handle large scale. But, the empirical results reported in this thesis in the Chapter 4.8, show failure of NPICA in two dimensions, as well, in higher dimensions for near independent sources. There are efforts to solve LSBSS problem using other than independence assumption by (30) and (19). In general, the LSnIBSS, through conventional independence assumption, is still an unsolved problem. The performance degradation in LSnIBSS is either in terms of the failure of an optimization techniques to converge to an optimal or in terms of highly increased computation. Atleast the brain signal processing area, for EEG (Electro-Encephalo-Graph) and MEG (Megneto-Encephalo-Graph) data analysis like applications, currently demands LSBSS. Conventionally, they are derived through sequentially executing Blind Signal Extraction (BSE) algorithm to extract one or few important signals, instead of BSS (37, 133). It is anticipated that the LSnIBSS solution will find applications in brain signal processing, feature extraction and other data analysis problems.

The LSnIBSS solution demands contrast providing optimization landscape without any spurious local optima and shift of global optima. It demands optimization algorithm that is computationally efficient and has good global convergence. The solution also demands study on near-independence scenario.

Overall, the requirements of LSnIBSS have been identified as the research problem for the thesis. This also justifies the title of the thesis.

1.7 The Motivation Summary and Work Directions

The thesis addresses the LSnIBSS problem in three directions:

1. Towards optimization criteria: As concluded in the previous Section 1.6, contrast that sticks to the blind assumption through kernel based nonparametric estimation, offers precision in separation quality, computationally efficient, without any local minima and using a 'prior' that does not violate the blind assumption is i demand. The work towards this direction is briefed in Chapter 2 and Chapter 3.

2. Towards optimization landscape: The near independent sources scenario needs to analyze situations affecting the optimization landscape, their consequences on separation quality and possible remedies. The related work is reported in Chapter 4.

3. Towards optimization technique: The Large Scale Global Optimization (LSGO) is reported to have linear time complexity ($O(d \ln d)$, where d is the dimension of search) for a specific type of problems and an exponential time complexity ($O(d^d)$ or $O(\exp(d \ln d))$) for an another type of problems (107). The BSS contrasts belong to which group of optimization functions for LSGO that need be identified first. Then, a suitable LSGO technique, either existing or newly defined, need be used for LSnIBSS. The work towards this direction, is reported in chapters 4, 5 and 6.

1.8 The Thesis Organization and Detailed Outline

The next Chapter 2 derives new contrasts for linear BSS. For differentiable multivariate functions with equal hyper volumes (region bounded by hyper surfaces) some results are proved relating equality of derivatives to equality of the functions. The results are applied to the independence definition stating equality of joint PDF and product of the marginal PDFs of a random vector. This avails new independence measures and BSS contrasts. The Chapter defines difference between the joint PDF and the product of the marginal PDFs as a Function Difference (FD) of a random vector. Similar to the Score Function Difference (SFD) definition in (7, 8), the gradient of FD (GFD) and the Hessian of FD (HFD) are defined. It is proved that FD, GFD, HFD all are zero everywhere when the corresponding random variables are independent. The results lead to derive minimization of L^p-Norm of FD, GFD and HFD as contrasts for BSS.

The estimation method should be computationally efficient to match the requirement specifically for LSBSS. The contrasts depending upon joint PDF and marginal PDFs both are usually computationally demanding though more accurate (88). Instead of a conventional two stage estimation approach for a quantity like FD, a direct single stage estimation is more accurate. This is concluded and applied for 'least squares' based density difference estimation in (123). Also, the kernel theory identifies the fact that it is computationally more efficient to estimate the integration of square of PDF than the estimation of actual PDF. The ITL theory has given significance to this fact by defining integration of the square of PDF as an Information Potential of a random variable. The analogy, with the existing potential theory in Physics, also has given other concepts related to the information field; like, information forces, information particle interactions and others (96, 140, 141). The Section 2.7 targets both the efficient estimation of the proposed contrasts and extension of the potential theory for an information field. The potential theory has a concept of reference potential and it is used to derive closed form expression for the relative analysis of potential field. Analogous to it, the Section 2.8 introduces the concepts of Reference Information Potential (RIP) and Cross Reference Information Potential (CRIP) based on the potential due to kernel function placed at selected sample points as basis in kernel methods. The quantities are used to derive closed form expressions for information field analysis using least squares. The expressions are derived through multiplicative kernel basis in two ways: (a) basis placed at the selected paired sample points (b) basis placed at the selected paired or un-paired sample points. The expressions are used to estimate the required contrast functions. They are used to estimate L^2-Norm of FD and L^2-Norm of GFD based contrasts.

The performance of a kernel method depends upon efficient bandwidth parameter selection. The most popular and simple Silverman's *Rule-of-Thumb* (ROT) (116) does not give precise bandwidth parameter and the precise *solve-the-equation based plug-in* methods are computationally too demanding. So, deriving data dependent bandwidth selection method for Kernel Density Estimation (KDE) that balances accuracy and computation is the focus of the next Chapter 3. It achieves this goal by deriving a novel *Extended rule-of-thumb* (ExROT). The ROT optimizes Asymptotic Mean Integrated Square Error (AMISE) with the assumption that the density being estimated is Gaussian. The ExROT uses infinite series expansion as an approximation to the unknown PDF. As an example here, the ExROT uses an extended assumption that the density being estimated is near Gaussian. The assumption makes it possible to use the Gram-Charlier (GC) A-series expansion of near Gaussian PDF with the same AMISE criteria for bandwidth optimization. There exist many other infinite series expansions of PDF based on which other variants of the ExROT could be derived. The multivariate ExROT is derived using the multivariate GC A-series. For that, the multivariate GC A-series is derived by generalizing a specific derivation in (13) for the univariate Generalized Gram-Charlier (GGC) series expansion to multivariate using Kronecker algebra. The ExROT is also derived for gradient of multivariate density estimate. The empirical results on the standard test set for univariate density show the superiority of ExROT over ROT in all unimodal density estimation cases - skewed or kurtotic or with outliers and some of the multimodal cases. Thus, ExROT is a better option to ROT with comparable cost. The Chapter ends with the application of ExROT for previously derived estimators of FD and GFD based contrasts.

The Chapter 4 is devoted to the near-Independent BSS. It provides both theoretical and empirical local minima analysis of selected BSS contrasts for various source distributions. Then, it derives ICA algorithm using, Genetic Algorithm (GA) like, search based global optimization technique to allow BSS of near-independent source. It verifies the newly derived L^2-Norm of FD and L^2-Norm of GFD as BSS contrasts. The contrasts are estimated using the RIP and CRIP based formal expressions. The bandwidth selection for estimation is provided through both ROT and ExROT. The SRICA facilitates comparison of separation quality due to various BSS contrasts against varying source distributions by allowing use of same optimization algorithm for all the contrasts. The experimental results show failure of most ICA algorithms, including SRICA, in BSS of two sources with specific distributions and in BSS of higher dimensions. All the experimental results put together, bring further understanding of the overlearning phenomena and a discussion on the equivalence of ICA and BSS even in linear case. The failure of SRICA also necessitates focus on the success of GA as an optimization technique in the Chapter 5 and misconvergence of GA in higher dimensions in the Chapter 6.

The success of GA is explained through a notion of Schema (a template of similarity among the search strings) and Schema Theorem. The next Chapter 5 extends this GA search theory (GA algebra). The notion of schema, has been further generalized to dependency relation based *Extended Forma* from the current generalization as an equivalence relation based *Forma*. There are derived some operators exploiting the *Extended Formae* (plural of *Forma*) based similarities. Over all, the generalization achieves theoretical maximum possible schemata (plural of schema) for both the string and non-string representation structures using maximal alphabet. This has an impact on the current discussion on whether small alphabet (bi-nary) or maximal alphabet (float) for GA representation. Taking inspiration from the nature, the work recommends use of either an intermediate level alphabet - balancing maximal alphabet to avail maximum schemata and minimal alphabet to overcome some of the disadvantages due to maximum schemata - or varying representations during various stages of search. The above representation and operators are empirically used to derive Mendelian Genetic Algorithm (MGA). MGA, with abundance of schema, is inferred to avoid this misconvergence at least partially.

The Chapter 6 collaborates LSGO and BSS. It starts with the state of the art large scale global optimization methods and the reason for possible misconvergence. The discussion also figures out that the BSS contrasts, in simultaneous mode, are non-separable functions, a difficult class of functions for LSGO. Towards the partial success to overcome misconvergence in GA and to reduce the computation for a non-separable global function optimization, there are discussed various search strategies with GA. The search strategies, for example, are - varying representations (gradual search), spiral search, delta search, refine search, population reinitialization and others. The concepts are mingled with existing Cooperative Coevolution search technique (95) and random grouping (143) for LSGO of standard test bench functions. The LSGO solutions are also applied to LSnIBSS.

Finally, the thesis ends with conclusion and possible future extensions in Chapter 7.

1.9 Contribution and Novelty

The contribution and novelty of the work can be briefed as under.

- For k^{th} order differentiable multivariate functions with equal hyper volumes (region bounded by hyper surfaces) and added condition of bounded support, it is proved that equality of k^{th} order derivatives implies equality of the corresponding functions.

- The L^p-norm of FD, GFD and HFD are derived as contrasts for BSS.

- The closed form expressions in terms of RIP and CRIP are derived for information field analysis using least squares and applied to estimate the derived L^2-norm of FD and GFD contrasts.

- The near-Gaussian PDF assumption and the Gram-Charlier A-Series based an *Extended Rule-of-thumb* is derived. It is experimentally proved to be better compare to ROT atleast in all sorts of (skewed or kurtotic or with outliers) unimodal density estimations. The ExROT is also derived for multivariate density estimation and gradient of univariate or multivariate density estimation.

- A specific derivation for the univariate GGC is extended to multivariate using kronecker algebra.

- The Search for Rotation based ICA (SRICA) algorithm using, GA like, global search based optimization method is derived.

- The near-Independent analysis brings some new conditions providing local minima. Also, it provides discussion on the use of ICA as BSS in linear case.

- The schema definition is further generalized to dependency relation based *Extended Forma* for both string and non-string structures of representation. The definition achieves theoretical maximum possible schemata with diploid representation (pair of chromosomes per individual) and corresponding operators. The new insights also leads to a discussion on whether maximal alphabet or minimal alphabet for representation.

- The novel Mendelian Genetic Algorithm using *Extended Forma* based representation and operators is derived.

- A varying representation, while search in progress, is achieved using gradual search and spiral search concepts. The concepts are empirically proved to be better than conventional search of a group of variables in nonseparable function global optimization. The concepts are mingled with existing Cooperative Coevolution search technique (95) and random grouping (143) for LSGO of standard test bench functions and a LSnIBSS application.

Chapter 2

Contrast Functions

The chapter defines contrast function for BSS, discusses the state of the art and derives some new contrast functions. The new contrasts are based on the L^p distance between the joint PDF and product of the marginal PDFs of a random vector; its gradient and Hessian. A direct estimate of L^2 distance based contrasts using least squares with Gaussian kernel basis is derived. The contrasts with their estimation methods are compared with existing other contrasts. Finally, the contrasts are tested as independence measures. The computational load for parameter selection bring motivation to derive an accurate and computationally efficient data dependent kernel bandwidth parameter selection method, which has been addressed in the next chapter.

2.1 Introduction

Contrast functions or simply contrasts[*] are the optimization functions to assure blind separation of unobserved sources from the available observation mixtures, when maximized. The independence definition, its various interpretations and their approximations are used to derive contrasts.

The initial phase of research on BSS contrasts focused on the Shanon entropy and Kullback-Leibler divergence (KLD) based information theoretic independence interpretations and their approximations through higher order statistics (36, 84). The other significant group of contrasts came from non-Gaussianity interpretations of independence and their approximations (65). More details on these widely used, conventional contrast functions can be found in (26, 62, 94).

The research towards new contrasts for BSS has the following motivations.

1. More accurate BSS solution seems an everlasting hunger. So, just out of mathematical vigor to search for a more accurate solutions, new contrasts are always of interest.

[*] The formal definition is in Section 2.6.

2. The Shanon entropy based contrasts are found to have spurious local optima (18, 90, 132). Therefore, the contrast functions without the existence of spurious local optima are desired.

3. The large scale in BSS requires balancing accuracy with computation. This has motivated direct and fast estimation methods to derive contrasts (88, 89, 124).

4. Some BSS contrasts with their estimation methods are biased towards a parametric family, say, subGaussian or superGaussian To achieve unbiased estimation of sources, the focus has shifted to BSS using kernel based non-parametric estimation of various independence measures, as in Nonparametric ICA (NPICA) (18) and kernel ICA (kICA) (10).

5. The use of 'prior' information with the independence assumption may find better estimations of the actual sources. Therefore, the contrast functions incorporating more generalized priors without violating the blind assumptions, other than the application specific priors used in Bayesian approach for BSS and semi-BSS problems, are of interest. The bounded support assumption is one of such assumptions, used by many geometry based ICA and BSS algorithms (126, 133).

Overall, the contrasts giving more accuracy at low computation, blind and without local minima are still in demand and open for further research.

To overcome this demand, the latest trend in BSS contrasts follows two directions.

1. Other than the conventional Shanon entropy and KLD as a divergence measure between two PDFs, there exists many alternative definitions and interpretations of entropy, PDF distance measures and independence interpretations (96, 112, 128). Inspired by the above motivations, the research community has started focusing on these alternatives to derive new BSS contrasts (10, 75).

2. The new independence interpretation should be incorporated with kernel based fast and non-parametric estimation technique to derive new BSS contrast.

Combining both the above directions, the latest trend is to use quadratic measures of independence for BSS. The article by Achard et al. (1) uses L^2 distance between the transformed characteristic functions of joint and product of the marginal PDFs. The Information Theoretic Learning (ITL) suggests many such quadratic independence measures, for example, Renyi's Entropy, Cross Information Potential (CIP), Euclidean distance (D_{ED}) based and Cauchy-Schwartz distance (D_{CS}) based Quadrature Mutual Information (QMI) (58, 96). The article by Seth et al. (112) provides ITL based unified framework to those quadratic distance measures and proposes a new parameter free distance measure for ICA.

The current chapter is inspired by all the above motivations and follows the latest trend. It derives some new independence interpretations relating gradient of the PDFs, specifically for bounded support random variables, and proposes new BSS contrasts. It achieves their nonparametric estimation with reduced computation by using least squares based direct estimation approach. The suitable choice of a kernel bandwidth parameter using data dependent bandwidth selection *Extended Rule-of-Thumb* by (**?**) achieves a parameter free contrast estimation.

There have been proved some results for generalized differentiable multivariate functions. Looking PDFs as a generalized functions, the results are applied on independence of random vectors. The results are: 1) The equality of the gradient of joint probability density function (PDF) and the gradient of product of the marginal PDFs imply independence. 2) The equality of the Hessian of joint PDF and the Hessian of product of the marginal PDFs imply independence, if the prior given that the random vector has bounded support i.e. its probability outside certain region is zero. These new independence interpretations are used to derive new independence measures and contrast functions for BSS. The bounded support condition is not very restricting. The reasons are: 1) Empirically, the sampled region is always bounded. 2) Numerically, the computers always work with definite range. Though may not be always, it might be a valid approach in most cases to blindly consider PDF outside the bounded sampled region to be zero. To achieve nonparametric estimation of the newly derived contrasts, there has been derived single stage direct estimation method using least squares. To take the advantage of the quadratic nature of the contrasts, there are defined concepts of Reference Information Potential (RIP) and Cross RIP (CRIP) that depend upon IP due to selected kernel basis. The concepts are used to achieve closed form expressions for information field analysis. The derived closed form expression are verified by applying them to obtain L^2-Norm of FD and L^2-Norm of GFD contrasts. The method uses Gaussian kernels as basis and has two variations. One, the basis are placed at the selected paired sample points only. Another, the basis are placed at selected sample points may be paired or unpaired.

The next Section 2.2 derives some results for generalized multivariate differentiable functions with bounded support. The results are applied to statistical independence condition in Section 2.3. To better exploit the results, it derives new definitions and their important properties. Corresponding to that, the new independence measures are derived in Section 2.4. The Section 2.5 briefs the BSS problem and the possible approach for solution. The previous results are used to derive new BSS contrasts; satisfying the important properties of Scale invariance, Dominance and Discrimination; in Section 2.6.. There is also done local minima analysis of the derived contrast. The next Section 2.7 discusses the contrast function estimation approaches and derives prerequisites of Kernel Theory and Information Potential (IP). The Section 2.8 defines the Reference IP (RIP) and related concepts. Then, the Section 2.8.4 derives the least squares based closed form expression for information field analysis. The expressions are used to derive FD based estimators LSFD and

LSFD2 in Section 2.9.1 and GFD based estimators LSGFD and LSGFD2 in Section 2.10. The Section 2.11 reports empirical verification of the derived independence measures and BSS contrasts. The Section 2.11.1 provides important discussion on required parameter selection for the derived estimators. Finally, the chapter ends with conclusion in Section 2.12.

2.2 Some Results On the Equality of Generalized Constrained Multivariate Functions

Definition 2.1. A function $f : \mathbb{R}^n \to f(\mathbb{R}^n)$ is said to have support $\mathcal{R}$ if $f(\mathbf{x}) = 0, \forall \mathbf{x} \in \mathcal{R}'$, where, $\mathcal{R} \subseteq \mathbb{R}^n$ and $\mathcal{R}'$ is its complement set. It is represented as $supp(f) = \mathcal{R}$. Any superset of $\mathcal{R}$ is also a support. If $\mathcal{R}$ is bounded above and bounded below then f is a said to be a bounded support function.

Let Conv$(\mathcal{R})$ be the convex hull of $\mathcal{R}$ that contains all convex combinations of points in $\mathcal{R}$. Then, the definition says that for the bounded support functions both the support $\mathcal{R}$ and its convex hull Conv$(\mathcal{R})$ have finite measures. If $\mathcal{R}$ is convex, both the support measure ($l(\mathcal{R})$) and its range ($l($Conv$(\mathcal{R}))$) are same, where l is the length of an interval. For example: let $\mathcal{R} = [-1, 1]$. Then, the support measure $l(\mathcal{R})$ and the range $l($Conv$(\mathcal{R})) = 2$. Now, let $\mathcal{R} = [-1, 1] \bigcup (2, 4] \setminus 3$. Then, $l(\mathcal{R})$ is 4. But, the Conv$(\mathcal{R})$ is $[-1, 4]$ and $l($Conv$(\mathcal{R})) = 5$.

For differentiable multivariate functions with equal hyper volumes (region bounded by hyper surfaces) the following results are derived. For some of the results, an added constraint of random vector having bounded support is required.

Theorem 2.2. *Let $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}$ and both satisfy the following conditions:*

1. *They have bounded support.*
2. *They are differentiable.*
3. $\int_{\mathbb{R}^n} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} g(\mathbf{x}) d\mathbf{x},\ \mathbf{x} \in \mathbb{R}^n$

Then, the following holds:

$$\nabla f(\mathbf{x}) = \nabla g(\mathbf{x}) \Rightarrow f(\mathbf{x}) = g(\mathbf{x}) \tag{2.1}$$

Proof. Let us prove this Theorem by mathematical induction.
The Base Case: $n = 1$
Without loss of generality, let $\mathbf{I} = [-a, a] \supseteq$ Conv$(\mathcal{R}), a \in \mathbb{R}$

Given $\int_{\mathbb{R}^n} f(x)dx = \int_{\mathbb{R}^n} g(x)dx$ and $\frac{d}{dx}f(x) = \frac{d}{dx}g(x)$.
Integrating both the sides of the latter equation leads to,

$$f(x) \quad = \quad g(x) + c \tag{2.2}$$

where, c is some arbitrary constant.
Integrating both the sides of Equation (2.2) with respect to x from $-a$ to a, brings:

$$\int_{-a}^{a} cdx = 0 \Rightarrow c = 0$$

This proves the Theorem for the base case.
The induction step: Given the Lemma holds for $n = k$, let us prove it for $n = k + 1$.
For the sake of simplicity, let us prove it for $k = 2$ i.e. $n = 3$, assuming it holds for $n = 2$.
Its generalization to $k > 2$ is obvious.
Without loss of generality, let $\mathbf{x} = (x_1, x_2, x_3)^T$ and $\mathbf{I} = [-a, a]^3 \supseteq \text{Conv}(\mathcal{R}), a \in \mathbb{R}$
Given, $\int_{\mathbf{I}} f(\mathbf{x})d\mathbf{x} = \int_{\mathbf{I}} g(\mathbf{x})d\mathbf{x}$ and $\nabla f(\mathbf{x}) = \nabla g(\mathbf{x})$.
$\Rightarrow \frac{\partial}{\partial x_1} f(\mathbf{x}) = \frac{\partial}{\partial x_1} g(\mathbf{x})$.
Integrating both the sides with respect to x_1 leads to:

$$f(\mathbf{x}) = g(\mathbf{x}) + c(x_2, x_3) \tag{2.3}$$

where, $c(x_2, x_3)$ is some arbitrary function of x_2 and x_3.
Integrating equation (2.3) over $\mathbf{I}$, we get: $\int_{x_3} \int_{x_2} c(x_2, x_3)dx_2dx_3 = 0$
Integrating equation (2.3) with respect to x_1 from $-a$ to a, we get:

$$f_1(x_2, x_3) = g_1(x_2, x_3) + 2ac(x_2, x_3) \tag{2.4}$$

where, $f_1(x_2, x_3) = \int_{-a}^{a} f(\mathbf{x})dx_1$ and $g_1(x_2, x_3) = \int_{-a}^{a} g(\mathbf{x})dx_1$.
Integrating equation (2.4) with respect to both x_2 and x_3, we get: $\int_{x_3} \int_{x_2} f_1(x_2, x_3)dx_2dx_3 = \int_{x_3} \int_{x_2} g_1(x_2, x_3)dx_2dx_3$.
Integrating $\frac{\partial}{\partial x_2} f(\mathbf{x}) = \frac{\partial}{\partial x_2} g(\mathbf{x})$ with respect to x_1 from $-a$ to a, we get: $\frac{\partial}{\partial x_2} f_1(x_2, x_3) = \frac{\partial}{\partial x_2} g_1(x_2, x_3)$
Integrating $\frac{\partial}{\partial x_3} f(\mathbf{x}) = \frac{\partial}{\partial x_3} g(\mathbf{x})$ with respect to x_1 from $-a$ to a, we get: $\frac{\partial}{\partial x_3} f_1(x_2, x_3) = \frac{\partial}{\partial x_3} g_1(x_2, x_3)$
Applying $n = 2$ case, with all the required conditions satisfied, we get: $f_1(x_2, x_3) = g_1(x_2, x_3)$
Therefore, from equation (2.4), $c(x_2, x_3) = 0$. This proves the Lemma for $n = k + 1$.
Combining both the base case and inductive step, by mathematical induction, the Theorem holds for all natural n. □

Lemma 2.3. *Let $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}$ and both satisfy the following conditions:*

1. *They are second order differentiable.*
2. $\int_{\mathcal{R}} f(\mathbf{x})d\mathbf{x} = \int_{\mathcal{R}} g(\mathbf{x})d\mathbf{x},\ \mathbf{x} \in \mathbb{R}^n,\ \mathcal{R} = \mathit{supp}(f) \bigcup \mathit{supp}(g)$
3. *They have bounded support.*

Then, the following holds:

$$\nabla^2 f(\mathbf{x}) = \nabla^2 g(\mathbf{x}) \Rightarrow f(\mathbf{x}) = g(\mathbf{x}) \tag{2.5}$$

Proof. Let us prove this Lemma by mathematical induction.

The Base Case: $n = 1$

Without loss of generality, let $\mathbf{I} = [-a, a] \supseteq \text{Conv}(\mathcal{R}), a \in \mathbb{R}$

Given $\int_{\mathbf{I}} f(x)dx = \int_{\mathbf{I}} g(x)dx$ and $\frac{d^2}{dx^2} f(x) = \frac{d^2}{dx^2} g(x)$.Double integrating both the sides of latter equation with respect to x leads to,

$$f(x) \quad = \quad g(x) + c_1 x + c_2 \tag{2.6}$$

where, c_1 and c_2 are some arbitrary constant.

Integrating both the sides of Equation (2.6) with respect to x from $-a$ to a, brings $c_2 = 0$.

Integrating both the sides of Equation (2.6) with respect to x from $-a$ to b, $b > a, , b \in \mathbb{R}$ brings $c_1 = 0$.

This proves the Theorem for the base case.

The induction step:

Given the Lemma holds for $n = k$, let us prove it for $n = k + 1$.

For the sake of simplicity, let us prove it for $k = 2$ i.e. $n = 3$, assuming it holds for $n = 2$. Its generalization to $k > 2$ is obvious.

Without loss of generality, let $\mathbf{x} = (x_1, x_2, x_3)^T$ and $\mathbf{I} = [-a, a]^3 \supseteq \text{Conv}(\mathcal{R}), a \in \mathbb{R}$

Given $\int_{\mathbf{I}} f(\mathbf{x})d\mathbf{x} = \int_{\mathbf{I}} g(\mathbf{x})d\mathbf{x}$; $\nabla^2 f(\mathbf{x}) = \nabla^2 g(\mathbf{x})$.

$\Rightarrow \frac{\partial^2}{\partial {x_1}^2} f(\mathbf{x}) = \frac{\partial^2}{\partial {x_1}^2} g(\mathbf{x})$. Integrating twice both the sides with respect to x_1 leads to:

$$f(\mathbf{x}) = g(\mathbf{x}) + c_1(x_2, x_3)x_1 + c_2(x_2, x_3) \tag{2.7}$$

where, $c_1(x_2, x_3)$ and $c_2(x_2, x_3)$ are some arbitrary functions of x_2 and x_3.

Integrating Equation (2.7) over $\mathbf{I}$, we get: $\int_{x_3} \int_{x_2} c_2(x_2, x_3)dx_2 dx_3 = 0$

Integrating Equation (2.7) with respect to x_1 from $-a$ to a, we get:

$$f_1(x_2, x_3) = g_1(x_2, x_3) + 2ac_2(x_2, x_3) \tag{2.8}$$

where, $f_1(x_2,x_3) = \int_{-a}^{a} f(\mathbf{x})dx_1$ and $g_1(x_2,x_3) = \int_{-a}^{a} g(\mathbf{x})dx_1$.
Integrating Equation (2.8) with respect to both x_2 and x_3, we get: $\int_{x_3}\int_{x_2} f_1(x_2,x_3)dx_2dx_3 = \int_{x_3}\int_{x_2} g_1(x_2,x_3)dx_2dx_3$
Integrating $\frac{\partial^2}{\partial {x_2}^2} f(\mathbf{x}) = \frac{\partial^2}{\partial {x_2}^2} g(\mathbf{x})$ with respect to x_1 from $-a$ to a, we get: $\frac{\partial^2}{\partial {x_2}^2} f_1(x_2,x_3) = \frac{\partial^2}{\partial {x_2}^2} g_1(x_2,x_3)$
Integrating $\frac{\partial^2}{\partial {x_3}^2} f(\mathbf{x}) = \frac{\partial^2}{\partial {x_3}^2} g(\mathbf{x})$ with respect to x_1 from $-a$ to a, we get: $\frac{\partial^2}{\partial {x_3}^2} f_1(x_2,x_3) = \frac{\partial^2}{\partial {x_3}^2} g_1(x_2,x_3)$
Applying, $n = 2$ case, with all conditions satisfied, we get: $f_1(x_2,x_3) = g_1(x_2,x_3)$
Therefore, from Equation (2.8), $c_2(x_2,x_3) = 0$.
Integrating the Equation (2.7) with respect to x_1 from $-a$ to b, $b > a, b \in \mathbb{R}$, we get: $c_1(x_2,x_3) = 0$
This proves the Lemma for $n = k+1$.
Combining both the base case and inductive step, by mathematical induction, the Lemma for all natural n. □

Lemma 2.4. *Let $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}$ and both satisfy the following conditions:*

1. *They are p^{th} order differentiable.*
2. $\int_{\mathcal{R}} f(\mathbf{x})d\mathbf{x} = \int_{\mathcal{R}} g(\mathbf{x})d\mathbf{x}$, $\mathbf{x} \in \mathbb{R}^n$, $\mathcal{R} = supp(f) \bigcup supp(g)$
3. *They have bounded support.*

Then, the following holds:

$$\nabla^p f(\mathbf{x}) = \nabla^p g(\mathbf{x}) \Rightarrow f(\mathbf{x}) = g(\mathbf{x}) \tag{2.9}$$

Proof. The Theorem 2.2 proves this for $p = 1$ and the Lemma 2.3 proves the same for $p = 2$. Here, it needs be proved for any $p > 2$.
Let us prove this Lemma by mathematical induction.
The Base Case: $n = 1$
Without loss of generality, let $\mathbf{I} = [-a_1, a_1] \supseteq \text{Conv}(\mathcal{R}), a_1 \in \mathbb{R}$
Given $\int_{\mathbf{I}} f(x)dx = \int_{\mathbf{I}} g(x)dx$ and $\frac{d^p}{dx^p} f(x) = \frac{d^p}{dx^p} g(x)$.
Integrating p times both the sides of latter equation with respect to x leads to,

$$f(x) \;\; = \;\; g(x) + c_1x^{p-1} + c_2x^{p-2} + \cdots + c_p \tag{2.10}$$

where, $c_i, i = \{1, 2, \ldots, p\}$ are some arbitrary constant.
We can have two cases: Let p be even.
Integrating both the sides of Equation (2.10) with respect to x from $-a_1$ to a_1, brings $a_{11}c_2 +$

$a_{12}c_4 + \ldots + a_{1q}c_p = 0$, where $q = p/2$ and a_{1i}s are the coefficients as a result of integration.
Let there be $q-1$ real numbers $a_i, i = \{2, 3, \ldots q\}$ such that $a_i > a_1$ and each one is different from the other. Then, integrating (2.10) with respect to x from $-a_i$ to a_i, brings over all q equations with coefficients $a_{ij}, i = \{1, 2, \ldots, q\}, j = \{1, 2, \ldots, q\}$. Representing them in a matrix form, $\mathbf{A}\mathbf{c} = 0$, where $\mathbf{A} = [a_{ij}], \forall\, a_{ij} \neq 0$ and $\mathbf{c} = (c_2, c_4, \ldots, c_p)^T$. The only solution to this equation is: $c_i = 0, i = \{2, 4, \ldots, p\}$ i.e. all $c_i, i = \forall$ even in Equation 2.10 are zero.
Now, let there be q real numbers $b_i > a_i, i = \{1, 2, \ldots, q\}$ such that none of them is equal to the other. Integrating both the sides of Equation (2.10) with respect to x from $-a_i$ to b_i brings $b_{i1}c_1 + b_{i2}c_3 + \ldots + b_{iq}c_{p-1} = 0$, where $q = p/2$ and $b_{ij}, j = \{1, 2, \ldots, q\}$ are the coefficients as a result of integration. In a matrix form, $\mathbf{B}\mathbf{c} = 0$, where $\mathbf{B} = [b_{ij}]$ and $\mathbf{c} = (c_1, c_3, \ldots, c_{p-1})^T$. This brings all $c_i, i =$ odd also to be zero.
This proves the lemma from Equation (2.10) for p even case.
The p odd case can also be solved similarly.
This proves the Lemma for the base case.
The induction step:
Given the Lemma holds for $n = k$, let us prove it for $n = k + 1$.
For the sake of simplicity, let us prove it for $k = 2$ i.e. $n = 3$, assuming it holds for $n = 2$.
Its generalization to $k > 2$ is obvious.
Without loss of generality, let $\mathbf{x} = (x_1, x_2, x_3)^T$ and $\mathbf{I} = [-a_1, a_1]^3 \supseteq \text{Conv}(\mathcal{R}), a \in \mathbb{R}$
Given $\int_{\mathbf{I}} f(\mathbf{x})d\mathbf{x} = \int_{\mathbf{I}} g(\mathbf{x})d\mathbf{x}$; $\nabla^p f(\mathbf{x}) = \nabla^p g(\mathbf{x})$.
$\Rightarrow \frac{\partial^p}{\partial {x_1}^p} f(\mathbf{x}) = \frac{\partial^p}{\partial {x_1}^p} g(\mathbf{x})$. Integrating p times both the sides with respect to x_1 leads to:

$$f(\mathbf{x}) = g(\mathbf{x}) + c_1(x_2, x_3)x_1^{p-1} + c_2(x_2, x_3)x_1^{p-2} + \ldots + c_p(x_2, x_3) \tag{2.11}$$

where, $c_i(x_2, x_3), i = \{1, 2, \ldots, p\}$ are some arbitrary functions of x_2 and x_3.
Let p be even.
Integrating Equation (2.11) over $\mathbf{I}$, we get:
$\int_{x_3}\int_{x_2} \{a_{11}c_2(x_2, x_3)x_1^{p-2} + a_{12}c_4(x_2, x_3)x_1^{p-4} + \ldots + a_{1q}c_p(x_2, x_3)\}dx_2dx_3 = 0$
where $q = p/2$ and a_{1i} are the relevant coefficients.
Integrating Equation (2.11) with respect to x_1 from $-a_1$ to a_1, we get:

$$f_1(x_2, x_3) = g_1(x_2, x_3) + a_{11}c_2(x_2, x_3)x_1^{p-2} + a_{12}c_4(x_2, x_3)x_1^{p-4} + \ldots + a_{1q}c_p(x_2, x_3) \tag{2.12}$$

where, $f_1(x_2, x_3) = \int_{-a}^{a} f(\mathbf{x})dx_1$ and $g_1(x_2, x_3) = \int_{-a}^{a} g(\mathbf{x})dx_1$.
Integrating Equation (2.12) with respect to both x_2 and x_3, we get: $\int_{x_3}\int_{x_2} f_1(x_2, x_3)dx_2dx_3 = \int_{x_3}\int_{x_2} g_1(x_2, x_3)dx_2dx_3$
Integrating $\frac{\partial^2}{\partial {x_2}^2} f(\mathbf{x}) = \frac{\partial^2}{\partial {x_2}^2} g(\mathbf{x})$ with respect to x_1 from $-a_1$ to a_1, we get: $\frac{\partial^2}{\partial {x_2}^2} f_1(x_2, x_3) =$

$\frac{\partial^2}{\partial {x_2}^2} g_1(x_2, x_3)$
Integrating $\frac{\partial^2}{\partial {x_3}^2} f(\mathbf{x}) = \frac{\partial^2}{\partial {x_3}^2} g(\mathbf{x})$ with respect to x_1 from $-a_1$ to a_1, we get: $\frac{\partial^2}{\partial {x_3}^2} f_1(x_2, x_3) = \frac{\partial^2}{\partial {x_3}^2} g_1(x_2, x_3)$
Applying, $n = 2$ case, with all conditions satisfied, we get: $f_1(x_2, x_3) = g_1(x_2, x_3)$
Therefore, from Equation (2.12), $a_{11}c_2(x_2, x_3)x_1^{p-2} + a_{12}c_4(x_2, x_3)x_1^{p-4} + \ldots + a_{1q}c_p(x_2, x_3) = 0$.
Similar to the $n = 1$ case, we can form $q - 1$ such other independent equations, solving them we get: $c_i = 0 . \forall i$ even
Integrating the Equation (2.11) with respect to x_1 from $-a_1$ to b_1, $b_1 > a_1, b_1 \in \mathbb{R}$, we get:
$b_{11}c_1(x_2, x_3)x_1^{p-1} + b_{12}c_3(x_2, x_3)x_1^{p-3} + \ldots + a_{1q}c_{p-1}(x_2, x_3)x_1 = 0$
Similar to the $n = 1$ case, we can form $q - 1$ such other independent equations, solving them we get: $c_i = 0 . \forall i$ odd
This proves the Lemma for $n = k + 1$.
Combining both the base case and inductive step, by mathematical induction, the Lemma for all natural n. □

Lemma 2.5. *Let $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}$ and both satisfy the following conditions:*

1. *They are p^{th} order differentiable.*
2. $\int_{\mathcal{R}} f(\mathbf{x})d\mathbf{x} = \int_{\mathcal{R}} g(\mathbf{x})d\mathbf{x},\ \mathbf{x} \in \mathbb{R}^n,\ \mathcal{R} = supp(f) \bigcup supp(g)$
3. *They have bounded support.*

Then, the following holds:

$$f(\mathbf{x}) = g(\mathbf{x}) \Leftrightarrow \nabla^p f(\mathbf{x}) = \nabla^p g(\mathbf{x}) \tag{2.13}$$

Proof. Given $f(\mathbf{x})$ and $g(\mathbf{x})$ are differentiable: $f(\mathbf{x}) = g(\mathbf{x}) \Rightarrow \nabla^p f(\mathbf{x}) = \nabla^p g(\mathbf{x})$
The converse part is proved in Lemma 2.4. This proves the current Lemma. □

Theorem 2.6. *Let $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}$ and both satisfy the following conditions:*

1. *They are p^{th} order differentiable.*
2. $\int_{\mathcal{R}} f(\mathbf{x})d\mathbf{x} = \int_{\mathcal{R}} g(\mathbf{x})d\mathbf{x},\ \mathbf{x} \in \mathbb{R}^n,\ \mathcal{R} = supp(f) \bigcup supp(g)$
3. *They have bounded support.*

Then, the following holds:

$$f(\mathbf{x}) = g(\mathbf{x}) \Leftrightarrow \nabla f(\mathbf{x}) = \nabla g(\mathbf{x}) \Leftrightarrow \ldots \Leftrightarrow \nabla^p f(\mathbf{x}) = \nabla^p g(\mathbf{x}) \tag{2.14}$$

Proof. Applying principle of transitivity of implication (Hypothetical syllogism) to Lemma 2.5 with varying values of p, this Theorem is proved. □

For a generalized functions, given any p^{th} order derivatives are equal, the only available information would be that the functions differ by a constant in their $(p-1)^{th}$ order derivative. It would require p initial conditions to decide about equality of the functions. The Theorem 2.2 proves that if the given condition for $p = 1$ is added with one more condition of equal hypervolumes then it brings equality of the functions. The above Theorem 2.6 proves further the strength of an added prior information that the function is also having bounded support. This prior implies that any p^{th} order derivatives are equal, the functions are equal. Conversely, given two functions with equal p^{th} derivative are not equal imply either of the conditions are not matching. For example; let $f(x)$ and $g(x)$ are constant functions with unequal constant values and unequal supports on real line such that area under them are same. The derivatives are same and zero everywhere. The example seems counterexample of the Theorem 2.2 as both derivatives are same but not the functions. More better observation clears that both the functions are discontinuous at boundary points. This violates the differentiability condition of Theorem. The derivative values given zero, actually excludes points with Lebesgue measure zero.

2.3 Applications of the Results On Independence

By definition, the probability density functions have area under the curve to be unity. The bounded support function assumption seems restricting application to many PDFs. But, as said in the Section 2.1, empirically and numerically this assumption is not restricting. So, it is natural to think of extending the previous results to independence condition. Looking similarity with the results on Score Function Difference (SFD) and its properties related to independence in (7), the topic is developed using matching terminology.
Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ is an n-dimensional random vector, where, $x_i, i = 1, 2, \ldots, n$ are random variables; $p_{\mathbf{x}}(x_1, x_2, \ldots, x_n)$ is the joint PDF of $\mathbf{x}$ and $\prod_{i=1}^{n} p_{x_i}(x_i)$ is the product of the marginal PDFs. For this description, the statistical independence as in (84), and other terms are defined.

Definition 2.7 (Statistical Independence)**.** The random variables $x_1, x_2, \ldots, x_n$ are said to be statistically independent, if

$$p_{\mathbf{x}}(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} p_{x_i}(x_i)$$

As the statistical independence finds many applications, it is worth defining the following term.

Definition 2.8 (Function Difference (FD))**.** The Function Difference (FD) of $\mathbf{x}$ is the difference between product of its marginal PDFs $\prod_{i=1}^{n} p_{x_i}(x_i)$ and its joint PDF $p_{\mathbf{x}}(x_1, x_2, \ldots, x_n)$, that is:

$$\boldsymbol{\Delta}(\mathbf{x}) \overset{def}{=} \prod_{i=1}^{n} p_{x_i}(x_i) - p_{\mathbf{x}}(x_1, x_2, \ldots, x_n)$$

From the Definition, $\boldsymbol{\Delta}(\mathbf{x}) \equiv 0$ implies independence.

With the assumption that the joint PDF and the marginal PDFs are differentiable, the followings are defined.

Definition 2.9 (GPF)**.** The Gradient of the Product Function (GPF) of $\mathbf{x}$ is the gradient of the product of the marginal PDFs $\prod_{i=1}^{n} p_{x_i}(x_i)$, that is:

$$\boldsymbol{\xi}(\mathbf{x}) \overset{def}{=} \nabla \left(\prod_{i=1}^{n} p_{x_i}(x_i) \right) = (\xi_1(x_1), \xi_2(x_2), \ldots, \xi_n(x_n))^T$$
$$\text{where, } \xi_l(x_l) \overset{def}{=} \frac{\partial}{\partial x_l} \left(\prod_{i=1}^{n} p_{x_i}(x_i) \right)$$

Definition 2.10 (GJF)**.** The Gradient of the Joint Function (GJF) of $\mathbf{x}$ is the gradient of the joint PDF $p_{\mathbf{x}}(x_1, x_2, \ldots, x_n)$, that is:

$$\boldsymbol{\zeta}(\mathbf{x}) \overset{def}{=} \nabla p_{\mathbf{x}}(x_1, x_2, \ldots, x_n) = (\zeta_1(\mathbf{x}), \zeta_2(\mathbf{x}), \ldots, \zeta_n(\mathbf{x}))^T$$
$$\text{where, } \zeta_l(\mathbf{x}) \overset{def}{=} \frac{\partial}{\partial x_l} p_{\mathbf{x}}(x_1, x_2, \ldots, x_n)$$

Definition 2.11 (GFD)**.** The Gradient Function Difference (GFD) of $\mathbf{x}$ is the difference between its GPF and GJF or equivalently it is the gradient of FD, that is:

$$\boldsymbol{\alpha}(\mathbf{x}) \overset{def}{=} \boldsymbol{\xi}(\mathbf{x}) - \boldsymbol{\zeta}(\mathbf{x}) = \nabla(\boldsymbol{\Delta}(\mathbf{x}))$$

The following property proves that GFD ($\boldsymbol{\alpha}(\cdot)$) contains important information about independence of the components of a random vector.

Property 1. *The components of a random vector* $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ *are independent if and only if* $\boldsymbol{\alpha}(\mathbf{x}) \equiv 0$*, that is:*

$$\boldsymbol{\xi}(\mathbf{x}) = \boldsymbol{\zeta}(\mathbf{x}) \tag{2.15}$$

Proof. Let us take for better analogy, $f(\mathbf{x}) = p_{\mathbf{x}}(x_1, x_2, \ldots, x_N)$ and $g(\mathbf{x}) = \prod_{i=1}^{N} p_{x_i}(x_i)$
$\Rightarrow \int_{\mathcal{R}} f(\mathbf{x}) d\mathbf{x} = \int_{\mathcal{R}} g(\mathbf{x}) d\mathbf{x} = 1$
Given $\boldsymbol{\xi}(\mathbf{x}) = \boldsymbol{\zeta}(\mathbf{x})$ or $\nabla f(\mathbf{x}) = \nabla g(\mathbf{x})$.

The required conditions are satisfied. So, applying Theorem 2.2 and the Definition 2.7 of Independence, the property is proved. □

For the same random vector x with added assumptions that the joint PDF and the product of the marginal PDFs are both second order differentiable and have bounded support, the following definitions and results are obtained.

Definition 2.12 (HPF)**.** The Hessian of the Product Function (HPF) of x is the Hessian of the product of the marginal PDFs $\prod_{i=1}^{n} p_{x_i}(x_i)$, that is:

$$\Xi(\mathrm{x}) \overset{def}{=} \nabla \boldsymbol{\xi}(\mathrm{x}) = \nabla^2 \left(\prod_{i=1}^{n} p_{x_i}(x_i) \right)$$

Definition 2.13 (HJF)**.** The Hessian of the Joint Function (HJF) of x is the Hessian of the joint PDF $p_{\mathrm{x}}(x_1, x_2, \ldots, x_n)$, that is:

$$\boldsymbol{Z}(\mathrm{x}) \overset{def}{=} \nabla \boldsymbol{\zeta}(\mathrm{x}) = \nabla^2 p_{\mathrm{x}}(x_1, x_2, \ldots, x_n)$$

Definition 2.14 (HFD)**.** The Hessian Function Difference (HFD) of x is the difference between its HPF and HJF, or equivalently it is the Hessian of FD, that is:

$$\begin{aligned} \boldsymbol{A}(\mathrm{x}) &\overset{def}{=} \Xi(\mathrm{x}) - \boldsymbol{Z}(\mathrm{x}) = \nabla\left(\boldsymbol{\xi}(\mathrm{x}) - \boldsymbol{\zeta}(\mathrm{x})\right) \\ &= \nabla^2(\boldsymbol{\Delta}(\mathrm{x})) = \nabla \boldsymbol{\alpha}(\mathrm{x}) \end{aligned}$$

The following property proves that HFD ($\boldsymbol{A}(\cdot)$) contains important information about independence of the components of a random vector.

Property 2. *The components of a bounded support random vector* $\mathrm{x} = (x_1, x_2, \ldots, x_n)^T$ *are independent if and only if* $\boldsymbol{A}(\mathrm{x}) \equiv 0$*, that is:*

$$\Xi(\mathrm{x}) = \boldsymbol{Z}(\mathrm{x}) \tag{2.16}$$

Proof. Applying Lemma 2.4 with $p = 2$, the property is proved. □

Corollary 2.15. *Let* $\mathrm{x} = (x_1, x_2, \ldots, x_n)^T$ *be an n-dimensional random vector;* $p_{\mathrm{x}}(x_1, x_2, \ldots, x_n)$ *be its joint PDF;* $\prod_{i=1}^{n} p_{x_i}(x_i)$ *be its product of the marginal PDF; the PDFs be second order differentiable with bounded support* $\mathcal{R} \subseteq \mathbb{R}^n$*. Then:*

$$p_{\mathrm{x}}(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} p_{x_i}(x_i) \Leftrightarrow \boldsymbol{\xi}(\mathrm{x}) = \boldsymbol{\zeta}(\boldsymbol{x}) \Leftrightarrow \Xi(\mathrm{x}) = \boldsymbol{Z}(\mathrm{x}) \tag{2.17}$$

Proof. Applying Theorem 2.6 and the Definition 2.7 of independence, the corollary is proved. □

The Property 1 of GFD, Property 2 of HFD and the Corollary 2.15 bring further interpretations on independence of bounded support random vector. Our goal is to develop new contrasts based on them. For that the quantities should be nonnegative to be quantified as measures. So, first let there be derived independence measures based on these results.

2.4 Deriving new Independence Measures

The goal here is to derive independence measures based on the quantities FD, GFD and HFD. But, the quantities do not assure nonnegativity to be quantified as measures. Assuming a class of L^p integrable PDFs, the L^p norm can be applied on them. Being norm, they satisfy all the properties of a *metric* and an added property of absolute scale invariance, as per the definition of norm. The details on the definitions of a measure, a *metric*, a norm and the specific L^p-norm are briefed in Appendix A.

It is desired that a distance measure between PDFs is invariant with respect to translation and scaling i.e. the deviation in mean and the variance should not affect the distance measure. The reason is, the nearness of the PDFs should imply their shapes are matching. The desired property of scale invariance, instead of the absolute scale invariance, can be assured by defining an independence measure that applies a norm on normalized PDFs i.e. converting them first into zero mean, univariance PDFs.

Proposition 2.16. *For a random vector* $\mathbf{x} \in \mathbb{R}^n$ *with* L^p *integrable joint and marginal PDFs, LpFD(*$\mathbf{x}$*) or* $\boldsymbol{\Delta}_p(\mathbf{x})$ *defined as under is an independence measure.*

$$\boldsymbol{\Delta}_p(\mathbf{x}) \overset{def}{=} ||\boldsymbol{\Delta}(\mathbf{z})||_p = \left(\int_{\mathbb{R}^n} |\boldsymbol{\Delta}(\mathbf{z})|^p \, d\mathbf{z}\right)^{\frac{1}{p}} \tag{2.18}$$

$$\textit{or } d_p\left(\prod_{i=1}^{n} p_{x_i}(x_i), p_{\mathbf{x}}(\mathbf{x})\right) = \left(\int_{\mathbf{x}} \left|\prod_{i=1}^{n} p_{x_i}\left(\frac{x_i - \bar{x}_i}{\sigma_{x_i}}\right) - p_{\mathbf{x}}\left(\frac{\mathbf{x} - \bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right)\right|^p d\mathbf{x}\right)^{\frac{1}{p}} \tag{2.19}$$

where, $\mathbf{z} = \left(\frac{\mathbf{x}-\bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right)$, $\bar{\mathbf{x}}$ *and* $\bar{x}_i$ *are consecutively mean of* $\mathbf{x}$ *and* x_i*,* $\sigma_{\mathbf{x}}$ *and* σ_{x_i} *are corresponding standard deviations.*

Proof. By definition, $\boldsymbol{\Delta}_p(\mathbf{x}) \geq 0$ and $\boldsymbol{\Delta}_p(\mathbf{x}) = 0$ if and only if $\boldsymbol{\Delta}(\mathbf{x}) \equiv 0$. Also, by Definition 2.7 of independence, $\boldsymbol{\Delta}_p(\mathbf{x}) = 0$ if and only if the components of $\mathbf{x}$ are independent.
This proves that $\boldsymbol{\Delta}_p(\mathbf{x})$ is an independence measure. More specifically, it is an independence metric with respect to $\boldsymbol{\Delta}(\mathbf{x})$, but not necessarily on the space of random vectors $\mathbf{x}$ themselves. □

The GFD is essentially a vector, whose value is an n-tuple of functions. Accordingly, $\boldsymbol{\alpha} : L^p \times L^p \times \ldots L^p \rightarrow \mathbb{R}$. So, L^p-norm can still be applied as under.

Proposition 2.17. *For an n-dimensional random vector* $\mathbf{x} = (x_1, x_2, ..., x_n)$ *with differentiable joint and marginal PDFs, LpGFD(*$\mathbf{x}$*) or* $\boldsymbol{\alpha}_p(\mathbf{x})$ *defined as under is an independence measure.*

$$\boldsymbol{\alpha}_p(\mathbf{x}) \overset{def}{=} ||\boldsymbol{\alpha}(\mathbf{z})||_p = \left(\sum_{i=1}^{n} (||\alpha_i(\mathbf{z})||_p)^p\right)^{\frac{1}{p}} \tag{2.20}$$

$$\text{or } d_p\left(\xi(\mathbf{x}), \zeta(\mathbf{x})\right) = \left(\sum_{i=1}^{n} \int_{\mathbf{z}} \left|\xi_i\left(\frac{\mathbf{x}-\bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right) - \zeta_i\left(\frac{\mathbf{x}-\bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right)\right|^p d\mathbf{z}\right)^{\frac{1}{p}} \tag{2.21}$$

where, $\mathbf{z} = \left(\frac{\mathbf{x}-\bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right)$, $\bar{\mathbf{x}}$ *is the mean of* $\mathbf{x}$ *and* $\sigma_{\mathbf{x}}$ *is the corresponding standard deviation.*

Proof. The differentiable PDF condition, assures L^p integrability.
By definition, $\boldsymbol{\alpha}_p(\mathbf{x}) \geq 0$ and $\boldsymbol{\alpha}_p(\mathbf{x}) = 0$ if and only if $\boldsymbol{\alpha}(\mathbf{x}) \equiv 0$. Applying Property 1, $\boldsymbol{\alpha}_p(\mathbf{x}) = 0$ if and only if the components of $\mathbf{x}$ are independent.
This proves that $\boldsymbol{\alpha}_p(\mathbf{x})$ is an independence measure. More specifically, it is an independence metric with respect to $\boldsymbol{\alpha}(\mathbf{x})$, but not necessarily on the space of random vectors $\mathbf{x}$ themselves. □

The HFD is essentially a matrix. So, matrix norms are applicable. The '*Entrywise*' norms treat matrix entries as a vector entries. The following independence measure can be defined.

Proposition 2.18. *For a bounded support random vector* $\mathbf{x} = (x_1, x_2, ..., x_N)$ *with second order differentiable joint and marginal PDFs, LpHFD(*$\mathbf{x}$*) or* $\boldsymbol{A}_p$ *is an independence measure, where:*

$$\boldsymbol{A}_p(\mathbf{x}) \overset{def}{=} ||\boldsymbol{A}(\mathbf{z})||_p = \left(\sum_{j=1}^{n}\sum_{i=1}^{n} (||A_{ij}(\mathbf{z})||_p)^p\right)^{\frac{1}{p}} \tag{2.22}$$

$$\text{or } d_p\left(\Xi(\mathbf{x}), \boldsymbol{Z}(\mathbf{x})\right) = \left(\sum_{j=1}^{n}\sum_{i=1}^{n} \int_{z_{ij}} \left|\Xi_{ij}\left(\frac{\mathbf{x}-\bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right) - Z_{ij}\left(\frac{\mathbf{x}-\bar{\mathbf{x}}}{\sigma_{\mathbf{x}}}\right)\right|^p dz_{ij}\right)^{\frac{1}{p}} \tag{2.23}$$

where, $\bar{\mathbf{x}}$ is the mean of $\mathbf{x}$ and $\sigma_{\mathbf{x}}$ is the corresponding standard deviation.

Proof. The second order differentiable PDF condition, assures L^p integrability.
By definition, $\boldsymbol{A}_p \geq 0$ and $\boldsymbol{A}_p(\mathbf{x}) = 0$ if and only if $\boldsymbol{A}(\mathbf{x}) \equiv 0$. Applying property 2, $\boldsymbol{A}_p = 0$ if and only if the components of $\mathbf{x}$ are independent.
This proves that $\boldsymbol{A}_p$ is an independence measure. More specifically, it is an independence metric with respect to $\boldsymbol{A}(\mathbf{x})$, but not necessarily on the space of random vectors $\mathbf{x}$ themselves. □

2.5 The Linear BSS Problem and Solution

The Blind Source Separation (BSS) model explains generation of an observed random vector $\mathbf{x}(t)$, as an transformation to another latent (hidden) random vector $\mathbf{s}(t)$. Assuming linear and instantaneous transformation, mathematically, $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$, where $\mathbf{x}(t) = [x_1(t); x_2(t); \ldots; x_m(t)]$; $\mathbf{s}(t) = [s_1(t); s_2(t); \ldots; s_n(t)]$; $x_i(t)$, $s_i(t)$ are random variables with values in $\mathcal{R}$; $m = n >= 2$ and $\mathbf{A}$ is full rank. Let there be available N umber of samples of each observed random variable. Assuming an identical distribution, the instantaneous model can be extended for N realizations. Let $\mathbf{X}(t) = [\mathbf{x}_1(t); \mathbf{x}_2(t); \ldots; \mathbf{x}_m(t)]$ be the $m \times N$ data or observation matrix and $\mathbf{S}(t)$ be the $n \times N$ component or source matrix. Then,

$$\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) \tag{2.24}$$

The problem of BSS is to estimate both the unknowns $\mathbf{A}$ and $\mathbf{S}(t)$, with the only assumption of $\mathbf{s}_i(t)$ being mutually the *most independent possible (m.i.p.)* random variables with respect to a given contrast. If $\mathbf{W}$ is the estimated inverse of the mixing matrix $\mathbf{A}$ then the estimated source or component matrix $\mathbf{Y}(t)$ is:

$$\mathbf{Y}(t) = \mathbf{A}^{-1}\mathbf{X}(t) = \mathbf{W}\mathbf{X}(t) = \mathbf{W}\mathbf{A}\mathbf{S}(t) \tag{2.25}$$

As, $\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) = (\mathbf{A}\Lambda^{-1}\mathbf{P}^{-1})(\mathbf{P}\Lambda\mathbf{S}(t))$, for any permutation matrix $\mathbf{P}$ and a scaling matrix Λ, there are going to be scaling and permutation ambiguities in the estimated components.

Given the unknown sources are *independent and identically distributed* (*i.i.d.*) with maximum one of them being Gaussian, a unique BSS solution is assured by Darmois-Skitovtch Theorem (33, 35, 46). Accordingly, the BSS solution for linear, instantaneous mixing system can be obtained by maximizing the independence among $y_i(t)$s with respect to the separation matrix $\mathbf{W}$, as:

$$\mathbf{y}^*(t) = \underset{\mathbf{W}}{\text{argmax}}\ \Phi(\mathbf{y}(t)) \tag{2.26}$$

where, $\Phi(\mathbf{y}(t))$ is the optimization function, based on independence or dependence measure, that assures source separation on maximization. It is identified as a contrast function or simply a 'contrast'. Overall, the BSS solution demands a suitable contrast function as an optimization criteria and a suitable optimization technique corresponding to that contrast function.

2.6 Deriving New Contrasts for ICA and BSS

A formal definition of contrasts, based on references (34) and (35, Chapter 3), for BSS is as under.

Definition 2.19 (Contrast for BSS)**.** Let $\mathcal{H}$ be a set of static transformations (filters) containing an identity transformation (filter) **I**; $\mathcal{S}$ be a set of source random variables that are independent and ; $\mathcal{X} = \mathcal{H} \cdot \mathcal{S}$ be the set of random variables obtained by the action of $\mathcal{H}$ on $\mathcal{S}$; Φ be a mapping from $\mathcal{H} \times \mathcal{H} \cdot \mathcal{S}$ to $\mathbb{R}$. Also, denoted by $\mathcal{T}$ the set of trivial filters of $\mathcal{H}$, which leave criterion Φ unchanged. A mapping $\Phi(\mathbf{H}; \mathbf{x})$ is a contrast if it depends solely on the PDF of $\mathbf{x}$ and if it satisfies the following three properties below.

a. Invariance: $\forall \mathbf{x} \in \mathcal{X}, \forall \mathbf{T} \in \mathcal{T}, \Phi(\mathbf{T}; \mathbf{x}) = \Phi(\mathbf{I}; \mathbf{x})$

b. Dominance: $\forall \mathbf{s} \in \mathcal{S}, \forall \mathbf{H} \in \mathcal{H}, \Phi(\mathbf{H}; \mathbf{s}) \leq \Phi(\mathbf{I}; \mathbf{s})$

c. Discrimination: $\forall \mathbf{s} \in \mathcal{S},\ if\ \mathbf{H} \in \mathcal{H}\ satisfies$
$\Phi(\mathbf{H}; \mathbf{s}) = \Phi(\mathbf{I}; \mathbf{s}),\ then\ \mathbf{H} \in \mathcal{T}$

The Dominance property assures that the actual sources have the global maxima. The Discrimination property assures that there is no other spurious solution achieving the global maxima. There is some discussion needed on the invariance property. It is obvious that the independence components found using a given measure, are still independent if permuted or scaled. So, one of the solutions is available, whole class of solutions related through permutation and scaling operation is available. The Invariance property assures this by stating that whole class should have a same measure. The widely used *KL-divergence* assures this property. But, it is known that many other PDF divergence measures such as; Itakura-Saito distance, density-power divergences do not assure this scale invariance property. To accommodate such a larger class of divergences, without deteriorating the BSS performance, there has been first defined and then proposed relative scale invariance property as the sufficient property with other properties to be quantified as contrast.

Definition 2.20. The contrast $\Phi : \mathcal{H} \times \mathcal{H} \cdot \mathcal{S} \to \mathbb{R}$ is said to have relative Scale Invariance property; if it satisfies the following condition: Given $\mathbf{y} = \mathbf{\Lambda x}$

$$\Phi(\mathbf{y}) = k(\Lambda)\Phi(\mathbf{x})$$

where, $k(\Lambda)$ is a fixed transformation as a function of Λ.

Proposition 2.21. $\Phi : \mathcal{H} \times \mathcal{H} \cdot \mathcal{S} \to \mathbb{R}$ *is a contrast for linear BSS, if it satisfies the Relative Scale Invariance property with other required properties satisfied.*

Proof. The following arguments justify the proposition.

- Given $\mathbf{T} \in \mathcal{T}$ is a scale matrix with diagonal entries only. As the source components are independent, $\Phi(\mathbf{s}) = 0$. From the definition of the relative scale invariance property, $k(\mathbf{T})$ is

a predefined transformation acting as a scaling factor. So, $\Phi(\mathbf{y}) = 0. \Rightarrow \forall \mathbf{T} \in \mathcal{T}, \Phi(\mathbf{T}; \mathbf{s}) = \Phi(\mathbf{I}; \mathbf{s}) = 0$
As per this argument, scale invariance is required corresponding to the source components $\mathbf{s}$ and not necessarily with respect to $\mathbf{x}$. This is satisfied by the contrasts measuring 0 corresponding to independence and satisfying relative scale invariant.

- By definition, the relation between the measures corresponding to $\mathbf{x}$ components and their scaled version $\mathbf{Tx}$ components is known. $\forall \mathbf{T} \in \mathcal{T}, \Phi(\mathbf{T}; \mathbf{x}) = k(\mathbf{T})\Phi(\mathbf{I}; \mathbf{x})$
This assures the contrast measure for whole equivalence class of solutions are known.

- For the most BSS algorithms or precisely the orthogonal approach BSS algorithms $\mathbf{y} = \mathbf{Wx}$, where $\mathbf{W}$ is the estimated unmixing orthogonal rotation transformation and $\mathbf{x}$ are the equivariant uncorrelated (whiten) components. This implies that the measure is applied on the solution set that is equally scaled. Mathematically, $\mathbf{y} = \mathbf{\Lambda x}$, but $\mathbf{\Lambda}$ is a constant for the whole solution set. Also, corresponding $k(\mathbf{\Lambda})$ is constant for the whole solution set.

□

Though the relative scale invariance property is sufficient for a quantity to be a contrast, in most of the cases the quantity can be easily converted into a scale invariant quantity. This has been demonstrated for L^p norm of FD, GFD and HFD distance measures. Now, let us verify whether the derived independence measures qualify to be a contrast or not.

Proposition 2.22. *Φ_p^{FD} or $\Phi_p^{\mathbf{\Delta}} : \mathcal{H} \times \mathcal{H} \cdot \mathcal{S} \to \mathbb{R}$ is a contrast for linear BSS, where:*

$$\Phi_p^{FD}(\mathbf{H}; \mathbf{x}) \; or \; \Phi_p^{\mathbf{\Delta}}(\mathbf{H}; \mathbf{x}) = \Phi_p^{\mathbf{\Delta}}(\mathbf{y}) \overset{def}{=} -\mathbf{\Delta}_p(\mathbf{y}) = -d_p\left(\prod_{i=1}^{n} p_{y_i}(y_i), p_{\mathbf{y}}(\mathbf{y})\right)$$

Proof. Let us verify the scale invariance property of the contrast for both without and with normalization. Let $\mathbf{T} \in \mathcal{T}$ be $n \times n$ diagonal scaling matrix, as a trivial filter, with the non-zero diagonal entries $t_i, i = 1, \ldots, n$.

$$\begin{aligned} p_{\mathbf{Tx}}(t_1x_1, t_2x_2, \ldots, t_nx_n) &= \frac{1}{|det\mathbf{T}|} p_{\mathbf{x}}(x_1, x_2, \ldots, x_n) \\ p_{(\mathbf{Tx})_\mathbf{i}}((\mathbf{Tx})_\mathbf{i}) &= \frac{1}{|t_i|} p_{x_i}(x_i) \\ \Rightarrow \prod_{i=1}^{N} p_{x_i}(x_i) &= \frac{1}{|det\mathbf{T}|} \prod_{i=1}^{N} p_{x_i}(x_i) \end{aligned}$$

$$\begin{aligned}\text{Now, } \Phi_p^{\boldsymbol{\Delta}}(\mathbf{y}) &= -\boldsymbol{\Delta}_p(\mathbf{y}) = -||\boldsymbol{\Delta}(\mathbf{y})||_p \\ &= -\left(\int_{\mathbf{y}} \left|\prod_{i=1}^{n} p_{y_i}(y_i) - p_{\mathbf{y}}(\mathbf{y})\right|^p d\mathbf{y}\right)^{\frac{1}{p}} \\ &= -\left(\int_{\mathbf{x}} \left(\frac{1}{|\det \mathbf{T}|}\left|p_{\mathbf{x}}(\mathbf{x}) - \prod_{i=1}^{n} p_{x_i}(x_i)\right|\right)^p |\det \mathbf{T}| d\mathbf{x}\right)^{\frac{1}{p}} \\ &= -\,|det\mathbf{T}|^{\frac{1-p}{p}}\,\boldsymbol{\Delta}_p(\mathbf{x})\end{aligned}$$

This proves that the contrast $\Phi_p^{\boldsymbol{\Delta}}(\mathbf{y})$, without normalization of PDFs, is scale invariant for $p = 1$ i.e. corresponding to L^1-norm of $\boldsymbol{\Delta}$. It assures relative scale invariance for $1 < p < \infty$. As already discussed either the relative scale invariance is a sufficient condition or the measures are applied on normalized densities (i.e. densities with zero mean and unit variance) the scale invariance property is satisfied. Corresponding to normalized density, $t_i = 1, \forall i = 1, 2, \ldots, n.$
The permutation invariance can be proved in a same way as $|\det \mathbf{T}| = 1$.
The Proposition 2.18 proves the Dominance property.
By Definition 2.7, $\boldsymbol{\Delta}_p(\mathbf{y}) = 0$ if and only if the components $\mathbf{y} = \mathbf{Hs}$ are independent. So, $\mathbf{H}$ should be a trivial filter in $\mathcal{T}$. This proves the Discrimination property. □

Similarly, let us now verify whether the GFD is qualified to be a BSS contrast or not.

Proposition 2.23. *Φ_p^{GFD} or $\Phi_p^{\boldsymbol{\alpha}} : \mathcal{H} \times \mathcal{H} \cdot \mathcal{S} \to \mathbb{R}$ is a contrast for linear BSS, where:*

$$\Phi_p^{GFD}(\mathbf{H}; \mathbf{x}) \textit{ or } \Phi_p^{\boldsymbol{\alpha}}(\mathbf{H}; \mathbf{x}) = \Phi_p^{\boldsymbol{\alpha}}(\mathbf{y}) \overset{def}{=} -\boldsymbol{\alpha}_p(\mathbf{y}) = -d_p\left(\xi_{\mathbf{y}}(\mathbf{y}), \zeta_{\mathbf{y}}(\mathbf{y})\right)$$

Proof. Let us verify the scale invariance property of the contrast for both without and with normalization. Let $\mathbf{T} \in \mathcal{T}$ be $n \times n$ diagonal scaling matrix, as a trivial filter, with the non-zero diagonal entries $t_i, i = 1, \ldots, n$.
To simplify, let us start with the gradient of one dimensional transformed variable.

$$\begin{aligned}Y = aX \Rightarrow p_Y(y) &= \frac{1}{a} p_X\left(\frac{y}{a}\right) \\ \Rightarrow \frac{dp_Y(y)}{dy} &= \frac{1}{a^2} p_X\left(\frac{y}{a}\right) \\ \Rightarrow \int_y \frac{dp_Y(y)}{dy} dy &= \frac{1}{a}\int_x \frac{dp_X(x)}{dx} dx\end{aligned}$$

Let $\mathbf{y} = \mathbf{T}\mathbf{x}$.

$$\begin{aligned}\Rightarrow \Phi_p^{\text{GFD}} &= -\boldsymbol{\alpha}_p(\mathbf{y}) = -\left(\sum_{i=1}^{n}\int_{y_i}(\zeta_i(\mathbf{y}) - \xi_i(\mathbf{y}))^p\, dy_i\right)^{\frac{1}{p}}\\ &= -\left(\sum_{i=1}^{n}\int_{x_i}\left|\frac{1}{t_i^2}(\zeta_i(\mathbf{x}) - \xi_i(\mathbf{x}))\right|^p t_i dx_i\right)^{\frac{1}{p}}\\ &= -\left(\sum_{i=1}^{n}|t_i|^{1-2p}\left\|\alpha_i(\mathbf{x})\right\|_p\right)^{\frac{1}{p}}\end{aligned}$$

This proves, $\boldsymbol{\alpha}_p(\mathbf{y})$, without normalization, is neither scale invariant nor relative scale invariant. So, without normalization it can not be a BSS contrast, though being an independence measure. But, as already discussed the measures are applied on normalized densities i.e. densities with zero mean and unit variance, the scale invariance property is satisfied. Corresponding to normalization, $t_i = 1, \forall i = 1, 2, \ldots, n$.
The permutation invariance can be proved in a same way as $|\det \mathbf{T}| = 1$.
The Proposition 2.17 proves the Dominance property.
By Property 1, $\boldsymbol{\alpha}_p(\mathbf{y}) = 0$ if and only if the components $\mathbf{y} = \mathbf{H}\mathbf{s}$ are independent. So, $\mathbf{H}$ should be a trivial filter in $\mathcal{T}$. This proves the Discrimination property. □

Similarly, let us decide whether HFD - with and without normalization is qualified to be a BSS contrast or not.

Proposition 2.24. Φ_p^{HFD} *or* Φ_p^{A} $: \mathcal{H} \times \mathcal{H}\cdot\mathcal{S} \to \mathbb{R}$ *is a contrast for linear BSS of sources with bounded support, where:*

$$\Phi_p^{HFD}(\mathbf{H};\mathbf{x}) \text{ or } \Phi_p^{A}(\mathbf{H};\mathbf{x}) = \Phi_p^{A}(\mathbf{y}) \overset{def}{=} -A_p(\mathbf{y}) = -d_p\left(\Xi_{\mathbf{y}}(\mathbf{y}), Z_{\mathbf{y}}(\mathbf{y})\right)$$

Proof. Let us verify the scale invariance property of the contrast for both without and with normalization. Let $\mathbf{T} \in \mathcal{T}$ be $n \times n$ diagonal scaling matrix, as a trivial filter, with the non-zero diagonal entries $t_i, i = 1, \ldots, n$.

To simplify, let us start with the Hessian of one dimensional transformed variable.

$$\begin{aligned} Y = aX &\Rightarrow p_Y(y) = \frac{1}{a} p_X\left(\frac{y}{a}\right) \\ &\Rightarrow \frac{d^2 p_Y(y)}{dy^2} = \frac{1}{a^3} p_X\left(\frac{y}{a}\right) \\ &\Rightarrow \int_y \frac{d^2 p_Y(y)}{dy^2} dy = \frac{1}{a^2} \int_x \frac{dp_X(x)}{dx} dx \end{aligned}$$

Let $\mathbf{y} = \mathbf{T}\mathbf{x}$.

$$\begin{aligned} A_p(\mathbf{y}) &= \left(\sum_{j=1}^{n} \sum_{i=1}^{n} \int_{y_{ij}} \left(Z_{ij}(\mathbf{y}) - \Xi_{ij}(\mathbf{y})\right)^p dy_{ij} \right)^{\frac{1}{p}} \\ &= \left(\sum_{i=1}^{n} \sum_{i=1}^{n} |t_i|^{1-3p} \left\| A_{ij}(\mathbf{x}) \right\|_p \right)^{\frac{1}{p}} \end{aligned}$$

This proves, $A_p(\mathbf{y})$, without normalization, is neither scale invariant nor relative scale invariant. So, without normalization it is not a BSS contrast, though being an independence measure.
But, as already discussed the measures are applied on normalized densities i.e. densities with zero mean and unit variance, the scale invariance property is satisfied. Corresponding to normalization, $t_i = 1, \forall i = 1, 2, \ldots, n$
The permutation invariance can be proved in a same way as $|\det \mathbf{T}| = 1$.
The Proposition 2.18 proves the Dominance property.
By Property 2, $\mathbf{A}_p(\mathbf{y}) = 0$ if and only if the components $\mathbf{y} = \mathbf{H}\mathbf{s}$ are independent. So, $\mathbf{H}$ should be a trivial filter in $\mathcal{T}$. This proves the Discrimination property. □

2.6.1 Local Minima Analysis of the Proposed Contrasts

The contrasts defined using L^p-norm over FD, GFD and HFD have one more advantage that they do not have any local minima. This is a known property of L^p-norm, $p > 1$, proved as under:

$$\begin{aligned} \frac{d}{d\left\| f(x) \right\|} \left\| f(x) \right\|_p &= p \left\| f(x) \right\|^{p-1} \\ \therefore \frac{d}{d\left\| f(x) \right\|} \left\| f(x) \right\|_p &= 0 \Rightarrow \left\| f(x) \right\| = 0 \Rightarrow f(x) = 0, \forall x \end{aligned}$$

So, there is no separate proof required to show that the contrasts $\mathbf{\Delta}_p(\mathbf{y}(\theta))$, $\boldsymbol{\alpha}_p(\mathbf{y}(\theta))$ and $A_p(\mathbf{y}(\theta))$ do not have local minima with respect to the corresponding functions. But, still they may have local minima with respect to θ. Also, the estimation method may add local minima. Actually, it could be easily proved that the contrasts may contain local optima, as under.

$$\begin{aligned}
&\nabla \mathbf{\Delta}_p(\mathbf{y}_0) = 0 \\
&\Rightarrow \mathbf{\Delta}_p(\mathbf{y}_0) = c \text{ (an arbitrary constant)}
\end{aligned}$$

Obviously, as only $c = 0$ imply independence, other values of c correspond to possible local optima. The more detailed analysis follows as under.

Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ be a bounded random vector and $\delta = (\delta_1, \ldots, \delta_n)^T$ be a 'small' random vector. Then, the interest here is in the differential of $\Phi_p^{\mathbf{\Delta}}$ or $\|\mathbf{\Delta}(\mathbf{x}+\delta)\|_p - \|\mathbf{\Delta}(\mathbf{x})\|_p$.

$$\|\mathbf{\Delta}(\mathbf{x}+\delta)\|_p - \|\mathbf{\Delta}(\mathbf{x})\|_p = \int_{\mathbf{x}} \left| \prod_{i=1}^{n} p_{x_i+\delta_i}(\mathbf{x}) - p_{\mathbf{x}+\delta}(\mathbf{x}) \right|^p d\mathbf{x} - \int_{\mathbf{x}} \left| \prod_{i=1}^{n} p_{x_i}(\mathbf{x}) - p_{\mathbf{x}}(\mathbf{x}) \right|^p d\mathbf{x}$$

Assuming $\mathbf{t}$ as the support of all the PDFs,

$$\begin{aligned}
\|\mathbf{\Delta}(\mathbf{x}+\delta)\|_p - \|\mathbf{\Delta}(\mathbf{x})\|_p &= \int_{\mathbf{t}} \left| \prod_{i=1}^{n} p_{x_i+\delta_i}(\mathbf{t}) - p_{\mathbf{x}+\delta}(\mathbf{t}) \right|^p d\mathbf{t} - \int_{\mathbf{t}} \left| \prod_{i=1}^{n} p_{x_i}(\mathbf{t}) - p_{\mathbf{x}}(\mathbf{t}) \right|^p d\mathbf{t} \\
&= \int_{\mathbf{t}} |a-b|^p - |c-d|^p \, d\mathbf{t} \qquad \text{using symbolic notations}
\end{aligned}$$

where, $a = \prod_{i=1}^{n} p_{x_i+\delta_i}(\mathbf{t})$, $b = p_{\mathbf{x}+\delta}(\mathbf{t})$, $c = \prod_{i=1}^{n} p_{x_i}(\mathbf{t})$ and $d = p_{\mathbf{x}}(\mathbf{t})$.

Let's assume $p = 1$:

$$\begin{aligned}
&\|\mathbf{\Delta}(\mathbf{x}+\delta)\|_1 - \|\mathbf{\Delta}(\mathbf{x})\|_1 = 0 \\
&\Rightarrow \text{Either } \int_{\mathbf{t}} |a-b| \, d\mathbf{t} = \int_{\mathbf{t}} |c-d| \, d\mathbf{t} \\
&\text{or } |a-b| = |c-d|, \forall \mathbf{t} \\
&\text{or } a = b \text{ and } c = d, \forall \mathbf{t}
\end{aligned}$$

The condition $\int_{\mathbf{t}} |a-b| \, d\mathbf{t} = \int_{\mathbf{t}} |c-d| \, d\mathbf{t}$ do not assure gradient zero for optimal indicating independence condition.

As per $|a-b| = |c-d|, \forall \mathbf{t}$, four different cases can be thought:

$$\begin{aligned}
\text{Case I: } a > b, c > d &\Rightarrow a - b = c - d \Rightarrow a - c = b - d \Rightarrow \xi_{\mathbf{x}}(\mathbf{x}) = \zeta_{\mathbf{x}}(\mathbf{x}) \\
\text{Case II: } a > b, c < d &\Rightarrow a - b = -c + d \Rightarrow a + c = b + d \Rightarrow \text{ spurious optima} \\
\text{Case III: } a < b, c > d &\Rightarrow -a + b = c - d \Rightarrow a + c = b + d \Rightarrow \text{ spurious optima} \\
\text{Case IV: } a < b, c < d &\Rightarrow -a + b = -c + d \Rightarrow a - c = b - d \Rightarrow \xi_{\mathbf{x}}(\mathbf{x}) = \zeta_{\mathbf{x}}(\mathbf{x})
\end{aligned}$$

The Case I and Case IV imply independence but not the other cases.
The condition $a = b$ and $c = d, \forall \mathbf{t}$ also implies independence.
Over all, the analysis implies that the contrast $\Phi_1^{\mathbf{\Delta}}$ may have gradient zero indicating spurious maxima.

Let's assume $p = 2$:

$$\begin{aligned}
&\|\mathbf{\Delta}(\mathbf{x}+\delta)\|_2 - \|\mathbf{\Delta}(\mathbf{x})\|_2 = 0\\
\Rightarrow &\text{ Either } \int_{\mathbf{t}} |a-b|^2 \, d\mathbf{t} = \int_{\mathbf{t}} |c-d|^2 \, d\mathbf{t}d\mathbf{t}\\
&\text{or } |a-b|^2 = |c-d|^2, \forall \mathbf{t}\\
&\text{or } a = b \text{ and } c = d, \forall \mathbf{t}
\end{aligned}$$

The condition $\int_{\mathbf{t}} |a-b|^2 \, d\mathbf{t} = \int_{\mathbf{t}} |c-d|^2 \, d\mathbf{t}$ do not assure gradient zero for optimal indicating independence condition.
As per $|a-b|^2 = |c-d|^2, \forall \mathbf{t} \Rightarrow$ two different cases can be thought:

$$\begin{aligned}
\text{Case I:} &a - b - c + d = 0 \Rightarrow a - c = b - d \Rightarrow \xi_{\mathbf{x}}(\mathbf{x}) = \zeta_{\mathbf{x}}(\mathbf{x})\\
\text{Case II: } &a - b + c - d = 0 \Rightarrow a + c = b + d \Rightarrow \text{ spurious optima}
\end{aligned}$$

The Case I imply independence but not the Case II.
The condition $a = b$ and $c = d, \forall \mathbf{t}$ also implies independence.
Over all, the analysis implies that the contrast $\Phi_2^{\mathbf{\Delta}}$ may have gradient zero indicating spurious maxima.
Same way, for other values of p also, existence of spurious optima can be proved.
Also, in a similar way, possible existence of local optima for contrasts $\Phi_p^{\boldsymbol{\alpha}}$ and Φ_p^{A} can be proved.

2.6.2 FD and its Stochastic Gradient

The previous relation of FD, GFD and HFD reminds us the relationship between mutual information and the SFD. As proved by Babaie-Zadeh et al. (9), SFD is the stochastic gradient and can be used to derive differential of mutual information. Also, it has been used to derive that mutual information has no local minima (8). So, it will be desired to investigate whether such results can be obtained with respect to FD, GFD and HFD.

Let us try to obtain differential of FD, in terms of GFD as defined in Section 2.3. Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ be a random vector and $\delta = (\delta_1, \ldots, \delta_n)^T$ be a 'small' random vector. Then, the interest here is in the differential function of FD that is, $\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})$.

$$\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})=\left(\prod_{i=1}^{n}p_{x_i+\delta_i}(x_i+\delta_i)-p_{\mathbf{x}+\delta}(\mathbf{x}+\delta)\right)-\left(\prod_{i=1}^{n}p_{x_i}(x_i)-p_{\mathbf{x}}(\mathbf{x})\right)$$

Assuming $\mathbf{t}$ as the support of all the PDFs,

$$\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})=\left(\prod_{i=1}^{n}p_{x_i+\delta_i}(t_i)-p_{\mathbf{x}+\delta}(\mathbf{t})\right)-\left(\prod_{i=1}^{n}p_{x_i}(t_i)-p_{\mathbf{x}}(\mathbf{t})\right)$$

Using Lemma 1 in (9), the following holds.

$$p_{\mathbf{x}+\delta}(\mathbf{t})-p_{\mathbf{x}}(\mathbf{t})=-\sum_{i=1}^{n}\frac{\partial}{\partial t_i}\{E_{\delta_i}\{\delta_i|\mathbf{x}=\mathbf{t}\}p_{\mathbf{x}}(\mathbf{t})\}+o(\delta) \tag{2.27}$$

$$=-E_{\delta}\{\delta^T\zeta_{\mathbf{x}}(\mathbf{x})\}+o(\delta) \tag{2.28}$$

Same can be applied to the product of the marginal PDFs, itself being a PDF.

$$\prod_{i=1}^{n}p_{x_i+\delta_i}(\mathbf{t})-\prod_{i=1}^{n}p_{x_i}(\mathbf{t})=-\sum_{i=1}^{n}\frac{\partial}{\partial t_i}\{E_{\delta_i}\{\delta_i|\mathbf{x}=\mathbf{t}\}\prod_{i=1}^{n}p_{x_i}(x_i)\}+o(\delta)$$

$$=-E_{\delta}\{\delta^T\boldsymbol{\xi}_{\mathbf{x}}(\mathbf{x})\}+o(\delta) \tag{2.29}$$

Combining Equation (2.28) and Equation (2.29), the differential function of FD can be given by,

$$\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})=-E_{\delta}\{\delta^T\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})\}+o(\delta)$$

This is the differential function and to convert it into a number, let us simply integrate it over $\mathbf{t}$.

$$\Rightarrow\nabla\int_{\mathbf{x}}\boldsymbol{\Delta}_{\mathbf{x}}(\mathbf{x})=\int_{\mathbf{t}}\left(\boldsymbol{\Delta}_{\mathbf{x}+\delta}(\mathbf{t})-\boldsymbol{\Delta}_{\mathbf{x}}(\mathbf{t})\right)d\mathbf{t}=\int_{\mathbf{t}}E_{\delta}\{\delta^T\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})\}d\mathbf{t}+o(\delta)$$

$$\Rightarrow\int_{\mathbf{x}}\left(\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})\right)d\mathbf{x}=\int_{\mathbf{x}}E_{\delta}\{\delta^T\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})\}d\mathbf{x}+o(\delta)=\delta^T\int_{\mathbf{x}}\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})+o(\delta)$$

$$\Rightarrow\nabla\int_{\mathbf{x}}\boldsymbol{\Delta}_{\mathbf{x}}(\mathbf{x})=\lim_{\delta\to0}\frac{\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})}{\delta}=\int_{\mathbf{x}}\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})d\mathbf{x}$$

Similarly, $\Rightarrow E\{\boldsymbol{\Delta}(\mathbf{x}+\delta)-\boldsymbol{\Delta}(\mathbf{x})\}=E\{E_{\delta}\{\delta^T\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})\}\}+o(\delta)=\delta^TE\{\boldsymbol{\alpha}_{\mathbf{x}}(\mathbf{x})\}+o(\delta)$

$$\Rightarrow\nabla E\{\boldsymbol{\Delta}(\mathbf{x})\}=E\{\boldsymbol{\alpha}(\mathbf{x})\}$$

The above result proves that the GFD ($\boldsymbol{\alpha}$) serves as a stochastic gradient of the integrated Function Difference or expectation FD of a random vector. So, it could have been easier prove that $\boldsymbol{\Delta}(\mathbf{x}+$

$\delta) - \mathbf{\Delta}(\mathbf{x}) = 0 \Leftrightarrow \boldsymbol{\alpha}(\mathbf{x}) = 0$ and that implies independence. But, the similar can not be proved for their corresponding l^p measures i.e. $\mathbf{\Delta}_p(\mathbf{x} + \delta) - \mathbf{\Delta}_p(\mathbf{x}) = 0 \not\Leftrightarrow \boldsymbol{\alpha}_p(\mathbf{x}) = 0$ can not be proved. The reason is the contrast defined use the L^p-norm of FD and not just the integration or expectation of FD, as this quantities do not assure nonnegativity. So, the effort to prove that the contrasts are without local minima in the previous Section 2.6.1, actually resulted into the proof for possible existence of spurious local optima for them.

Overall, the contrast $\Phi_p^{\mathbf{\Delta}}(\mathbf{y}(\theta))$, $\Phi_p^{\boldsymbol{\alpha}}(\mathbf{y}(\theta))$ and $\Phi_p^{A}(\mathbf{y}(\theta))$ do not have any local maxima with respect to itself. But, it may still have local maxima as a function of θ (or some other variable), as $\mathbf{y}$ itself is a function of the search parameter θ. The next Section 2.7 focuses on the empirical estimation of these contrasts.

2.7 Preliminary background on Estimation of the Derived Contrasts

Usually, the independence measures avoid estimation of joint PDF, as higher dimension joint PDF estimation is less accurate or requires more samples than marginal PDF estimation. The article (88) notes that the measures based on estimation of joint PDF and marginal PDF both, try to cancel out estimation errors compare to the measures only estimating the marginal entropies. The minimization of L^p-norm of FD, GFD and HFD are the BSS contrasts belong to this class of contrasts. The conventional way is to estimate them following a two stage process. In the first stage, separate estimation of joint PDF and marginal PDFs for $\Phi_p^{\mathbf{\Delta}}$, their gradients for $\phi_p^{\boldsymbol{\alpha}}$ and their Hessians for Φ_p^{A} is achieved. Then, the second stage estimates their difference or L^p-norm. The separate estimation of the PDFs and their derivatives can be achieved through histogram based technique or kernel based method. The histogram based PDF estimation method is fast but less accurate compare to the kernel method. The estimation theory basics says that two stage estimation process for a required quantity amplifies the error in estimation. So, either separate estimation of joint and marginal PDFs and then their difference or the first joint PDF estimation, then based on it the marginal PDFs estimation and then the difference - this both way are indirect estimation method. Compare to them, the direct estimation of the required quantity from the data is supposed to be less erroneous. Though theoretically any real $p \geq 1$ is allowed, either $p = 1$ or $p = 2$ are more suitable for computation. The Kernel theory says that a quantity based on the square of the PDF requires less computations than that based on PDF; if a Gaussian kernel is used.

In general, compare to the estimation of PDFs, their derivatives and Hessians have more inaccuracies or require more samples for same precision. So, the chapter derives only the contrasts based on FD and GFD. In the light of these observations, there is proposed direct estimation of the

L^2 based contrasts using 'least squares' approach. There are two different estimation approaches based on the sample locations selected to place the kernel basis. The first approach is to select the joint sample locations to place the multivariate kernel basis. The corresponding estimator for FD is identified as Φ_2^{LSFD} and that for GFD is identified as Φ_2^{LSGFD}. The methods require $O(b^2)$ computations, where b is the number of basis selected. The another approach places kernel basis at selected paired or un-paired sample locations. It requires $O(b^3)$ computations with better estimations. It is to be noted that the estimation of the same contrasts without the least square based approach requires $O(N^2)$ or $O(N^3)$ order of computations where N is number of samples. Also, using Fast Gauss Transform (FGT) and Incomplete Cholskey Factorization the computational complexity can be further reduced. Similar methods are already in use for direct estimation of density difference (123), density ratio (121, 142) and squared loss mutual information (106, 120, 122). The information potential due to such an arrangement of basis functions is identified as the Reference Information Potential (RIP). The chapter extends Information field theory to incorporate the new concepts of Reference Information Potential (RIP) and Cross-RIP (CRIP). The concepts are demonstrated, through above four estimators, to be useful to derive closed form expressions for information field analysis.

2.7.1 Kernel Basics and Information Potential

Given N realizations of an unknown PDF $f(x)$, the kernel density estimate $f(\hat{x})$ is given by

$$f(\hat{x}) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{h}K\left(\frac{x-x_i}{h}\right) \tag{2.30}$$

where, $K(u)$ is the kernel function and h is the bandwidth parameter deciding the spread of the kernel. Usually, $K(u)$ is a symmetric, positive definite and bounded function, i.e. it satisfies the following properties:

$$K(u) \geq 0,\ \int_{-\infty}^{\infty} K(u)du = 1,\ \int_{-\infty}^{\infty} uK(u)Du = 0,\ \int_{-\infty}^{\infty} u^2K(u)du = \mu_2(K) < \infty \tag{2.31}$$

It is known that the convolution (symbol '$*$') of two Gaussian functions is still a Gaussian function($G(\cdot,\cdot)$). In a single dimension,

$$G(\mathbf{v},\sigma_1) * G(\mathbf{u}-\mathbf{v},\sigma_2) = G(\mathbf{u},\sqrt{\sigma_1^2+\sigma_2^2}) \tag{2.32}$$

Let us use this property to estimate the expectation of the square of PDF. Let the Gaussian kernel be given as,

$$G_\sigma(x - m_x) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{1}{2}\left(\frac{x-m_x}{\sigma}\right)^2} dx \tag{2.33}$$

Then,

$$\int \{\hat{f(x)}^2\} dx = \int_{-\infty}^{\infty} \left(\frac{1}{N} \sum_{i=1}^{N} G_\sigma \left(x - x_i\right) \right)^2 dx \tag{2.34}$$

$$= \int_{-\infty}^{\infty} \frac{1}{N^2} \sum_{j=1}^{N} \sum_{i=1}^{N} G_\sigma(x - x_i) G_\sigma(x - x_j) dx \tag{2.35}$$

$$= \frac{1}{N^2} \sum_{j=1}^{N} \sum_{i=1}^{N} \int_{-\infty}^{\infty} G_\sigma(x - x_i) G_\sigma(x - x_j) dx \tag{2.36}$$

$$= \frac{1}{N^2} \sum_{j=1}^{N} \sum_{i=1}^{N} G_{\sigma\sqrt{2}}(x_i - x_j) \tag{2.37}$$

Thus, the integration of the square of PDF is achieved in a computationally efficient way, avoiding the continuous integration. The ITL theory has given significance to this property by identifying it as a quadratic information potential. The details on IP, related independence measure QMI_{ED} and information forces follow in the Appendix B.

2.8 Extention to IP Theory

One of the interpretations describes potential as the amount of work done required to bring a unit charge (for electric field) or unit mass (for gravity field) from infinity to the point in the force field, where infinity implies a point with zero potential. The particle contains amount of potential energy that has been applied to work against the force. It is customary in potential theory to think of a reference potential i.e. assuming that a particle is moved from a reference point in the field instead from the infinity. This helps analyzing a potential or gravitational field through a reference framework instead absolute. In a gravitational field theory, the potential energy at a hight from a sea level reference or some other local reference; in an electric field theory potential difference with respect to the common/neutral of the system or earth - are the respective examples. Moreover, during field analysis it is a general practice to start with a reference potential and then to express the required related quantities as a function of this reference potential. For example, in nodal analysis for electrical circuit analysis, reference potential is assumed at every node.

Once defined IP, the natural query is whether it is possible to derive the concept of reference potential for information fields? Further, whether there can be derived some laws for information field analysis using the reference IP concept? The first question is answered defining RIP and related quantities in Section 2.8.1, 2.8.2 and 2.8.3 . The second question is answered in the Section 2.8.4.

2.8.1 Reference Information Potential (RIP)

In kernel analysis, it is customary to initially assume a set of kernel basis placed at some selected sample points and then the required quantities are expressed as a function of the basis. The potential due to kernel basis can be identified as a Reference Information Potential (RIP). Analogous with the laws in electric circuit analysis, the least squares like approaches can be thought to bring functional relationship between a required quantity and the reference potential.

Let $\Psi(x) = \{\psi(x_1), \psi(x_2), \ldots, \psi(x_b)\}$ be the set of kernel functions consecutively placed at selected sample locations[†] $x_i, i = 1, 2, \ldots, b$. They act as basis as potential at any point in the field is measured using linear combinations of them. . The selected sample points can be seen as the occurrences of a random variable X_ψ. Then, the potential of X_ψ ($\hat{V}_\alpha(X_\psi)$) is a Reference Information Potential (RIP). More specifically, quadratic RIP is the integral of the square of the PDF of X_ψ ($p_{x_\psi}(x)$), as under:

$$\begin{aligned}
\text{RIP}_2 = V_R^2 &\overset{def}{=} \int_x p_{X_\psi}^2(x)dx \\
\hat{V}_R^2 &= \int_x \hat{p}_{X_\psi}^2(x)dx \\
&= \int_x \left(\frac{1}{b} \sum_{i=1}^{b} \psi_\sigma\left(x - x_i\right) \right)^2 dx \\
&= \int_x \frac{1}{b^2} \sum_{j=1}^{b} \sum_{i=1}^{b} \psi_\sigma(x - x_i)\psi_\sigma(x - x_j)dx \\
&= \frac{1}{b^2} \sum_{j=1}^{b} \sum_{i=1}^{b} \int_x \psi_\sigma(x - x_i)\psi_\sigma(x - x_j)dx \\
&= \hat{V}_2(X_\psi)
\end{aligned}$$

[†] usually, the basis are placed at sample points but can be placed at some other points in the field

For a Gaussian kernel, $\psi(x_i) = G(x_i)$, the following holds:

$$\begin{aligned}\hat{V}_R &= \frac{1}{b^2}\sum_{j=1}^{b}\sum_{i=1}^{b}\int_x G_\sigma(x-x_i)G_\sigma(x-x_j)dx \\ &= \frac{1}{b^2}\sum_{j=1}^{b}\sum_{i=1}^{b} G_{\sigma\sqrt{2}}(x_i-x_j)\end{aligned}$$

The quadratic RIP definition can be generalized to α RIP, as:

$$RIP_\alpha = \mathbf{V}_R^\alpha \stackrel{def}{=} \int_x (p_{\hat{X}_\psi}(x))^\alpha dx$$

Once defined RIP, two more related concepts can be defined to bring the closed form expression for information field analysis.

2.8.2 Cross Reference Information Potential (CRIP)

The Cross Information Potential (CIP) is defined as the entropy of a PDF $f(x)$ with respect to an another PDF $g(x)$: CIP $= E\{f(x)\} = \int f(x)g(x)dx$. With reference to the newly defined RIP concept, entropy of a PDF $f(x)$ with respect to the reference PDF $\hat{p}_{X_\psi}$ is called CRIP. The CRIP estimates the potential on the selected locations as the basis due to the interactions of locations from the sample space of $f(x)$ (or vice versa).

$$\begin{aligned}\text{CRIP}_2 = \mathcal{V}_R^2(f) \stackrel{def}{=} \mathcal{V}_2(f, X_\psi) &= \int f(x)p_{X_\psi}dx \\ \hat{\mathcal{V}}_2(f, X_\psi) &= \int_x \frac{1}{N}\sum_{i=1}^{N}\psi_\sigma(x-x_f(i))\frac{1}{b}\sum_{j=1}^{b}\psi_\sigma(x-x(j))dx \\ &= \frac{1}{Nb}\sum_{j=1}^{b}\sum_{i=1}^{N}\int_x \psi_\sigma(x-x_f(i))\psi_\sigma(x-x(j))dx\end{aligned}$$

For a Gaussian kernel, $\psi(x-x_i) = G(x-x_i)$, then:

$$\begin{aligned}\hat{\mathcal{V}}_R &= \frac{1}{Nb}\sum_{j=1}^{b}\sum_{i=1}^{N}\int_x G_\sigma(x-x_f(i))G_\sigma(x-x(j))dx \\ &= \frac{1}{Nb}\sum_{j=1}^{b}\sum_{i=1}^{b} G_{\sigma\sqrt{2}}(x_f(i)-x(j))\end{aligned}$$

2.8.3 Information Interaction Matrix (IIM)

The analysis may not just require the final scalar outcome, but may depend upon the intermediate information interactions. So, let there be defined an Information Interaction Matrix (IIM) as the matrix due to each interaction. There can be IIM for potential, IIM for reference potential and IIM for information forces etc. For example, the field with N sample points will have N^2 interactions that will be the size of the IIM for potential. Similarly, the IIM for reference potential will be of dimension $b \times b$ and IIM for CRIP will be of dimension $N \times b$. This is analogous to the Gram Matrix. Let us symbolize $(V_\alpha(x_i, x_j))$ as the potential on x_j due to interaction with x_i and $\mathbf{V}_\alpha(X)$ as the IIM for the potential of random variable X. Also, $V_\alpha(X)$ is already symbolized as the scalar quantity IP of X. Accordingly, V_R^α is the reference potential and $\mathbf{V}_R$ is the Reference potential IIM. In short, $V(x(i), x(j)) = \int \psi(x, x(i))\psi(x, x(j))dx$, $[\mathbf{V}_2(X)]_{ij} = V(x(i), x(j))$ and $V_2(X) = \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{j}[\mathbf{V}_2(X)]_{ij}$. Similarly, $\mathcal{V}(x_f(i), x(j)) = \int_x \psi_\sigma(x - x_f(i))\psi_\sigma(x - x(j))dx$, $[\mathcal{V}_R^2]_{ij} = \mathcal{V}(x_f(i), x(j))$ and $\mathcal{V}_R^2(X) = \frac{1}{Nb}\sum_{j=1}^{b}\sum_{i=1}^{b}[\mathcal{V}_R^2]_{ij}$.

2.8.4 Closed-Form Expression using Reference Potential through Least Squares

Conventionally, the concept of reference potential is used in both electric field and gravitation field analysis. Analogous to them, this section demonstrates that existing 'least squares' like principles can be used to derive closed-form expressions; in terms of the RIP and related concepts; for information field analysis. A mathematical expression is of type closed form if it requires finite number of constants, variables and operations.

The method of 'least squares' aims at estimating the model parameters that minimize the sum of squared errors between the true and the estimated quantity. Without loss of generality, let us use assume that $f(x)$ is the quantity to be estimated and $g(x) = \hat{f}(x)$ is the estimation. Then:

$$\begin{aligned}\text{lsf} &= \int_{\mathbb{R}} (g(x) - f(x))^2 dx \\ &= \int_{x\in\mathbb{R}} g(x)^2 dx - 2\int_{x\in\mathbb{R}} g(x)f(x)dx + \int_{x\in\mathbb{R}} f(x)^2 dx \\ &= V_2(g(x)) - 2\mathcal{V}(g(x), f(x)) \ (\because \text{ the last term has no effect on optimal lsf}) \end{aligned} \tag{2.38}$$

where, $V_2(g(x)) = ||\hat{f}(x)||_2$ is the potential of the estimated $f(x)$ and $\mathcal{V}(g(x), f(x)) = ||\hat{f}(x)f(x)||$ is the cross information potential between the actual and estimated $f(x)$. So, both quantities represent the estimation of $\int_{\mathbb{R}} f(x)^2 dx$. But, as proved by Sugiyama et al. (123), the linear combination of them, the lsf, is more bias corrected estimator of $\int_{\mathbb{R}} f(x)^2 dx$.

Let us assume further that $g(x)$ is given by a linear combination of the selected basis functions placed at the selected sample points.

$$g(x) = \sum_{i=1}^{b} \theta_i \psi_i(x) = \boldsymbol{\theta}(x)^T \Psi(x) \tag{2.39}$$

where, b denotes the number of basis functions; $\boldsymbol{\theta}(x) = (\theta_1, \theta_b, ..., \theta_b)^T$ is the parameter vector and $\Psi(x) = (\psi_1, \psi_2, ..., \psi_b)^T$ is the basis function vector. So, with regularization function $R(\boldsymbol{\theta}) = \boldsymbol{\theta}^T\boldsymbol{\theta}$ and λ as the regularization parameter,

$$\text{lsf}(\boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{V}_R \boldsymbol{\theta} - 2\mathcal{V}_R^T \boldsymbol{\theta} + \lambda \boldsymbol{\theta}^T \boldsymbol{\theta} \tag{2.40}$$

$$\text{where, } \mathbf{V}_{R(b\times b)} = \int_{\mathbb{R}} \Psi(x)\Psi^T(x)dx \tag{2.41}$$

$$\mathcal{V}_{R(b\times 1)} := \mathcal{V}_{b\times 1}((\boldsymbol{\psi}(x), f(x))) = \int_{\mathbb{R}} \Psi(x) f(x) dx \tag{2.42}$$

The estimator depends upon the IIM for RIP ($\mathbf{V}_R$) and the IIM for CRIP ($\mathcal{V}_R$) of $f(x)$. The optimal value of parameter vector $\boldsymbol{\theta(x)}$ can be obtained by minimizing the gradient of lsf.

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\text{argmin}} \text{ lsf}(\boldsymbol{\theta}) \tag{2.43}$$

$$\frac{\partial}{\partial \boldsymbol{\theta}} \text{lsf} = \mathbf{V}_R \boldsymbol{\theta} + \lambda \boldsymbol{\theta} - \mathcal{V}_R \tag{2.44}$$

$$\boldsymbol{\theta}^* = (\mathbf{V}_R + \lambda \mathbf{I}_b)^{-1} \mathcal{V}_R \tag{2.45}$$

where, $\mathbf{I}_b$ is a b-dimensional identity matrix. Thus, obtaining IIMs $\mathbf{V}_R$ and $\mathcal{V}_R(\Psi(x), f(x))$ gives the parameter vector ($\boldsymbol{\theta}$) and that, in turn, gives least squares estimator lsf. Overall, the Equation (2.40), with Equation (2.45), imply closed-form equation is available for the estimation of $f(x)$ or $||f(x)||_2$. This also justifies the purpose to define the quantities RIP, CRIP and IIMs.

2.9 Φ_2^{FD} Estimation

The section targets estimation of the contrast Φ_2^{FD} using closed form expressions, in terms of the RIP and related concepts, derived in the previous Section 2.8.4. Instead of the conventional two stage approach, here, the estimation is achieved directly in a single stage through 'least squares'. Similar methods are already in use for direct estimation of density difference (123), density ratio (121, 142) and squared loss mutual information (106, 120, 122). Also, it is worth noting that $\Phi_2^{\text{FD}} := -||\boldsymbol{\Delta}(\cdot)|_2$ and $QIM_{ED} = ||\boldsymbol{\Delta}(\cdot)|_2$; where, QIM_{ED} denotes the Euclidean Distance based Quadratic Independence Measure defined by Principe (96). Overall, the section also estimates QIM_{ED} directly, in a single stage.

Without loss of generality, let us estimate $\mathbf{\Delta}(\cdot)$ of a two dimensional random vector and then the results be generalized to higher dimensions. The $\mathbf{\Delta}(\cdot)$ of a two-dimensional random vector is:

$$\mathbf{\Delta}(x, y) := p_{xy}(x, y) - p_x(x)p_y(y) \tag{2.46}$$

Let $g(x, y) = \hat{\mathbf{\Delta}}(x, y)$ be the estimated $\mathbf{\Delta}(x, y)$ and LSFD be the Least Squares based FD estimator. Then:

$$\begin{aligned} \text{LSFD} &= \int_{\mathbb{R}^2}\int (g(x, y) - \mathbf{\Delta}(x, y))^2 dxdy && (2.47)\\ &= V_2(g(x, y)) - 2\mathcal{V}(g(x, y), \mathbf{\Delta}(x, y)) \;(\because \text{ using Equation (2.38)}) && (2.48)\end{aligned}$$

where, $V_2(g(x, y)) = ||\hat{\mathbf{\Delta}}||_2$ is the potential of the estimated $\mathbf{\Delta}(x, y)$ and $\mathcal{V}(g(x, y), \mathbf{\Delta}(x, y)) = ||\hat{\mathbf{\Delta}}\mathbf{\Delta}||$ is the cross information potential between the actual and estimated $\mathbf{\Delta}(x, y)$.
Let us assume further that $g(x, y)$ is given by a linear combination of the selected basis functions placed at the selected sample points.

$$g(x, y) = \sum_{i=1}^{b} \theta_i \psi_i(x, y) = \boldsymbol{\theta}(x, y)^T \Psi(x, y) \tag{2.49}$$

where, b denotes the number of basis functions; $\boldsymbol{\theta}(x, y) = (\theta_1, \theta_b, ..., \theta_b)^T$ is the parameter vector and $\Psi(x, y) = (\psi_1, \psi_2, ..., \psi_b)^T$ is the basis function vector. So, with regularization function $R(\boldsymbol{\theta}) = \boldsymbol{\theta}^T\boldsymbol{\theta}$ and λ as the regularization parameter,

$$\begin{aligned} \text{LSFD}(\boldsymbol{\theta}) &= \boldsymbol{\theta}^T \mathbf{V}_R \boldsymbol{\theta} - 2\mathcal{V}_R^T\boldsymbol{\theta} + \lambda\boldsymbol{\theta}^T\boldsymbol{\theta} && (2.50)\\ \text{where, } \mathbf{V}_{R(b\times b)} &= \int_{\mathbb{R}}\int_{\mathbb{R}} \Psi(x, y)\Psi^T(x, y)dxdy && (2.51)\\ \mathcal{V}_{R(b\times 1)} := \mathcal{V}_{b\times 1}((\psi(\mathbf{x}, \mathbf{y}), \mathbf{\Delta})) &= \int_{\mathbb{R}}\int_{\mathbb{R}} \Psi(x, y)(p_{xy}(x, y) - p_x(x)p_y(y))dxdy && (2.52)\\ [\mathcal{V}_R]_1 := \mathcal{V}_R(\psi(x_l, y_l), \mathbf{\Delta}) &= \sum_{i=1}^{N} [\mathcal{V}_R(\Psi(x_l, y_l), \mathbf{\Delta})']_{li} && (2.53)\end{aligned}$$

The estimator depends upon the IIM for RIP and the IIM for CRIP of $\mathbf{\Delta}(x, y)$. The optimal value of parameter vector $\boldsymbol{\theta(x, y)}$ can be using the following equation.

$$\theta^* = (\mathbf{V}_R + \lambda\mathbf{I}_b)^{-1}\mathcal{V}_R \tag{2.54}$$

where, $\mathbf{I}_b$ is a b-dimensional identity matrix. Thus, obtaining IIMs $\mathbf{V}_R$ and $\mathcal{V}_R(\Psi(x_l, y_l), \mathbf{\Delta})$ gives the parameter vector (θ). Finally, the least squares estimator LSFD gives the bias corrected estimation of Φ_2^{FD}.

2.9.1 Φ_2^{FD} Estimation through Multiplicative Kernel Model

Let us use multiplicative Gaussian kernel function as a basis function placed at the selected sample points. So,

$$g(x,y) = \sum_{i=1}^{b} \theta_i K(x,x_i) L(y,y_i) = \boldsymbol{\theta}^T [\mathbf{k}(x) \text{ o } \mathbf{l}(y)] \tag{2.55}$$

where, $K(x,x_i)$ and $L(y,y_i)$ are the kernel functions at x_i and y_i consecutively;
$k(x) = (K(x,x_1), K(x,x_2), ..., K(x,x_b))^T$ and $l(y) = (L(y,y_1), L(y,y_2), ..., L(y,y_b))^T$ are the kernel vectors and the operator o denotes *Hadamard product*. This gives

$$\begin{aligned} [\mathbf{V}_R(x,y)]_{ij} &= \int_{\mathbb{R}} \int_{\mathbb{R}} K(x,x_i) L(y,y_i) K(x,x_j) L(y,y_j) dxdy \\ \Rightarrow \hat{\mathbf{V}}_{R(b\times b)}(x,y) &= \mathbf{V}_R(x) \text{ o } \mathbf{V}_R(y) \\ \text{or } [\hat{\mathbf{V}}_R(x,y)]_{ij} &= \left(\frac{1}{\sqrt{\pi} 2\sigma}\right)^2 \exp\left(-\frac{(x_i - x_j)^2}{4\sigma^2} - \frac{(y_i - y_j)^2}{4\sigma^2}\right) \end{aligned}$$

where, $\mathbf{V}_R(x)$ is a $b \times b$ matrix with entries $[\mathbf{V}_R(x)]_{ij} = K(x,x_i) * K(x,x_j)$ and $\mathbf{V}_R(y)$ is a $b \times b$ matrix with entries $[\mathbf{V}_R(y)]_{ij} = L(y,y_i) * L(y,y_j)$ and $*$ is the symbol for convolution operation.

The IIM for RIP $\mathbf{V}_{R(b\times b)}$, for an n-dimensional quantity, is obtained using b^n multiplications. The computations can be reduced by replacing multiplications through additions of the exponents. This will require square of the nb^2 terms, $\frac{nb^2}{2}$ additions of exponents and then taking exponents of b^2 terms. Now, the sample estimate of $\mathcal{V}_R$ ($\hat{\mathcal{V}_R}$) can be obtained as under:

$$\mathcal{V}_{R(b\times 1)}(\Psi(x,y),\boldsymbol{\Delta}(x,y)) = \int_{\mathbb{R}}\int_{\mathbb{R}} (\mathbf{k}(x) \text{ o } \mathbf{l}(y))(p_{xy}(x,y) - p_x(x)p_y(y))dxdy \tag{2.56}$$

$$[\hat{\mathcal{V}}_R(\Psi(x,y),\boldsymbol{\Delta}(x,y))]_l = \mathcal{V}(\psi(x_l,y_l),p_{xy}(x,y)) - \mathcal{V}(\psi(x_l,y_l),p_x(x)p_y(y)) \tag{2.57}$$

$$\mathcal{V}(\psi(x_l,y_l),p_{xy}(x,y)) = \int_{\mathbb{R}}\int_{\mathbb{R}} (K(x,x_l)L(y,y_l)) \left(\frac{1}{N}\sum_{i=1}^{N}(K(x,x_i)L(y,y_i))\right) dxdy \tag{2.58}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\mathcal{V}_R(x_i,x_l)\mathcal{V}_R(y_i,y_l) \tag{2.59}$$

$$= \frac{1}{4\pi\sigma^2 N}\sum_{i=1}^{N}\exp\left\{-\frac{(x_i-x_l)^2+(y_i-y_l)^2}{4\sigma^2}\right\} \tag{2.60}$$

$$\mathcal{V}(\psi(x_l,y_l),p_x(x)p_y(y)) = \int_{\mathbb{R}}\int_{\mathbb{R}} (K(x,x_l)L(y,y_l)) \left(\frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N}(K(x,x_i)L(y,y_j))\right) dxdy \tag{2.61}$$

$$= \frac{1}{N^2}\sum_{i=1}^{N}\left(\mathcal{V}_R(x_i,x_l)\left(\sum_{j=1}^{N}\mathcal{V}_R(y_j,y_l)\right)\right) \tag{2.62}$$

$$= \frac{1}{N^2}\left(\sum_{i=1}^{N}\mathcal{V}_R(x_i,x_l)\right)\left(\sum_{j=1}^{N}\mathcal{V}_R(y_j,y_l)\right) \tag{2.63}$$

The estimation of $\mathcal{V}(\psi(x_l,y_l),p_{xy}(x,y))$ is obtained by replacing the Hadamard product through addition of the exponents, as for the estimation of $\mathbf{V}_R(x,y)$. Each entry requires $n-1$ additions and N exponents followed by N additions. Each entry $\mathcal{V}(\psi(x_l,y_l),p_x(x)p_y(y))$ is obtained through nN additions and $(n-1)$ multiplication. So, over all vector $\hat{\mathcal{V}}_R(\Psi(x,y),p_{xy}(x,y))$ requires $(N+n-1)b$ additions and Nb number of exponents. The estimation of vector $\hat{\mathcal{V}}_R(\Psi(x,y),p_x(x)p_y(y))$ requires nNb additions and $b(n-1)$ multiplications.

Once $\mathbf{V}_R(x,y)$ and $\mathcal{V}_R$ are available, the linear coefficients ($\boldsymbol{\theta}$) can be obtained solving Equation (2.54). The time complexity for this is $O(b^2)$. Based on Equation 2.48, the required $\Phi_2^{\boldsymbol{\Delta}} = -$LSFD, is estimated. Also, the method estimates both the Function Difference ($\boldsymbol{\Delta}$) and Φ_2^{FD} of a random vector simultaneously. The time complexity is usually measured in terms of the number of multiplications. With this, the total multiplication time complexity is only $O(b^2 + b(N+n-1))$. It can be further reduced by taking exponent of values corresponding to $(x_i - x_j)^2 < (3\sigma)^2$ or $(y_i - y_j)^2 < (3\sigma)^2$ as zero. Though not the time complexity, the performance directly depends upon the number of samples available; specifically in higher dimensions. To effectively increase the available samples for estimation, the next section uses basis placed at both paired and unpaired samples to estimate $\hat{\mathbf{\Delta}}$. The estimator is identified as LSFD2.

2.9.2 LSFD2 Estimation through Multiplicative Kernel Basis Placed at Paired and Un-paired Samples

The estimation method places the multiplicative kernels as basis at unpaired samples also. This allows the use of *Kronecker* structure to reduce the computational cost. The approximation $g(x, y)$ is defined as:

$$\begin{aligned} g(x,y) &= \sum_{j=1}^{b}\sum_{i=1}^{b}\theta_{ij}K(x,x_i)L(y,y_j) \\ &= \mathrm{vec}(\mathbf{\Theta})^T[(\mathbf{I}_b \otimes \mathbf{k}(x)) \text{ o } (\mathbf{l}(y) \otimes \mathbf{I}_b)] \end{aligned}$$

where, $\mathbf{\Theta}$ is a $b \times b$ parameter matrix, vec$(\cdot)$ is a vectorization function and $\otimes$ implies the *Kronecker* product. Accordingly,

$$\begin{aligned} [\mathbf{V}_R]_{(i\cdot(b-1)+j,k\cdot(b-1)+l)} &= \int\int K(x,x_i)L(y,y_j)K(x,x_k)L(y,y_l)dxdy \\ \Rightarrow \mathbf{V}_{R(b^n\times b^n)}(x,y) &= \mathbf{V}_R(y) \otimes \mathbf{V}_R(x) \end{aligned} \tag{2.64}$$

where, $\mathbf{V}_R(x)$ is a $b \times b$ matrix with entries $[\mathbf{V}_R(x)]_{ij} = K(x, x_i) * K(x, x_j)$ and $\mathbf{V}_R(y)$ is a $b \times b$ matrix with entries $[\mathbf{V}_R(y)]_{ij} = L(y, y_i) * L(y, y_j)$.

The sample estimate of $\mathcal{V}_R$ ($\hat{\mathcal{V}}_R$) can be obtained as under:

$$\mathcal{V}_{R(b^n\times 1)}(\mathbf{\Psi}(x,y),\mathbf{\Delta}) = \int_{\mathbb{R}}\int_{\mathbb{R}}[(\mathbf{I}_b \otimes \mathbf{k}(x)) \text{ o } (\mathbf{l}(y) \otimes \mathbf{I}_b)](p_{xy}(x,y) - p_x(x)p_y(y))dxdy \tag{2.65}$$

$$\hat{\mathcal{V}}_{R(l\cdot(b-1)+l')}(\mathbf{\Psi}(x,y),\mathbf{\Delta}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{V}(x_i,x_l)\mathcal{V}(y_i,y_l') - \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N}\mathcal{V}(x_i,x_l)\mathcal{V}(y_j,y_l') \tag{2.66}$$

$$\Rightarrow \hat{\mathcal{V}}_R(\mathbf{\Psi}(x,y),\mathbf{\Delta}) = \mathcal{V}_R(\mathbf{\Psi}(x,y),p_{xy}(x,y)) - \mathcal{V}_R(\mathbf{\Psi}(x,y),p_x(x)p_y(y)) \tag{2.67}$$

where,

$$\mathcal{V}_R(\mathbf{\Psi}(x,y),p_{xy}(x,y)) = \begin{cases} \frac{1}{N}\mathrm{vec}\left(\mathcal{V}_R^T(x)\mathcal{V}_R(y)\right), & \text{for } n=2 \\ [\mathcal{V}_R(p_{xy}(x,y))]_{l+(l'-1)\cdot b+(l''-1)\cdot b^2} & \text{for } n=3 \end{cases}$$

$$\text{where, } [\mathcal{V}_R(p_{xy}(x,y))]_{l+(l'-1)\cdot b+(l''-1)\cdot b^2} = \frac{1}{N^n}\sum_{i=1}^{N}\mathcal{V}(x_i,x_l)\mathcal{V}(y_i,y_l')\mathcal{V}(z_i,z_l'') \tag{2.68}$$

$$\mathcal{V}_R(\boldsymbol{\Psi}(x,y), p_x(x)p_y(y)) = \begin{cases} [\mathcal{V}_R]_{l+(l'-1)\cdot b} = \frac{1}{N^2}\left(\sum_{i=1}^{N} \mathcal{V}(x_i, x_l)\right)\left(\sum_{j=1}^{N} \mathcal{V}(y_j, y'_l)\right) & \text{for } n = 2 \\ [\mathcal{V}_R(p_x(x)p_y(y))]_{l+(l'-1)\cdot b+(l''-1)\cdot b^2} & \text{for } n = 3 \end{cases}$$

$$\text{where, } [\mathcal{V}_R(p_x(x)p_y(y))]_{l+(l'-1)\cdot b+(l''-1)\cdot b^2} = \frac{1}{N^3}\sum_{i=1}^{N} \mathcal{V}(x_i, x_l)\sum_{j=1}^{N} \mathcal{V}(y_j, y'_l)\sum_{k=1}^{N} \mathcal{V}(z_k, z''_l) \quad (2.69)$$

The equation of $\mathcal{V}_R(p_{xy}(x,y))$ for $n = 2$ is not extensible as it is to $n > 2$. The equations for $n = 3$ show the way to get generalization for higher dimensions. As explained previously, the estimation of vector $\mathcal{V}_R(p_{xy}(x,y))$ requires $(N + n - 1)b^n$ additions and Nb^n number of exponents. The estimation of vector $\mathcal{V}_R(p_x(x)p_y(y))$ requires b^2N additions and $n(b^n)$ multiplications.

To estimate $\Phi_2^{\boldsymbol{\Delta}}$, the optimal parameter matrix $\boldsymbol{\Theta}$ is needed. The Equation (2.54) can be written as:

$$\mathbf{V}_R\text{vec}(\boldsymbol{\Theta}) + \lambda\text{vec}(\boldsymbol{\Theta}) = \mathcal{V}_R$$

This is the famous *discrete Sylvester equation* and requires $O(b^3)$ computations to solve it.

Now, the Equation (2.40) can be given as under:

$$\begin{aligned} \text{LSFD2} &= \text{vec}(\boldsymbol{\Theta})^T(\mathbf{V}_R(y) \otimes \mathbf{V}_R(x))\text{vec}(\boldsymbol{\Theta}) - 2\mathcal{V}_R^T\text{vec}(\boldsymbol{\Theta}) && (2.70) \\ &= \text{vec}(\boldsymbol{\Theta})^T\text{vec}(\mathbf{V}_R(x)\boldsymbol{\Theta}\mathbf{V}_R(y)^T) - 2\mathcal{V}_R^T\text{vec}(\boldsymbol{\Theta}) && (2.71) \\ &= \text{trace}(\boldsymbol{\Theta}^T\mathbf{V}_R(x)\boldsymbol{\Theta}\mathbf{V}_R(y)^T) - 2\mathcal{V}_R^T\text{vec}(\boldsymbol{\Theta}) && (2.72) \end{aligned}$$

The computational complexity of above direct estimation of least square error in FD estimation is $O(b^3)$. Overall, the multiplicative kernel used as basis at sampled and unsampled pairs has $O(b^3 + (N + n)b^n)$ computations but expected to be more accurate specifically, with higher dimensions and less number of samples.

2.9.3 A Note on the CRIP Estimations in above Derivations

The least squares method has been used for direct estimation of density difference and density ratio, as it is mention in the Section 2.9. One could have noted that the $\mathcal{V}_R$ has been calculated in this chapter in a different way compare to elsewhere. For example, calculation for $[\mathcal{V}_R]_l =$

$\int_{\mathbb{R}} \psi(x, x_l)p_x(x)dx$ in both the ways is demonstrated here. This chapter simplifies $[\mathcal{V}_R]_l$ as under:

$$\begin{aligned}[\mathcal{V}_R]_l &= \int_{\mathbb{R}} G_\sigma(x, x_l)p_x(x)dx \\ &= \int_{\mathbb{R}} G_\sigma(x, x_l)\left(\frac{1}{N}\sum_{i=1}^{N} G_\sigma(x, x_i)dx\right) \\ &= \frac{1}{N}\sum_{i=1}^{N} G_{\sqrt{2}\sigma}(x_i, x_l)\end{aligned}$$

Intuitively, the kernel interacts with each sample of the PDF $p_x(x)$. The interaction is a convolution resulting into Gaussian with parameter $\sqrt{2}\sigma$.

The other references simplify $[\mathcal{V}_R]_l$ as under:

$$\begin{aligned}[\mathcal{V}_R]_l = \int G_\sigma(x, x_l)p_x(x)dx = \int G_\sigma(x, x_l)\left(\frac{1}{N}\sum_{i=1}^{N} \delta(x, x_i)dx\right) \\ = \frac{1}{N}\sum_{j=1}^{N} G_\sigma(x_l - x_i)\end{aligned}$$

This is also correct, as $[\mathcal{V}_R]_l$ can be thought as a convolution of the actual PDF with direct delta.

Overall, both the approaches are correct. But, the first approach is more precise, as better approximates the PDF $p_x(x)$ through Gaussian kernel, than the delta kernel in the second approach. The empirical results, not reported here, also justify this approach. The Cross IP estimation in (96) follows the first approach, though conventionally the second approach is more popular.

2.10 Φ_2^{GFD} Estimation

To remind; $\Phi_2^{GFD} := -||\alpha(\cdot)||_2$. The least squares approximation of either $\alpha(\cdot) = \nabla(\boldsymbol{\Delta}(\cdot))$ or directly $||\alpha(\cdot)||_2$ and either $A(\cdot) = \nabla^2(\boldsymbol{\Delta}(\cdot))$ or directly $||A(\cdot)||_2$ can be achieved in the same way as that for $\boldsymbol{\Delta}$ in the previous Section 2.9. In general, $\boldsymbol{\Delta}^{(r)}(\cdot)$ is the r^{th} order derivative of $\boldsymbol{\Delta}$ and is a multi-linear vector function. Say, for $r = 1$, it is a vector function and for $r = 2$ it is a matrix of functions.

Without loss of generality, the estimator can be derived for two dimensions. Then, the components can be estimated using the linear approximation $g(x, y) := \hat{\boldsymbol{\Delta}}^{(r)}(x, y) = \boldsymbol{\theta}(x, y)^T\Psi^{(r)}(x, y)$. It is a customary approach to use multiplicative kernels for multivariate density estimation and then multiplicative derivative kernels for the derivative of multivariate density estimation. Accordingly, all major equations for $\boldsymbol{\Delta}^{(r)}(x, y)$ estimation remain as they are in $\boldsymbol{\Delta}(x, y)$ estimation, with simply the basis $\Psi(x,y)$ replaced by $\Psi^{(r)}(x,y)$. For example, with Gaussian kernel $\Psi(x,y) = G_h(x)oG_h(y)$, we get $\Psi^{(r)}(x,y) = \nabla^{(r)}(G_h(x)oG_h(y))$. More specifically, for $r = 1$ and multiplicative Gaussian kernel $\Psi^{(1)}(x,y) = \nabla^{(1)}(G_h(x)oG_h(y)) = [h^{-1}H_1(x)(G(x)oG(y)) \quad h^{-1}H_1(y)(G(x)oG(y)$

Let the least squares estimator for GFD be called LSGFD (Least Square GFD) and be derived as under. Let $g(x,y) := \hat{\alpha}(x,y) = [\frac{\hat{\partial}}{\partial x}\boldsymbol{\Delta}(x,y) \quad \frac{\hat{\partial}}{\partial y}\boldsymbol{\Delta}(x,y)]^T = [g_x(x,y) \quad g_y(x,y)]^T$. and $\alpha(x,y) = [\alpha_x(x,y) \quad \alpha_y(x,y)]^T$. Then,

$$\text{LSGFD} = \int_{\mathbb{R}}\int_{\mathbb{R}} (g_x(x,y) - \alpha_x(x,y))^2 + (g_y(x,y) - \alpha_y(x,y))^2 dxdy \tag{2.73}$$

$$= (V_2(g_x(x,y)) + V_2(g_y(x,y))) - 2\left(\mathcal{V}(g_x(x,y), \alpha_x(x,y)) + \mathcal{V}(g_y(x,y), \alpha_y(x,y))\right) \tag{2.74}$$

$$= V_2(g(x,y)) - 2\mathcal{V}(g(x,y), \alpha(x,y)) \tag{2.75}$$

where, $V_2(g(x,y)) = V_2(g_x(x,y)) + V_2(g_y(x,y))$ is the $\|\hat{\alpha}\|_2 = \|\hat{\alpha}_x\|_2 + \|\hat{\alpha}_y\|_2$ and $\mathcal{V}(g(x,y), \alpha_x(x,y)) = \mathcal{V}(g_x(x,y), \alpha_x(x,y)) + \mathcal{V}(g_y(x,y), \alpha_y(x,y))$ is $\|\hat{\alpha}\alpha\| = \|\hat{\alpha}_x\alpha_x\| + \|\hat{\alpha}_y\alpha_y\|$. So, both the quantities represent the required contrast. But, as proved by Sugiyama et al. (123) the linear combination of them, LSGFD, is more bias corrected estimator. Also, let

$$g(x,y) = \boldsymbol{\theta}(x,y)^T\Psi^{(1)}(x,y) = [\boldsymbol{\theta}(x,y)^T\frac{\partial}{\partial x}\Psi(x,y) \quad \boldsymbol{\theta}(x,y)^T\frac{\partial}{\partial y}\Psi(x,y)] \tag{2.76}$$

where, $\boldsymbol{\theta}(x,y) = (\theta_1, \theta_b, ..., \theta_b)^T$ is the parameter vector; b denotes the number of basis functions and $\Psi(x,y) = (\psi_1, \psi_2, ..., \psi_b)^T$ is the basis function vector. So, with regularization function $R(\boldsymbol{\theta}) = \boldsymbol{\theta}^T\boldsymbol{\theta}$ and λ as the regularization parameter,

$$\text{LSGFD}(\boldsymbol{\theta}) = \boldsymbol{\theta}^T\left(\mathbf{V}_{Rx} + \mathbf{V}_{Ry}\right)\boldsymbol{\theta} - 2\left(\mathcal{V}_{Rx} + \mathcal{V}_{Ry}\right)^T\boldsymbol{\theta} + \lambda\boldsymbol{\theta}^T\boldsymbol{\theta} \tag{2.77}$$

$$\text{where, } \mathbf{V}_{R(b\times b)} = \int_{\mathbb{R}}\int_{\mathbb{R}} \Psi^{(1)}(x,y)\Psi^{(1)T}(x,y)dxdy \tag{2.78}$$

$$\mathbf{V}_{Rx(b\times b)} = \int_{\mathbb{R}}\int_{\mathbb{R}} \frac{\partial}{\partial x}\Psi(x,y)\frac{\partial}{\partial x}\Psi^T(x,y)dxdy \tag{2.79}$$

$$\mathbf{V}_{Ry(b\times b)} = \int_{\mathbb{R}}\int_{\mathbb{R}} \frac{\partial}{\partial y}\Psi(x,y)\frac{\partial}{\partial y}\Psi^T(x,y)dxdy \tag{2.80}$$

$$\mathcal{V}_{R(b\times 1)} = \int_{\mathbb{R}}\int_{\mathbb{R}} \Psi^{(1)}(x,y)(\nabla p_{xy}(x,y) - \nabla(p_x(x)p_y(y)))dxdy \tag{2.81}$$

$$\mathcal{V}_{Rx(b\times 1)} = \int_{\mathbb{R}}\int_{\mathbb{R}} \frac{\partial}{\partial x}\Psi(x,y)\frac{\partial}{\partial x}(p_{xy}(x,y) - (p_x(x)p_y(y)))dxdy \tag{2.82}$$

$$\mathcal{V}_{Ry(b\times 1)} = \int_{\mathbb{R}}\int_{\mathbb{R}} \frac{\partial}{\partial y}\Psi(x,y)\frac{\partial}{\partial y}(p_{xy}(x,y) - (p_x(x)p_y(y)))dxdy \tag{2.83}$$

The optimal value of parameter vector $\boldsymbol{\theta}(x,y)$ can be obtained by minimizing the gradient of

LSGFD and obtained as

$$\boldsymbol{\theta}^* = (\mathbf{V}_R + \lambda \mathbf{I}_b)^{-1} \mathcal{V}_R \tag{2.84}$$

where, $\mathbf{I}_b$ is a b-dimensional identity matrix.

2.10.1 Φ_2^{GFD} Estimation through Multiplicative Kernel Model

Let us use multiplicative Gaussian kernel function as a basis function placed at the selected sample points. So, $\Psi(x,y) = \mathbf{k}(x)$ o $\mathbf{l}(y)$ and $[\Psi(x,y)]_l = K(x,x_l)L(y,y_l)$. The required quantities $\mathbf{V}_{Rx}$, $\mathbf{V}_{Ry}$ and the sample estimates of $\mathcal{V}_{Rx}$ and $\mathcal{V}_{Ry}$ are obtained as under:

$$[\mathbf{V}_{Rx(b\times b)}]_{ll'} = \int_{\mathbb{R}}\int_{\mathbb{R}} h^{-2}(x_l - x)K(x,x_l)L(y,y_l) \cdot h^{-2}(x_{l'} - x)K(x,x_{l'})L(y,y_{l'})dxdy \tag{2.85}$$

Using, the convolution property of Gaussian in equation (2.32), the following result can be derived.

$$\int_{\mathbb{R}} (x_l - x)(x_{l'} - x)G(x,x_l)G(x,x_{l'})dx = \frac{1}{2\sqrt{\pi}h}\left[\frac{2h^2 - (x_l - x_{l'})^2}{4}\right]\exp\left(-\frac{(x_l - x_{l'})^2}{4h^2}\right) \tag{2.86}$$

Applying the result in Equation (2.86) to Equation (2.85), we get:

$$[\mathbf{V}_{Rx(b\times b)}]_{ll'} = \left[\frac{2h^2 - (x_l - x_{l'})^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_{l'})^2 + (y_l - y_{l'})}{4h^2}\right) \tag{2.87}$$

Similarly,

$$[\mathbf{V}_{Ry(b\times b)}]_{ll'} = \left[\frac{2h^2 - (y_l - y_{l'})^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_{l'})^2 + (y_l - y_{l'})}{4h^2}\right) \tag{2.88}$$

$$\Rightarrow [\mathbf{V}_{R(b\times b)}]_{ll'} = \left[\frac{2nh^2 - \{(x_l - x_{l'})^2 + (y_l - y_{l'})^2\}}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_{l'})^2 + (y_l - y_{l'})}{4h^2}\right) \tag{2.89}$$

Same way,

$$
\begin{aligned}
[\mathcal{V}_{Rx(b\times 1)}]_l = & \int_{\mathbb{R}}\int_{\mathbb{R}} h^{-2}(x_l - x)K(x, x_l)L(y, y_l)\left\{\sum_{i=1}^{N} h^{-2}(x_i - x)K(x, x_i)L(y, y_i)\right. \\
& \left. - \sum_{j=1}^{N}\sum_{i=1}^{N} h^{-2}(x_i - x)K(x, x_i)L(y, y_j)\right\} dxdy \qquad (2.90) \\
= & \frac{1}{N}\sum_{i=1}^{N}\left[\frac{2h^2 - (x_l - x_i)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_l - y_i)^2}{4h^2}\right) \\
& - \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{2h^2 - (x_l - x_i)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_l - y_j)^2}{4h^2}\right) \qquad (2.91)
\end{aligned}
$$

$$
\begin{aligned}
[\mathcal{V}_{Ry(b\times 1)}]_l = & \int_{\mathbb{R}}\int_{\mathbb{R}} h^{-2}(y_l - y)K(x, x_l)L(y, y_l)\left\{\sum_{i=1}^{N} h^{-2}(y_i - y)K(x, x_i)L(y, y_i)\right. \\
& \left. - \sum_{j=1}^{N}\sum_{i=1}^{N} h^{-2}(y_j - y)K(x, x_i)L(y, y_j)\right\} dxdy \qquad (2.92) \\
= & \frac{1}{N}\sum_{i=1}^{N}\left[\frac{2h^2 - (y_l - y_i)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_l - y_i)^2}{4h^2}\right) \\
& - \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{2h^2 - (y_l - y_j)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_l - y_j)^2}{4h^2}\right) \qquad (2.93)
\end{aligned}
$$

$$
\begin{aligned}
\Rightarrow [\mathcal{V}_{R(b\times 1)}]_l = & \frac{1}{N}\sum_{i=1}^{N}\left[\frac{2nh^2 - \{(x_l - x_i)^2 + (y_l - y_i)^2\}}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_l - y_i)^2}{4h^2}\right) \\
& - \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{2nh^2 - \{(x_l - x_i)^2 + (y_l - y_i)^2\}}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_l - y_j)^2}{4h^2}\right) \\
& \qquad (2.94)
\end{aligned}
$$

Thus, the parameter vector ($\boldsymbol{\theta}$), the scalar value LSGFD $= ||\boldsymbol{\Delta}||_2$ and the $\boldsymbol{\Delta}(x, y)$ - all are obtained in terms of the reference and cross reference IIMs.

The interaction matrices for RIP ($\mathbf{V}_R$) and CRIP ($\mathcal{V}$) for LSGFD estimation could be same as those used to estimate the LSFD. But, for more precise estimations it is better to recalculate them using suitable bandwidth parameter for density derivative estimator, which is usually smaller than that used for density estimation.

2.10.2 Φ_2^{GFD} Estimation through Multiplicative Kernel Basis Placed at Paired and Un-paired Samples

Similar to the LSFD2 estimator, LSGFD2 estimator can be derived using multiplicative kernel basis at unpaired samples and *Kronecker* structure to achieve precise computation. So, the basis vector is defined as: $\Psi(x, y) = [(\mathbf{I}_b \otimes \mathbf{k}(x))$ o $(\mathbf{l}(y) \otimes \mathbf{I}_b)]$; where, o denotes *Hadamard* product and $\otimes$ denotes the *Kronecker* product. The approximation $g(x, y)$ is defined as:

$$g(x, y) = \text{vec}(\boldsymbol{\Theta})^T \Psi^{(1)}(x, y) \tag{2.95}$$

$$= \begin{bmatrix} \sum_{j=1}^{N} \sum_{i=1}^{N} \theta_{ij} h^{-2}(x_i - x) K(x, x_i) L(y, y_j) \\ \sum_{j=1}^{N} \sum_{i=1}^{N} \theta_{ij} h^{-2}(y_j - y) K(x, x_i) L(y, y_j) \end{bmatrix} \tag{2.96}$$

where, $\boldsymbol{\Theta}$ is a $b \times b$ parameter matrix and vec$(\cdot)$ is a vectorization function. The required quantities $\mathbf{V}_{Rx}$, $\mathbf{V}_{Ry}$ and the sample estimates of $\mathcal{V}_{Rx}$ and $\mathcal{V}_{Ry}$ are obtained as under:

$$[\mathbf{V}_{Rx(b\times b)}]_{mn} = \int_{\mathbb{R}} \int_{\mathbb{R}} h^{-4}(x_i - x) K(x, x_i) L(y, y_j)(x_k - x) K(x, x_k) L(y, y_l) dx dy \tag{2.97}$$

$$\text{where, } m = i + (j-1)b \text{ and } n = k + (l-1)b \tag{2.98}$$

$$= \left[\frac{2h^2 - (x_i - x_k)^2}{2^{d+2}\pi^{d/2}h^{d+4}} \right] \exp\left(-\frac{(x_i - x_k)^2 + (y_j - y_l)}{4h^2} \right) \tag{2.99}$$

$$[\mathbf{V}_{Ry(b\times b)}]_{mn} = \left[\frac{2h^2 - (y_j - y_l)^2}{2^{d+2}\pi^{d/2}h^{d+4}} \right] \exp\left(-\frac{(x_i - x_k)^2 + (y_j - y_l)}{4h^2} \right) \tag{2.100}$$

$$\Rightarrow [\mathbf{V}_{R(b\times b)}]_{ll'} = \left[\frac{2nh^2 - \{(x_i - x_k)^2 + (y_j - y_l)^2\}}{2^{d+2}\pi^{d/2}h^{d+4}} \right] \exp\left(-\frac{(x_i - x_k)^2 + (y_j - y_l)}{4h^2} \right) \tag{2.101}$$

Same way,

$$
\begin{aligned}
[\mathcal{V}_{Rx(b\times 1)}]_q &= \int_{\mathbb{R}}\int_{\mathbb{R}} h^{-2}(x_l - x)K(x,x_l)L(y,y_{l'})\left\{\sum_{i=1}^{N} h^{-2}(x_i - x)K(x,x_i)L(y,y_i)\right. \\
&\quad \left. -\sum_{j=1}^{N}\sum_{i=1}^{N} h^{-2}(x_i - x)K(x,x_i)L(y,y_j)\right\} dxdy \qquad (2.102) \\
&\quad \text{where, } q = l + (l'-1)b \qquad (2.103) \\
&= \frac{1}{N}\sum_{i=1}^{N}\left[\frac{2h^2 - (x_l - x_i)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_{l'} - y_j)^2}{4h^2}\right) \\
&\quad -\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{2h^2 - (x_l - x_i)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_{l'} - y_j)^2}{4h^2}\right) \qquad (2.104) \\
[\mathcal{V}_{Ry(b\times 1)}]_q &= \int_{\mathbb{R}}\int_{\mathbb{R}} h^{-2}(y_{l'} - y)K(x,x_l)L(y,y_{l'})\left\{\sum_{i=1}^{N} h^{-2}(y_i - y)K(x,x_i)L(y,y_i)\right. \\
&\quad \left. -\sum_{j=1}^{N}\sum_{i=1}^{N} h^{-2}(y_j - y)K(x,x_i)L(y,y_j)\right\} dxdy \qquad (2.105) \\
&\quad \text{where, } q = l + (l'-1)b \qquad (2.106) \\
&= \frac{1}{N}\sum_{i=1}^{N}\left[\frac{2h^2 - (y_{l'} - y_j)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_{l'} - y_i)^2}{4h^2}\right) \\
&\quad -\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{2h^2 - (y_{l'} - y_j)^2}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_{l'} - y_j)^2}{4h^2}\right) \qquad (2.107) \\
\Rightarrow [\mathcal{V}_{R(b\times 1)}]_l &= \frac{1}{N}\sum_{i=1}^{N}\left[\frac{2nh^2 - \{(x_l - x_i)^2 + (y_{l'} - y_j)^2\}}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_{l'} - y_i)^2}{4h^2}\right) \\
&\quad -\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{2nh^2 - \{(x_l - x_i)^2 + (y_{l'} - y_j)^2\}}{2^{d+2}\pi^{d/2}h^{d+4}}\right]\exp\left(-\frac{(x_l - x_i)^2 + (y_{l'} - y_j)^2}{4h^2}\right) \\
&\qquad (2.108)
\end{aligned}
$$

To estimate Φ_2^{α}, the optimal parameter matrix $\mathbf{\Theta}$ is needed. The Equation (2.54) can be written as:

$$\mathbf{V}_R \text{vec}(\mathbf{\Theta}) + \lambda \text{vec}(\mathbf{\Theta}) = \mathcal{V}_R$$

This is the famous *discrete Sylvester equation* and requires $O(b^3)$ computations to solve it.
Now, the Equation (2.50) can be given as under:

$$\text{LSGFD2} = \text{trace}(\mathbf{\Theta}^T \mathbf{V}_R(x)\mathbf{\Theta}\mathbf{V}_R(y)^T) - 2\mathcal{V}_R^T\text{vec}(\mathbf{\Theta}) \tag{2.109}$$

2.11 Empirical Verification of the Derived Estimators as Independence Measures

The derived four estimators - LSFD, LSFD2, LSGFD and LSGFD2 - need empirical verification. A simple test experiment is designed that verifies their ability to separate the independent and dependent signals. Further testing, as a BSS contrast, has been left for the future sections. The estimators need bandwidth parameter $(\mathbf{h})$ selection for multivariate kernel density estimation (KDE) and the regularization parameter λ. Conventionally, the least squares based direct estimation methods use a Cross Validation (CV) method to select both the parameters. The CV method is computationally demanding if good number of choices for a free parameter are provided to obtain accuracy in estimation. Instead, the Silverman's *rule-of-thumb* (ROT) (116), balancing computation and optimal parameter value, is used for selecting $(\mathbf{h})$. The Experiment uses ROT for $\lambda = 0.01$ for the test experiments.

Experiment (Independence test): Let there be generated two uniformly distributed independent signals $X(1,:)$ and $X(2,:)$ with 300 samples each. Let there be generated a dependent signal: $Y(1,:) = \sin(X(1,:)/20 * \pi)$. Find the estimated values for the independent signals - $X(1,:)$ and $X(2,:)$ - and dependent signals - $X(1,:)$ and $Y(1,:)$.

The results are tabulated in the following Table 2.1. Each entry in the Table is a mean of 100 trials. The results show that all the estimators are able to give estimator value sufficiently low for independent signals than dependent signals.

Experiment (Independence test against varying number of samples): Let there be generated two uniformly distributed independent signals $X(1,:)$ and $X(2,:)$ with varying number of samples each. Let there be generated a dependent signal: $Y(1,:) = \sin(X(1,:)/20 * \pi)$ for each experi-

Table 2.1: Performances of the derived independence measures with their estimation techniques: on the test set with independence and dependence signals; number of samples 300; kernel bandwidth parameter h using ROT; regularization parameter $\lambda = 0.01$. The table entries indicate mean of 100 trials.

Test Condition	**LSFD**	**LSFD2**	**LSGFD**	**LSGFD2**
independent signals	0.4725e-03	0.5057e-03	0.2915e-03	0.1165e-03
dependent signals	0.0180	0.0411	0.0055	0.0091

ment. Find the estimated values for the independent signals - $X(1,:)$ and $X(2,:)$ - and dependent signals - $X(1,:)$ and $Y(1,:)$.

The results are tabulated in the following Table 2.2, for LSFD and LSFD2 estimators, and Table 2.3, for LSGFD and LSGFD2 estimators. Each entry in the tables is a mean of 100 trials. For a given estimator there is tabularized measure value for independent signals (X_1 and X_2), dependent signals (X_1 and Y_1), their differences and the mean time taken to calculate the measure for dependent and independent signals in given iterations. The results show that all the estimators are able to give estimator value sufficiently low for independent signals than dependent signals. With increase in number of samples, the measure for independent signals decreases and that for dependent signals increases. Thus, with increasing number of samples the measures are more able to separate the independent and dependent signals. The exception is LSGFD2 estimator. For it, the dependence value is also decreasing with increasing samples. Still, there is a sufficient gape that can separate the dependent and independent signals. The results show that the estimators LSFD and LSGFD take almost same amount of time. Similarly, the estimators LSFD2 and LSGFD2 take almost same amount of time; but quite higher than that for LSFD and LSGFD estimators The figure 2.1 show the relative time complexity of LSGFD and LSGFD2 estimators. The time is in milliseconds The experiments were performed on a Intel(R) Core(TM)2 Duo CPU, 2.93 GHz, 4.00 GB Internal RAM, 32 bit Windows 7 Professional, MATLAB R2010a.

Table 2.2: Performances of the derived independence measures with their estimation techniques against varying number of samples: on the test set with independence and dependence signals; kernel bandwidth parameter h using ROT; regularization parameter $\lambda = 0.01$. The **table entries** indicate mean of 100 trials and **after multiplication by** $1e03$.

Column:C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
No. of Samples	**LSFD**				**LSFD2**			
	(X_1, X_2)	(X_1, Y_1)	$C_3 - C_2$	time(Sec)	(X_1, X_2)	(X_1, Y_1)	$C_7 - C_6$	time(Sec)
50	1.3350	6.5648	5.2298	0.9597	1.4513	12.6976	11.2463	5.4452
100	1.1493	9.2332	8.0839	2.4046	1.1806	18.4183	17.2377	22.320
150	0.9277	10.7252	9.7975	5.3477	0.9406	21.6348	20.6942	61.929
200	0.7508	12.1338	11.3830	10.2709	0.7576	24.6161	23.8585	190.87
300	0.6726	14.1059	13.4333	25.3974	0.6755	28.6221	27.9466	862.99
500	0.5338	17.5576	17.0238	40.6980	0.5358	35.0553	34.5194	1900.2

2.11.1 Parameter Selection in the Derived Estimators for BSS

The Experiment justified the use of ROT for bandwidth selection, instead CV for the same. But, both the methods have one more problem for BSS like signal processing and machine learning applications. Compare to the applications in previous experiment, where the comparison was at an event or at a point, those applications require to find the most optimal from a given solution set. If

Table 2.3: Performances of the derived independence measures with their estimation techniques against varying number of samples: on the test set with independence and dependence signals; kernel bandwidth parameter h using ROT; regularization parameter $\lambda = 0.01$. The **table entries** indicate mean of 100 trials and **after multiplication by** $1e03$.

Column:C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
No. of Samples	**LSGFD**				**LSGFD2**			
	(X_1, X_2)	(X_1, Y_1)	$C_3 - C_2$	time(Sec)	(X_1, X_2)	(X_1, Y_1)	$C_7 - C_6$	time(Sec)
50	0.8631	5.0004	4.1372	0.9149	1.3400	8.7550	7.4150	4.98629
100	0.7176	6.1479	5.4303	2.6821	0.6148	6.4483	5.8334	22.5187
150	0.5337	6.4147	5.8810	6.2353	0.3363	5.1740	4.8377	62.8617
200	0.4421	6.7454	6.3033	12.351	0.2246	4.5042	4.2796	191.534
300	0.3713	7.0058	6.6345	32.491	0.1401	3.6119	3.4718	865.969
500	0.2672	7.4779	7.2106	52.082	0.0717	2.8410	2.7692	1907.48

CV method is used, there needs to be found new parameter value at every point in consideration. That will be computationally too demanding. The ROT assumes Gaussian distribution for the unknown PDF. The feasible solution set for the problem is expected to have varying properties, like, varying distances from Gaussianity and others. Ideally, same bandwidth parameter is not best for all points. For example; in case of the BSS application, the goal is to find the most non-Gaussian (independent) components. For this goal, assuming Gaussianity for the whole solution set is contradictory and sure way to bring estimation errors. This brings the need to use data dependent rules for kernel smoothing parameter that takes into consideration the variation in the distributions of solutions and is also computationally efficient. Such a rule, identified as Extended ROT (ExROT) is derived in the next Chapter 3 based on an assumption that the density being estimated is near Gaussian and can be approximated using Gram-Charlier A Series. The rule is used for the contrast estimation in BSS in the following experiment.

2.12 Conclusion

The chapter proves that the Gradient Function Difference (GFD) being zero everywhere imply independence. For a bounded support random vector the Hessian Function Difference (HFD) being zero everywhere imply independence. Accordingly, L^P measure of FD, GFD and HFD are proved to be independence measures. They are used to derive contrast functions for simultaneous ICA and BSS. The contrast functions are proved to satisfy the properties of Scale Invariance, Dominance and Discrimination, avoiding spurious global maxima. There has also been derived least squares based two methods to estimate L^2 of FD and L^2 of GFD contrasts using multiplicative kernel basis. In the first method the basis are placed at only joint samples and in the another method basis are placed at both paired and unpaired samples. The first method requires computations of the order of $O(b^2 + N(b + n - 1))$ and the second method requires that of the order of $O(b^3 + (N + n)b^n))$.

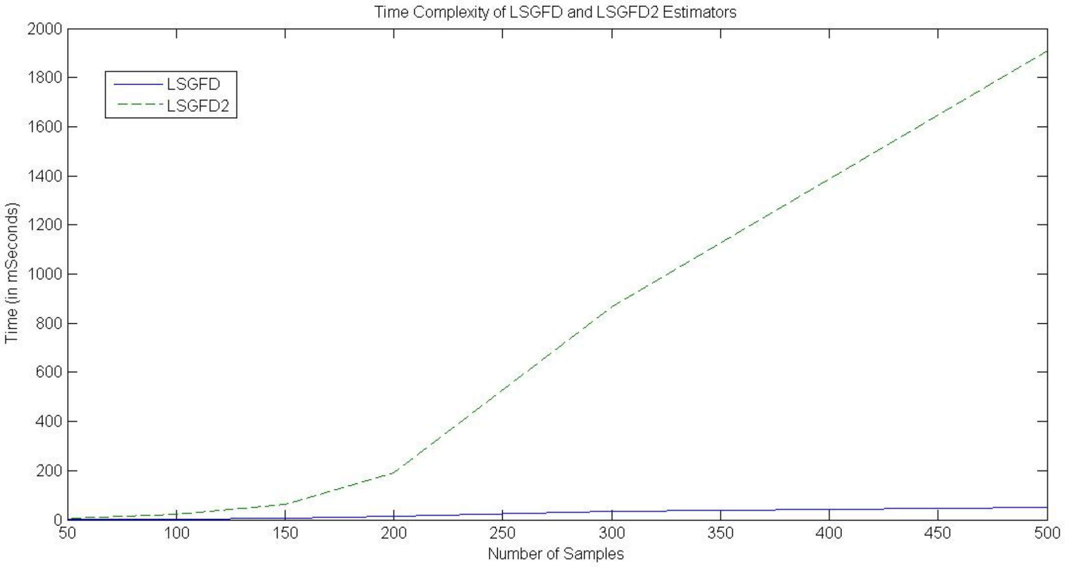

Figure 2.1: Time complexity of LSGFD and LSGFD2 estimators against varying number of samples; Time is in milli Seconds; Experiments were performed on Intel(R) Core(TM)2 Duo CPU, 2.93 GHz, 4.00 GB Internal RAM, 32 bit Windows 7 Professional, MATLAB R2010a

But, the second method requires less samples for the same performance. The empirical verification justifies the derived contrasts for BSS applications. But, further experiments are needed to have the comparison with other contrasts on separation quality against varying number of sources and varying number of samples. The required performance analysis is restricted here and targeted in the future versions of this article. The assumption of Gaussianity for the whole solution set of BSS, when the goal is to achieve maximization of Gaussianity, does not seem proper. Over all, the estimation methods lag computationally efficient data dependent bandwidth selection method. So, deriving a suitable bandwidth parameter estimation technique for BSS applications is the target for next Chapter 3.

Chapter 3

Extended Rule-of-Thumb for Bandwidth Selection in Kernel Methods

The chapter derives a novel Gram-Charlier A (GCA) Series based Extended *Rule-of-Thumb* (ExROT) for bandwidth selection in Kernel Density Estimation (KDE). There are existing various bandwidth selection rules achieving minimization of the Asymptotic Mean Integrated Square Error (AMISE) between the estimated probability density function (PDF) and the actual PDF. The rules differ in a way to estimate the integration of the squared second order derivative of an unknown PDF $(f(\cdot))$, identified as the roughness $R(f''(\cdot))$. The simplest *Rule-of-Thumb* (ROT) estimates the $R(f''(\cdot))$ with an assumption that the density being estimated is Gaussian. Intuitively, better estimation of $R(f''(\cdot))$ and consequently better bandwidth selection rules can be derived, if the unknown PDF is approximated through an infinite series expansion based on a more generalized density assumption. As a demonstration and verification to this concept, the ExROT derived in the chapter uses an extended assumption that the density being estimated is near Gaussian. This helps use of the GCA expansion as an approximation to the unknown near Gaussian PDF. The ExROT for univariate KDE is extended to that for multivariate KDE. The required multivariate AMISE criteria is re-derived using elementary calculus of several variables, instead of Tensor calculus. The derivation uses the Kronecker product and the vector differential operator to achieve the AMISE expression in vector notations. There is also derived ExROT for kernel based density derivative estimator.

The part of the chapter derives multivariate Generalized Gram-Charlier (GGC) series that expands an unknown joint probability density function (*pdf*) of a random vector in terms of the differentiations of the joint *pdf* of a reference random vector. Conventionally, the higher order differentiations of a multivariate *pdf* in GGC series will require multi-element array or tensor representations. But, the current chapter derives the GGC series in vector notations. The required higher order differentiations of a multivariate *pdf* in vector notations are achieved through application of a specific Kronecker product based differentiation operator. Overall, the chapter uses only elementary calculus of several variables; instead Tensor calculus; to achieve the extension of an existing specific derivation for GGC series in univariate to multivariate. The derived multivariate GGC expression is more elementary as using vector notations compare to the coordinatewise tensor notations and more comprehensive as apparently more nearer to its counterpart for univariate. The same advantages are shared by the other expressions obtained in the chapter; such as the mutual relations between cumulants and moments of a random vector, integral form of a multivariate *pdf*, integral form of the multivariate Hermite polynomials, the multivariate Gram-Charlier A (GCA) series and others.

3.1 Introduction

Continuous probability density function (PDF) estimation using kernel methods is widely used in statistics, machine learning and signal processing (116). The optimal estimation depends upon the selected kernel function and its spread decided by the smoothing or bandwidth parameter. The selection of kernel has limited impact on optimal PDF estimation, although Epanechnikov kernel is the most optimal kernel (45). On the other hand, the optimal value of bandwidth parameter avoids either too rough or too smooth estimation of an unknown PDF.

There exist variety of rules for bandwidth selection in KDE. The rules vary based on the criteria to measure accuracy of density estimation and to satisfy the used criteria. The brief survey of data driven bandwidth selectors is provided by Jones et al. (69), Park and Marron (85), Park and Turlach (86), Sheather (115), Wand and Jones (136). The Asymptotic Mean Integrated Square Error (AMISE) between the estimated PDF and the actual PDF is the most widely used performance criteria to derive the rules, though there are many others. The AMISE criteria requires estimating the roughness of the squared second order PDF $(R(f''(\cdot)))$ as a prior step to estimate the kernel bandwidth parameter, where the roughness of a function is defined as integration of the squared function. The rules, based on the AMISE as a performance criteria, differ in a way to estimate the $R(f''(\cdot))$. The simplest *Rule-of-Thumb (ROT)*, satisfying the AMISE, by Silverman (116) assumes Gaussian distribution for the unknown density. It is not the most optimal bandwidth selector but is used either as a very fast reasonably good estimator or as a first estimator in multistage bandwidth selectors. More precise *solve-the-equation plug-in* rules (113, 114) use estimation of integrated squared density derivative functional to estimate $R(f''(\cdot))$. They demand high computations to solve a non-linear equation using iterative methods. They use ROT as a very first estimate. The fastest ϵ-exact approximation algorithm based *solve-the-equation plug-in* rule (101, 102) requires $O(N + M)$ order of computations, where N is number of samples and M is the selected number of evaluation points. So, deriving a data dependent bandwidth parameter selection rule for KDE that balances accuracy and computation is the goal of this chapter.

The chapter achieves this goal by deriving an *Extended Rule-of-Thumb* (ExROT). The assumption about Gaussianity of an unknown PDF in ROT is too restrictive. Expressing an unknown PDF, in terms of an infinite series of higher order statistics, based on a more generalized assumption should result into a better bandwidth selection rule. As a verification and demonstration to this concept, the ExROT extends the Gaussian assumption in ROT to near Gaussian assumption. This facilitates use of cumulants based Gram-Charlier A (GCA) Series expansion as an approximation for the unknown PDF to satisfy the same AMISE criteria. The empirical simulations prove that the ExROT for bandwidth selection is better than the ROT, with respect to an integrated mean square error (IMSE) or MISE performance criteria, for all types of nongaussian unimodal distributions including the skewed, the kurtotic and with outlier distributions. This is achieved with computation comparable to the ROT and too less compare to the *ϵ-exact solve-the-equation plug-in* rule.

The ExROT for bandwidth selection in univariate KDE is extended to the similar for multivariate KDE and kernel based multivariate density derivative estimation. The ExROT for multivariate KDE requires multivariate expression for AMISE, multivariate Taylor Series expansion, multivariate Hermite polynomials and multivariate GCA Series expansion. The required multivariate AMISE is conventionally derived using gradient and Hessian of the PDF of a random vector (136). Conventionally, the other required multivariate expressions are also derived using Tensor calculus, as higher order derivatives of a multivariate functions are involved. Often, the corresponding final expressions requires coordinatewise representations. But recently, the higher order cumulants (105, 125) and multivariate Hermite polynomials (61, 125) are derived using only elementary calculus of several variables. This is achieved by replacing conventional multivariate differentiations by repeated applications of the Kronecker product to vector differential operator. The derived expressions are also more elementary as using vector notations and more comprehensive as apparently more nearer to their counterparts in univariate. The same approach has been used here to derive multivariate AMISE criteria in a vector notations. Overall, the multivariate ExROT is derived using the multivariate Taylor Series, the multivariate cumulants and the multivariate Hermite polynomials derived by Holmquist (61), Terdik (125), the multivariate GCA derived by Bhaveshkumar (15) and the multivariate AMISE obtained in Section 3.15 of this chapter. There is also derived bandwidth selection rule for kernel based density derivative estimation.

The next Section 3.2 derives the univariate AMISE criteria and gives brief on the existing rules for data driven kernel bandwidth selectors. The Section 3.3 derives the ExROT. The performance analysis is done using two separate experiments in Section 3.4. The preliminary background on multivariate KDE, Kronecker product, multivariate Taylor Series and others is briefed in Section **??**. A compact derivation for multivariate GGC Series and GCA Series is provided in Section 3.13. Then, the Section 3.15 derives multivariate AMISE in a vector form using Kronecker Product. The multivariate AMISE and multivariate GCA Series are used to derive ExROT for multivariate

KDE in Section 3.16. Similarly, the Section 3.18 derives ExROT for density derivative estimation. Finally, the chapter is concluded in Section 3.19.

3.2 Bandwidth Selection Criteria and Selectors

Let there be a random variable X with an unknown PDF $f(x)$. Given N realizations $x_1, x_2, \ldots, x_N$ of X, the kernel density estimate $f(\hat{x})$ is given by

$$f(\hat{x}) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{h} \mathcal{K}\left(\frac{x - x_i}{h}\right) \tag{3.1}$$

where, $\mathcal{K}(u)$ is the kernel function and h is the bandwidth parameter deciding spread of the kernel. Usually, $\mathcal{K}(u)$ is a symmetric, positive definite and bounded function, i.e. it satisfies the following properties:

$$\mathcal{K}(u) \geq 0, \; \int_{-\infty}^{\infty} \mathcal{K}(u) du = 1,$$
$$\int_{-\infty}^{\infty} u\mathcal{K}(u) du = 0, \; \int_{-\infty}^{\infty} u^2 \mathcal{K}(u) du = \mu_2(\mathcal{K}) < \infty$$

The accuracy of a PDF estimation can be quantified by the available distance measures between PDFs; like; L_1 norm based mean integrated absolute measure, L_2 norm based mean integrated square error (MISE), Kullback-Leibler divergence and others. The optimal smoothing parameter (the bandwidth) h is obtained by minimizing the selected distance measure. The most widely used criteria MISE or IMSE (Integrated Mean Square Error) based bandwidth selection rule, as in (116, 136), is detailed in the Appendix .1. It is briefed as under:

$$\begin{aligned} \text{MISE}(f(\hat{x})) &= E\{ISE(f(x), f(\hat{x}))\} = E\left\{\int_{-\infty}^{\infty} (f(\hat{x}) - f(x))^2 dx\right\} \\ &= \int_{-\infty}^{\infty} \text{Bias}^2(f(\hat{x}))dx + \int_{-\infty}^{\infty} \text{Var}(f(\hat{x}))dx \\ &= \frac{h^4}{4}\mu_2^2(\mathcal{K})R(f'') + \frac{1}{Nh}R(\mathcal{K}) + O(h^4) + O\left(\frac{h}{N}\right) \end{aligned} \tag{3.2}$$

where, $\mu_2(\mathcal{K}) = \int_{\mathbb{R}} z^2\mathcal{K}(z)dz$, $R(f'') = \int_{\mathbb{R}} (f''(x))^2 dx$ and $R(\mathcal{K}) = \int_{\mathbb{R}} \mathcal{K}^2(z)dz$. In general, $R(g) = \int g^2(z)dz$ is identified as the roughness of function $g(x)$. An asymptotic large sample approximation AMISE is obtained, assuming $\lim_{N\to\infty} h = 0$ and $\lim_{N\to\infty} Nh = \infty$ i.e. h reduces to 0 at a rate slower than $1/N$.

$$\text{AMISE}(\hat{f(x)}) = \frac{h^4}{4}\mu_2^2(\mathcal{K})R(f'') + \frac{1}{Nh}R(\mathcal{K}) \tag{3.3}$$

The Equation (3.3) interprets that a small h increases estimation variance, whereas, a larger h increases estimation bias. An optimal h minimizing the total AMISE($\hat{f(x)}$) is given by,

$$\Rightarrow h_{AMISE} = \left(\frac{R(\mathcal{K})}{\mu_2^2(\mathcal{K})R(f'')N}\right)^{\frac{1}{5}} \tag{3.4}$$

$$= (CN)^{-1/5} \quad \text{where, } C = \frac{\mu_2^2(\mathcal{K})R(f'')}{R(\mathcal{K})} \tag{3.5}$$

Thus, the optimal bandwidth parameter depends upon some of the kernel parameters, number of samples and the second derivative of the actual PDF being estimated.

3.2.1 Brief Survey on the bandwidth selectors

As the bandwidth selection rules vary based on the choice of performance criteria for density estimation, they also vary based on the way the performance criteria is optimized. The various rules, satisfying AMISE, for bandwidth parameter selection differ in the way $R(f'')$ is estimated. The first group of rules named *scale measures* give rough estimate of the bandwidth parameter with less computation. It includes Silverman's *Rule-of-Thumb* that estimates h assuming $f(x)$ being Gaussian (116). For a Gaussian PDF $R(f'') = \frac{3\sigma^{-5}}{8\sqrt{\pi}}$ and for a Gaussian kernel $R(k) = \frac{1}{2\sqrt{\pi}}$. Accordingly, using equation (3.4), we get:

$$h_{rot} = 1.0592\sigma N^{-1/5} \tag{3.6}$$

where, σ is the standard deviation of $f(x)$. There are many other rules based on the assumption of other parametric family. There are also rules based on oversmoothed h, difference based h and others briefed by (67).

An another group of rules is based on the more accurate at high computation *plug-in* rules. They plug-in the kernel based estimate of the $R(f'')$. The *direct plug-in* rules estimate derivative of the density functional instead of estimating actual derivatives. Every r^{th} order derivative functional estimation requires $(r+2)^{th}$ order estimate and pilot bandwidth to start with. Assuming, some parametric density for the $(r+2)^{th}$ order density the pilot bandwidth is selected and cumulatively the bandwidth parameter to estimate $f(x)$ is obtained. The *solve-the-equation plug-in* rules use the same approach but, instead of assuming bandwidth parameter, they optimize it by directly putting it into the AMISE. This requires solving a non-linear equation iteratively. They have better performances at high computation. Other than the rules for bandwidth selection, there are also cross-validation methods selecting the best from a user-defined list of bandwidth parameter based on some performance criteria. But, there is always a compromise between a length of the list for possible bandwidth parameters and the amount of computation.

Over all, a bandwidth selection rule that achieves precise bandwidth parameter at low computation is still open for research.

3.3 Extended *Rule-of-Thumb* (ExROT) Bandwidth Selector

The Gaussianity assumption for an unknown PDF to estimate $R(f'')$ is too restrictive. Intuitively, better PDF estimations can be derived if $R(f'')$ is estimated by approximating PDFs through infinite series expansion. As a verification and demonstration to this concept, ExROT is derived in this section using cumulants based Gram-Charlier A (GCA) series expansion of $f(x)$ based on near Gaussianity assumption for an unknown PDF $f(x)$. There exists multiple ways to derive univariate GCA series, as reported by Hald (2000). It's generalization to univariate Generalized Gram-Charlier (GGC) series is derived by Schleher (1977); Cohen (1998); Berberan-Santos (2007); Cohen (2011). The GCA series is given as:

$$f(x) = \exp\left[\sum_{r=0}^{\infty}(k_r - \gamma_r)\frac{(-D)^{(r)}}{r!}\right] G(x;\mu,\sigma) \tag{3.7}$$

where, $G(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$ is the Gaussian PDF; k_r is the r^{th} order cumulant of $f(x)$; γ_r is the r^{th} order cumulant of $G(x)$ and D is the derivative operator with respect to x. The n^{th} derivative of Gaussian is given by

$$D^{(n)}G(x;\mu,\sigma) = \frac{d^{(n)}G(x;\mu,\sigma)}{dx^n} = \frac{1}{\sigma^n}(-1)^n H_n\left(\frac{x-\mu}{\sigma}\right)G(x;\mu,\sigma) \tag{3.8}$$

where, H_n is the nth order Hermite polynomial. Accordingly, with $k_1 = \gamma_1 = \mu$, $k_2 = \gamma_2 = \sigma^2$, approximating upto the fourth order cumulants and taking $z = \frac{x-\mu}{\sigma}$; we get:

$$f(x) \approx \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}z^2\right)\left[1 + \frac{k_3}{3!\sigma^3}H_3(z) + \frac{k_4}{4!\sigma^4}H_4(z)\right] \tag{3.9}$$

Taking derivative twice with respect to x of the Equation (3.7) yields:

$$f''(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[\sum_{r=0}^{\infty}(k_r - \gamma_r)\frac{(-D)^{(r+2)}}{r!}\right]\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \tag{3.10}$$

Again, with $k_1 = \gamma_1$, $k_2 = \gamma_2$ and approximating upto fourth order cumulants; we get:

$$f''(x) \approx \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}z^2\right)\left[\frac{1}{\sigma^2}H_2(z) + \frac{k_3}{3!\sigma^5}H_5(z) + \frac{k_4}{4!\sigma^6}H_6(z)\right] \tag{3.11}$$

$$\Rightarrow R(f'') = \int_{\mathbb{R}} f''(x)^2 dx = \int_{\mathbb{R}} \frac{\exp\left(-z^2\right)}{2\pi\sigma}\left[\frac{1}{\sigma^2}H_2(z) + \frac{k_3}{3!\sigma^5}H_5(z) + \frac{k_4}{4!\sigma^6}H_6(z)\right]^2 d \tag{3.12}$$

The integration is obtained using the following rules:

$$\int_{-\infty}^{\infty} \mathrm{e}^{-ax^2} dx = \sqrt{\frac{\pi}{a}}$$

$$\int_{-\infty}^{\infty} x^n \mathrm{e}^{-ax^2} dx = \begin{cases} \frac{(n)!!}{(2a)^{n/2}}\sqrt{\frac{\pi}{a}} & \text{for } n \text{ even} \\ 0 & \text{for } n \text{ odd, as the function becomes odd} \end{cases}$$

where, $(n)!! = (n-1)(n-3)(n-5)\ldots$ is called the odd factorial. Accordingly, the quantities in Equation (3.12) are obtained as under:

$$\begin{aligned}
T_1 &= \int_{-\infty}^{\infty} G^2(z)\left[\frac{1}{\sigma^2}H_2(z)\right]^2 dx = \frac{1}{\sigma^5}\frac{3}{8\sqrt{\pi}} \\
T_2 &= \int_{-\infty}^{\infty} G^2(z)\left[\frac{1}{\sigma^5}H_5(z)\right]^2 dx = \frac{1}{\sigma^{11}}\frac{945}{64\sqrt{\pi}} \\
T_3 &= \int_{-\infty}^{\infty} G^2(z)\left[\frac{1}{\sigma^6}H_6(z)\right]^2 dx = \frac{1}{\sigma^{13}}\frac{10395}{128\sqrt{\pi}} \\
T_4 &= \int_{-\infty}^{\infty} G^2(z)\left[\frac{2}{\sigma^7}H_2(z)H_5(z)\right] dx = 0 \\
T_5 &= \int_{-\infty}^{\infty} G^2(z)\left[\frac{2}{\sigma^{11}}H_5(z)H_6(z)\right] dx = 0 \\
T_6 &= \int_{-\infty}^{\infty} G^2(z)\left[\frac{2}{\sigma^8}H_2(z)H_6(z)\right] dx = \frac{1}{\sigma^9}\frac{105}{16\sqrt{\pi}}
\end{aligned}$$

This gives:

$$R(f'') = \frac{1}{\sigma^5}\frac{3}{8\sqrt{\pi}}\left[1 + \frac{1}{\sigma^6}\frac{315}{8}\left(\frac{k_3}{6}\right)^2 + \frac{1}{\sigma^8}\frac{3465}{16}\left(\frac{k_4}{24}\right)^2 + \frac{1}{\sigma^4}\frac{35}{2}\frac{k_4}{24}\right]$$

$$R(f'') = \frac{1}{\sigma^5}\frac{3}{8\sqrt{\pi}}\left[1 + 1.0938\frac{k_3^2}{\sigma^6} + 0.3764\frac{k_4^2}{\sigma^8} + 0.7292\frac{k_4}{\sigma^4}\right] \tag{3.13}$$

Combining above Equation (3.13) with Equation (3.5), the Gram-Charlier A Series based an *Extended Rule-of-Thumb* for bandwidth parameter h_{GC} selection using near Gaussian PDF assumption and Gaussian kernel is given as under. As shown, the Silverman's *Rule-of-Thumb* is one case of the extended rule.

$$h_{ExGCA} = 1.0592\sigma(CN)^{-\frac{1}{5}} \tag{3.14}$$

$$\text{where, } C = \begin{cases} 1 & \text{both } k_3 = K_4 = 0 \text{ i.e. Gaussian PDF} \\ 1 + 0.3764\frac{k_4^2}{\sigma^8} + 0.7292\frac{k_4}{\sigma^4} & \text{if } k_3 = 0 \text{ i.e. symmetric PDF,} \\ 1 + 1.0938k_3^2 + 0.3764k_4^2 + 0.7292k_4 & \text{if } \sigma = 1, \\ 1 + 1.0938\frac{k_3^2}{\sigma^6} + 0.3764\frac{k_4^2}{\sigma^8} + 0.7292\frac{k_4}{\sigma^4} & \text{otherwise} \end{cases}$$

3.4 Performance Analysis of ExROT Bandwidth Selector

There has been performed two separate experiments to test the performance of bandwidth selector. In both the experiments, the performance is tested on a set of 15 densities selected as a test-bed for density estimators by J. S. Marron (66). The densities are shown in Figure 3.1. In both the experiments, the performance of ExROT is compared against Silverman's *Rule-of-Thumb* and the ϵ-exact approximation algorithm based *solve-the-equation plug-in* rule (101, 102).

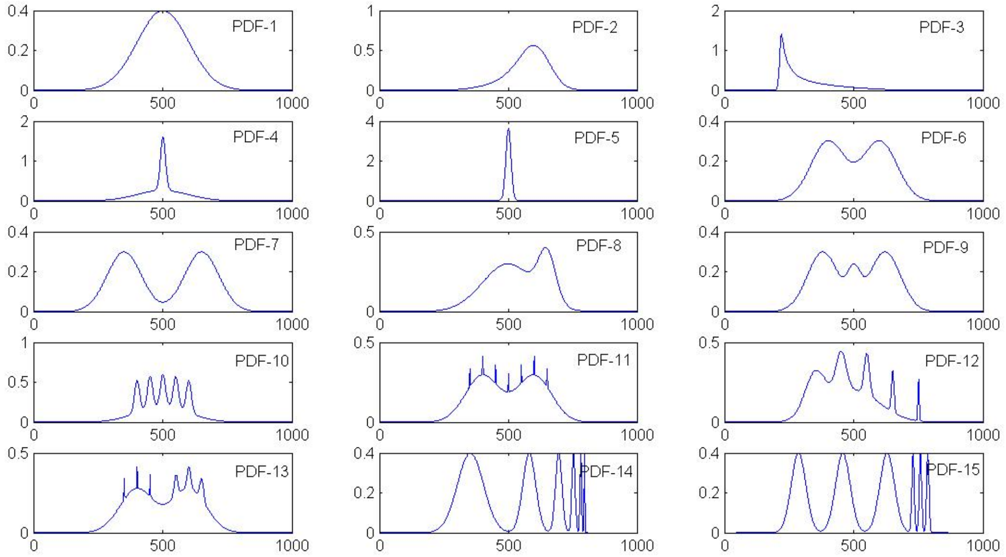

Figure 3.1: The Probability density functions (PDFs), generated through Normal mixtures, used to have performance comparison of various bandwidth selection rules for Kernel Density Estimation (KDE): (1) Gaussian (2) Skewed Unimodal (3) Strongly Skewed (4) Kurtotic Unimodal (5) Outlier (6) Bimodal (7) Separated Bimodal (8) Skewed Bimodal (9) Trimodal (10) Claw (11) Double Claw (12) Asymmetric Claw (13) asymmetric Double Claw (14) Smooth Comb (15) Discrete Comb

3.4.1 Experiment (Performance against varying PDFs)

The first experiment is done to test the performance against varying PDFs. The experiment is done with 50000 samples and for 100 trials. The results are shown in Table 3.1. Each table entry is an average of the 100 trials. The selected three rules are compared for performances against three parameters - the value of bandwidth estimated, the corresponding IMSE between the estimated PDF and the theoretical PDF and the time taken in Seconds to estimate the bandwidth. The theoretical PDF, required to calculate IMSE, is obtained from the normal mixture equations. That is why, all the 15 selected densities are generated using normal mixture equations.

The *ϵ-exact solve-the-equation plug-in* rule is the best giving minimum IMSE error in all the cases accept the pure Gaussian density estimation. For Gaussian density, ROT is slightly better than remaining both the rules. The best IMSE performance of *ϵ-exact solve-the-equation plug-in* rule is at the cost of very high computation time. The mean time to estimate the bandwidth parameter for ROT is less than one millisecond. For ExROT it is about 10 to 20 milliseconds and the same for *ϵ-exact solve-the-equation plug-in* rule is about 30 to 60 seconds. That means, the ExROT has time complexity comparable to that of the ROT. So, the IMSE performance of these two needs a comparision. The boldface values for IMSE comparison in Table 3.1, show a better between these two. It shows that in all non-Gaussian unimodal density estimation cases - skewed or kurtotic or with outlier - ExROT has outperformed ROT. The worst performance of ExROT in multimodal density estimation is due to wrong estimation of the skewness and kurtosis. The ExROT has also outperformed ROT in some of the - claw and Asymmetric claw - multimodal density estimation cases. Thus, ExROT surely is a better option to ROT in unimodal density estimation.

3.4.2 Experiment (Performance against varying number of samples)

The second experiment is done to have the performance comparision of the same selected three bandwidth estimators against varying number of samples. The results of estimated bandwidth parameter, the IMSE and the estimation time (in Seconds) versus number of samples for varying PDFs are tabulated in Table 3.2. For better interpretations, the IMSE versus log of the number of samples are plotted; for all 15 PDFs and number of samples varying from 100 to the 50000; as shown in Figure 3.2.

The IMSE performances against varying number of samples (Nsamples) for varying PDFs are similar to that discussed in Experiment 1. The ExROT performance is better for unimodal skewed, kurtotic or with outlier densities. Even it is better in some cases of multimodal skewed densities. Also, for small number of samples the ExROT performance is better compare to ROT. The convergence performance of ExROT is matching other two rules. The IMSE decreases almost inversely with increase in number of samples.

Table 3.1: Performance comparison of the bandwidth selection rules for Kernel Density Estimation (KDE) using 50000 samples. The results show the mean of the 100 trials. The boldface IMSE value show a better between the *Rule-of-Thumb* (IMSErot) and *extended Rule-of-Thumb* (IMSEexrot) rules. The time calculation is on a machine with features: Intel(R) Core(TM)2 Duo CPU, 2.93 GHz, 4.00 GB Internal RAM, 32 bit Windows 7 Professional, MATLAB R2010a

PDF	Bandwidth (h)			Integrated MSE (IMSE)*10^5			Estimation Time (in Sec)		
Type	**hrot**	**hexrot**	**heqfast**	**rot**	**exrot**	**eqfast**	**Trot**	**Texrot**	**Teqfast**
1	0.1217	0.1216	0.1213	**0.1424**	**0.1424**	0.1426	0.0005	0.0171	30.9182
2	0.0993	0.0860	0.0818	0.1770	**0.1702**	0.1702	0.0005	0.0157	35.9430
3	0.1263	0.0964	0.0200	3.7034	**2.8516**	0.4061	0.0006	0.0155	52.1223
4	0.0996	0.0693	0.0204	3.0476	**1.8294**	0.3834	0.0008	0.0154	48.2712
5	0.0402	0.0044	0.0129	1.9249	**0.7042**	0.4425	0.0006	0.0148	41.9652
6	0.1464	0.1632	0.0967	**0.1908**	0.2220	0.1504	0.0006	0.0149	41.0530
7	0.1999	0.2065	0.0925	**0.3681**	0.3897	0.1640	0.0006	0.0147	40.6588
8	0.1333	0.1596	0.0736	**0.3122**	0.4117	0.1920	0.0006	0.0153	41.7342
9	0.1552	0.1700	0.0785	**0.3438**	0.3954	0.1781	0.0005	0.0145	41.1552
10	0.1058	0.1042	0.0242	2.3709	**2.3269**	0.3392	0.0005	0.0149	48.2730
11	0.1459	0.1629	0.0895	**0.7810**	0.7961	0.7357	0.0005	0.0146	41.7426
12	0.1356	0.1350	0.0323	1.6634	**1.6586**	0.5287	0.0005	0.0156	54.3546
13	0.1450	0.1652	0.0461	**1.1161**	1.1926	0.5791	0.0005	0.0147	47.9025
14	0.2001	0.2058	0.0280	**3.2280**	3.2763	0.8810	0.0005	0.0140	51.0985
15	0.2059	0.2111	0.0230	**3.2382**	3.2824	0.4599	0.0006	0.0145	50.8321

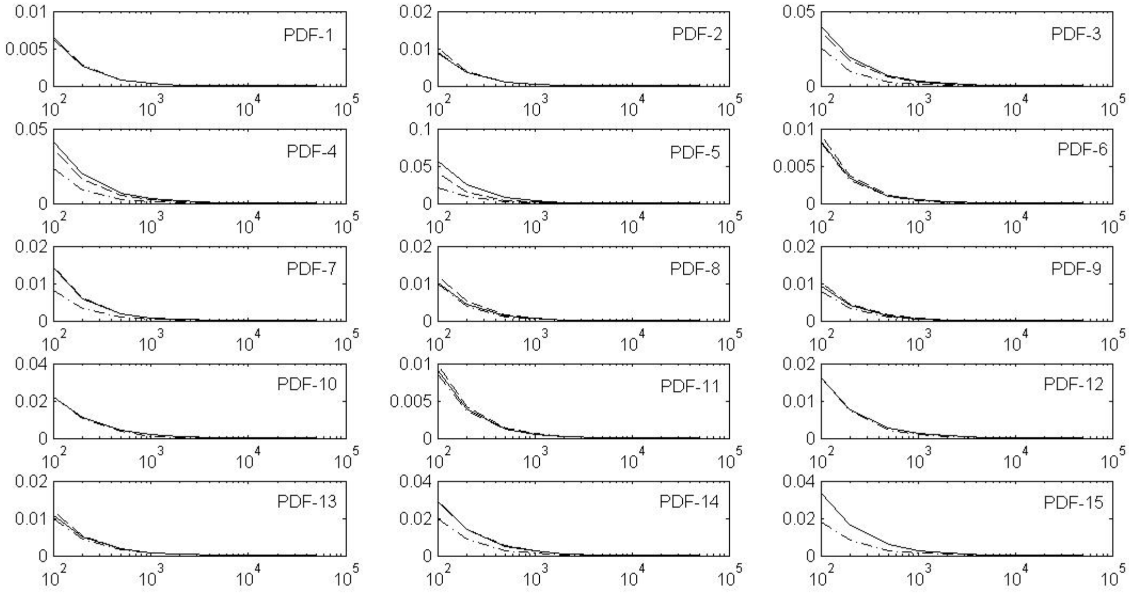

Figure 3.2: The Integrated Mean Square Error (IMSE) comparision of the bandwidth selection rules for Kernel Density Estimation (KDE) of various PDFs against varying number of samples. The solid lines (-) indicate *Rule-of-Thumb*; dashed lines (- -) indicate *extended Rule-of-Thumb*; the dash-dot lines (-.) indicate the *ϵ-exact solve-the-equation plug-in* rule

Table 3.2: Performance comparison of the bandwidth selection rules for Kernel Density Estimation (KDE) against varying number of samples. The results show the mean of the 50 trials. The time calculation is on a machine with features: Intel(R) Core(TM)2 Duo CPU, 2.93 GHz, 4.00 GB Internal RAM, 32 bit Windows 7 Professional, MATLAB R2010a

PDF	**Nsamples**	Bandwidth Parameter (h)			Integrated MSE (IMSE)*10^3			Estimation Time (in Sec)		
Type	*$10^{-}3$	**hrot**	**hexrot**	**heqfast**	**rot**	**exrot**	**eqfast**	**Trot**	**Texrot**	**Teqfast**
1	0.0100	0.4189	0.4379	0.4059	6.2430	6.5102	6.5579	0.0011	0.0004	0.0968
	0.0200	0.3613	0.3797	0.3573	2.6582	2.6653	2.7344	0.0001	0.0002	0.1210
	0.0500	0.3038	0.3099	0.3008	0.7831	0.7855	0.8064	0.0001	0.0004	0.3287
	0.1000	0.2659	0.2681	0.2602	0.3148	0.3176	0.3239	0.0001	0.0006	0.6734
	0.2000	0.2324	0.2337	0.2287	0.1253	0.1262	0.1274	0.0001	0.0010	1.3470
	0.5000	0.1929	0.1931	0.1911	0.0346	0.0346	0.0349	0.0001	0.0024	3.2634
	1.0000	0.1680	0.1678	0.1672	0.0127	0.0128	0.0128	0.0002	0.0038	6.3346
	2.0000	0.1464	0.1466	0.1455	0.0050	0.0050	0.0050	0.0002	0.0073	12.8339
2	0.0100	0.3454	0.3723	0.2944	8.9172	10.3399	9.2006	0.0001	0.0002	0.0720
	0.0200	0.2993	0.3130	0.2533	3.4302	3.8549	3.4604	0.0001	0.0002	0.1420
	0.0500	0.2465	0.2339	0.2026	0.9394	0.9924	0.9915	0.0001	0.0004	0.3852
	0.1000	0.2165	0.1870	0.1819	0.3954	0.3974	0.3945	0.0001	0.0007	0.7551
	0.2000	0.1893	0.1663	0.1565	0.1399	0.1374	0.1381	0.0001	0.0010	1.5336
	0.5000	0.1571	0.1367	0.1296	0.0425	0.0422	0.0422	0.0001	0.0023	3.7511
	1.0000	0.1370	0.1198	0.1134	0.0168	0.0162	0.0161	0.0002	0.0036	7.2034
	2.0000	0.1194	0.1038	0.0986	0.0066	0.0064	0.0064	0.0003	0.0065	14.2614
3	0.0100	0.4429	0.3498	0.1650	40.3373	36.7614	25.9410	0.0001	0.0002	0.0864
	0.0200	0.3846	0.2996	0.1188	18.8112	16.7646	9.6509	0.0001	0.0002	0.1738
	0.0500	0.3183	0.2459	0.0844	6.8916	6.0170	2.8246	0.0001	0.0004	0.4436
	0.1000	0.2775	0.2125	0.0653	3.2038	2.7464	1.0352	0.0001	0.0006	0.9069
	0.2000	0.2410	0.1845	0.0510	1.4770	1.2452	0.3778	0.0001	0.0010	1.9033
	0.5000	0.2003	0.1532	0.0384	0.5282	0.4362	0.1080	0.0001	0.0021	4.9425
	1.0000	0.1744	0.1332	0.0310	0.2397	0.1939	0.0377	0.0002	0.0032	10.3360
	2.0000	0.1518	0.1158	0.0257	0.1080	0.0856	0.0151	0.0003	0.0058	21.1275
4	0.0100	0.3500	0.2631	0.1247	41.5059	36.6378	23.6608	0.0001	0.0002	0.1076
	0.0200	0.2999	0.2169	0.0934	19.3056	16.1740	8.6166	0.0001	0.0002	0.2175
	0.0500	0.2517	0.1805	0.0686	7.0407	5.6630	2.3757	0.0001	0.0003	0.5464
	0.1000	0.2167	0.1509	0.0557	3.2218	2.4513	0.9320	0.0001	0.0006	1.0900
	0.2000	0.1899	0.1333	0.0449	1.4626	1.0710	0.3232	0.0001	0.0010	2.1309
	0.5000	0.1582	0.1102	0.0350	0.5031	0.3439	0.0875	0.0001	0.0022	4.8694
	1.0000	0.1375	0.0957	0.0296	0.2205	0.1448	0.0344	0.0002	0.0037	9.6654
	2.0000	0.1197	0.0833	0.0252	0.0951	0.0600	0.0136	0.0002	0.0058	19.3117
5	0.0100	0.1329	0.0158	0.0511	56.5031	41.3602	21.4029	0.0003	0.0003	0.1138
	0.0200	0.1215	0.0145	0.0429	25.6093	14.6056	8.5744	0.0001	0.0002	0.2072
	0.0500	0.0992	0.0110	0.0352	8.1567	4.2720	2.5379	0.0001	0.0003	0.4943
	0.1000	0.0881	0.0100	0.0297	3.4558	1.6044	0.9628	0.0001	0.0006	0.9657
	0.2000	0.0761	0.0084	0.0257	1.4268	0.6472	0.4009	0.0001	0.0010	1.7802
	0.5000	0.0640	0.0072	0.0209	0.4287	0.1696	0.1052	0.0001	0.0017	4.2263
	1.0000	0.0556	0.0062	0.0181	0.1705	0.0652	0.0397	0.0002	0.0028	8.4271
	2.0000	0.0483	0.0053	0.0156	0.0675	0.0244	0.0156	0.0002	0.0056	16.8823

Table 3.3: Performance comparison of the smoothing (bandwidth) parameter selection rules for Kernel Density Estimation (KDE) against varying number of samples. The results show the mean of the 50 trials. The time calculation is on a machine with features: Intel(R) Core(TM)2 Duo CPU, 2.93 GHz, 4.00 GB Internal RAM, 32 bit Windows 7 Professional, MATLAB R2010a

PDF	**Nsamples**	Bandwidth Parameter (h)			Integrated MSE (IMSE)*10^3			h Calculation Time (T)		
Type	*10$^-$3	**hrot**	**hexrot**	**heqfast**	**rot**	**exrot**	**eqfast**	**Trot**	**Texrot**	**Teqfast**
6	0.0100	0.5128	0.5781	0.3875	8.3749	9.1194	8.1524	0.0001	0.0002	0.0670
	0.0200	0.4421	0.4948	0.3186	3.2862	3.6643	3.0184	0.0001	0.0002	0.1404
	0.0500	0.3652	0.4096	0.2540	0.9741	1.1004	0.8728	0.0001	0.0004	0.3652
	0.1000	0.3190	0.3558	0.2254	0.4326	0.4840	0.3629	0.0001	0.0006	0.7373
	0.2000	0.2780	0.3105	0.1927	0.1644	0.1873	0.1335	0.0001	0.0010	1.5391
	0.5000	0.2315	0.2583	0.1558	0.0483	0.0550	0.0402	0.0001	0.0019	3.9894
	1.0000	0.2020	0.2251	0.1356	0.0186	0.0214	0.0145	0.0002	0.0028	8.0967
	2.0000	0.1759	0.1960	0.1174	0.0072	0.0083	0.0056	0.0003	0.0054	16.3610
7	0.0100	0.6923	0.7155	0.3566	14.2086	14.7060	8.3466	0.0001	0.0002	0.0742
	0.0200	0.5997	0.6202	0.3015	5.8889	6.1350	3.1726	0.0001	0.0002	0.1494
	0.0500	0.5005	0.5175	0.2410	1.7750	1.8618	0.8813	0.0001	0.0003	0.3908
	0.1000	0.4367	0.4512	0.2086	0.7399	0.7760	0.3704	0.0001	0.0006	0.7544
	0.2000	0.3807	0.3933	0.1800	0.2883	0.3037	0.1360	0.0001	0.0010	1.5005
	0.5000	0.3167	0.3272	0.1489	0.0854	0.0901	0.0397	0.0001	0.0018	3.8518
	1.0000	0.2759	0.2850	0.1285	0.0321	0.0340	0.0141	0.0002	0.0031	7.9127
	2.0000	0.2402	0.2481	0.1117	0.0130	0.0138	0.0058	0.0002	0.0054	16.0277
8	0.0100	0.4647	0.5979	0.3305	10.3872	11.9845	9.8866	0.0001	0.0002	0.0731
	0.0200	0.4010	0.5166	0.2932	4.3730	5.1674	3.8251	0.0001	0.0002	0.1431
	0.0500	0.3390	0.4076	0.2206	1.4163	1.6669	1.1203	0.0001	0.0004	0.4028
	0.1000	0.2907	0.3550	0.1834	0.5798	0.7038	0.4435	0.0001	0.0007	0.8379
	0.2000	0.2538	0.3065	0.1534	0.2290	0.2850	0.1614	0.0001	0.0010	1.7051
	0.5000	0.2114	0.2527	0.1225	0.0674	0.0856	0.0441	0.0001	0.0019	4.1356
	1.0000	0.1841	0.2201	0.1045	0.0268	0.0345	0.0173	0.0002	0.0030	8.2747
	2.0000	0.1602	0.1917	0.0896	0.0106	0.0139	0.0068	0.0002	0.0054	16.6340
9	0.0100	0.5406	0.5921	0.3817	9.4215	10.1084	7.9126	0.0003	0.0002	0.0782
	0.0200	0.4662	0.5133	0.3161	4.0153	4.3435	3.4017	0.0001	0.0002	0.1410
	0.0500	0.3881	0.4268	0.2522	1.3227	1.4426	1.0262	0.0001	0.0003	0.3687
	0.1000	0.3397	0.3718	0.2046	0.5546	0.6094	0.3941	0.0001	0.0007	0.7998
	0.2000	0.2952	0.3235	0.1741	0.2397	0.2638	0.1620	0.0001	0.0010	1.6285
	0.5000	0.2461	0.2696	0.1354	0.0713	0.0800	0.0426	0.0001	0.0018	4.1529
	1.0000	0.2143	0.2345	0.1148	0.0293	0.0330	0.0168	0.0002	0.0030	8.1136
	2.0000	0.1865	0.2042	0.0979	0.0119	0.0135	0.0066	0.0002	0.0055	16.2836
10	0.0100	0.3648	0.3635	0.3384	22.3775	22.4492	22.3534	0.0001	0.0002	0.0631
	0.0200	0.3172	0.3375	0.2825	10.9313	10.9990	10.8045	0.0001	0.0002	0.1406
	0.0500	0.2635	0.2995	0.1727	4.2370	4.2810	3.5900	0.0001	0.0004	0.6232
	0.1000	0.2304	0.2471	0.0805	2.0575	2.0706	0.9325	0.0001	0.0006	1.0865
	0.2000	0.2011	0.2032	0.0563	0.9881	0.9879	0.3128	0.0001	0.0010	2.0001
	0.5000	0.1675	0.1668	0.0427	0.3624	0.3609	0.0834	0.0001	0.0018	5.3371
	1.0000	0.1459	0.1454	0.0359	0.1648	0.1642	0.0329	0.0002	0.0028	10.9701
	2.0000	0.1271	0.1254	0.0303	0.0728	0.0719	0.0124	0.0002	0.0055	21.3417

Table 3.4: Performance comparison of the smoothing (bandwidth) parameter selection rules for Kernel Density Estimation (KDE) against varying number of samples. The results show the mean of the 50 trials. The time calculation is on a machine with features: Intel(R) Core(TM)2 Duo CPU, 2.93 GHz, 4.00 GB Internal RAM, 32 bit Windows 7 Professional, MATLAB R2010a

PDF	**Nsamples**	Bandwidth Parameter (h)			Integrated MSE (IMSE)*10^3			h Calculation Time (T)		
Type	*10^{-}3	**hrot**	**hexrot**	**heqfast**	**rot**	**exrot**	**eqfast**	**Trot**	**Texrot**	**Teqfast**
11	0.0100	0.5102	0.5779	0.3879	9.1443	9.9017	8.6536	0.0001	0.0002	0.0664
	0.0200	0.4393	0.4940	0.3334	3.9503	4.2723	3.6995	0.0001	0.0002	0.1355
	0.0500	0.3681	0.4133	0.2540	1.2492	1.3551	1.1489	0.0001	0.0003	0.3606
	0.1000	0.3193	0.3569	0.2210	0.5709	0.6103	0.5283	0.0001	0.0006	0.7447
	0.2000	0.2778	0.3103	0.1894	0.2513	0.2663	0.2331	0.0001	0.0010	1.5428
	0.5000	0.2311	0.2585	0.1560	0.0902	0.0941	0.0848	0.0001	0.0018	3.9834
	1.0000	0.2013	0.2250	0.1332	0.0422	0.0435	0.0399	0.0002	0.0028	8.1323
	2.0000	0.1754	0.1958	0.1140	0.0204	0.0210	0.0193	0.0003	0.0055	16.4165
12	0.0100	0.4851	0.4874	0.4530	16.3492	16.3720	16.2242	0.0001	0.0002	0.0590
	0.0200	0.4049	0.4045	0.3320	7.6083	7.6088	7.1462	0.0001	0.0002	0.1398
	0.0500	0.3403	0.3400	0.1940	2.8506	2.8496	2.2314	0.0001	0.0003	0.4242
	0.1000	0.2963	0.2950	0.1373	1.3352	1.3324	0.9125	0.0001	0.0006	0.8973
	0.2000	0.2583	0.2578	0.1070	0.6246	0.6240	0.3869	0.0001	0.0010	1.8709
	0.5000	0.2151	0.2138	0.0748	0.2231	0.2223	0.1165	0.0001	0.0018	5.2212
	1.0000	0.1871	0.1864	0.0583	0.1022	0.1020	0.0480	0.0002	0.0028	10.5090
	2.0000	0.1628	0.1620	0.0449	0.0469	0.0468	0.0189	0.0002	0.0053	21.5027
13	0.0100	0.5067	0.5846	0.3639	10.9392	11.8958	10.0277	0.0009	0.0002	0.0855
	0.0200	0.4336	0.5002	0.3073	4.8686	5.2675	4.5417	0.0001	0.0002	0.1442
	0.0500	0.3617	0.4161	0.2481	1.7277	1.8489	1.5896	0.0001	0.0004	0.3750
	0.1000	0.3177	0.3619	0.2087	0.7961	0.8408	0.7279	0.0001	0.0006	0.7774
	0.2000	0.2758	0.3146	0.1767	0.3753	0.3933	0.3341	0.0001	0.0010	1.6145
	0.5000	0.2298	0.2622	0.1320	0.1372	0.1429	0.1116	0.0001	0.0022	4.6669
	1.0000	0.2002	0.2279	0.0903	0.0652	0.0680	0.0441	0.0002	0.0033	9.4382
	2.0000	0.1742	0.1985	0.0650	0.0307	0.0323	0.0179	0.0003	0.0057	19.0233
14	0.0100	0.6986	0.7179	0.2714	29.3331	29.6279	19.9200	0.0001	0.0002	0.0831
	0.0200	0.6026	0.6199	0.2071	13.8366	13.9978	8.7139	0.0001	0.0002	0.1697
	0.0500	0.5026	0.5169	0.1452	5.1074	5.1724	2.8475	0.0001	0.0004	0.4440
	0.1000	0.4371	0.4496	0.1157	2.4004	2.4320	1.2555	0.0001	0.0006	0.9195
	0.2000	0.3807	0.3916	0.0894	1.1236	1.1389	0.5365	0.0001	0.0010	1.9072
	0.5000	0.3169	0.3260	0.0638	0.4101	0.4159	0.1757	0.0001	0.0022	4.9003
	1.0000	0.2762	0.2840	0.0497	0.1913	0.1941	0.0738	0.0002	0.0035	9.9515
	2.0000	0.2404	0.2472	0.0388	0.0889	0.0903	0.0302	0.0003	0.0055	20.1263
15	0.0100	0.7044	0.7231	0.2236	33.5149	33.7061	18.1842	0.0001	0.0002	0.0791
	0.0200	0.6215	0.6373	0.1796	16.1934	16.3224	8.1126	0.0001	0.0002	0.1713
	0.0500	0.5164	0.5295	0.1367	5.9831	6.0581	2.8509	0.0001	0.0004	0.4384
	0.1000	0.4496	0.4610	0.1143	2.7692	2.8119	1.3265	0.0001	0.0006	0.8907
	0.2000	0.3924	0.4022	0.0948	1.2618	1.2839	0.5925	0.0001	0.0010	1.8308
	0.5000	0.3262	0.3345	0.0738	0.4410	0.4493	0.1936	0.0001	0.0022	4.7281
	1.0000	0.2841	0.2913	0.0561	0.1987	0.2024	0.0721	0.0002	0.0034	9.7408
	2.0000	0.2474	0.2536	0.0347	0.0905	0.0920	0.0200	0.0002	0.0055	19.9594

3.5 Multivariate Generalized Gram-Charlier Series in Vector Notations

The Generalized Gram-Charlier (GGC) series expands an unknown *pdf* as a linear combination of the increasing order differentiations of a reference *pdf*, where the coefficients of expansion involve cumulant differences between those of an unknown *pdf* and a reference *pdf*. The GGC expansions are used to approximate *pdf* and functions of *pdf* in Statistics, Machine Learning, Economics, Chemistry, Astronomy and other application areas. There have been used Poisson's distribution (5), log-normal distribution (6), binomial distribution (103), gamma distribution (20) and others as reference *pdf*s. But, the Gram-Charlier (GC) expansion with Gaussian density as a reference *pdf* is the most popular and identified as the Gram-Charlier A (GCA) series. Specifically, the GCA series is used for test of Gaussianity or near Gaussian *pdf* approximations (6, 40, 48, 68, 130), for entropy measure and independence measure approximations (3, 65), for optima analysis through derivatives of *pdf* (17), for time-frequency analysis (31) and others. The rearrangement of the terms in GCA series results into the Edgeworth series with better convergence property.

There exists multiple ways to derive univariate GCA series, as reported by Hald (54), Hald and Steffensen (55). It's generalization to univariate GGC series is derived by Berberan-Santos (13), Cohen (31, 32), Schleher (111).

The multivariate GGC or GCA series derivation requires multivariate representations of the Taylor series, the increasing order differentiations of a reference *pdf* and the cumulants. To signify the representation issue in required multivariate extensions; it is worth quoting Terdik (125) that says, ' . . . though the generalizations to higher dimensions may be considered straightforward, the methodology and the mathematical notations get quite cumbersome when dealing with the higher order derivatives of the characteristic function or other functions of a random vector . . . '.

Conventionally, the higher order differentiations of a multivariate *pdf*; and therefore, the multivariate cumulants and multivariate Hermite polynomials; require multilinear representations. It is known and acknowledged historically that going from matrix like notations to tensor notations for multivariate cumulants and multivariate Hermite polynomials have made the representation more transparent and the calculations more simpler (81). Though the tensor notations have advantages over the matrix notations, they require componentwise separate representations and are more tedious compare to the vector notations. As a unified and comprehensive solution to this, there has been used an approach based only on elementary calculus of several variables by Terdik (125). The approach uses a specific Kronecker product based differential operator, identified as the 'K-derivative operator', to achieve vectorization of the Jacobian matrix of a multivariate vector function. The successive applications of this K-derivative operator achieves vectorization of the higher order derivatives also. Using this approach, there have been derived the multivariate

Taylor series and the higher order cumulants (105, 125); as well, the multivariate Hermite polynomials (125) in vector notations resulting into more transparent representations. In fact, it could be noticed that the same approach has been first used to derive vector Hermite polynomials by Holmquist (61); and later* it has been formalized and generalized by Terdik (125).

There exists various approaches deriving multivariate GCA series with various representations. They include GCA series representation using multi-element matrix notations for moments and cumulants by Sauer and Heydt (110); using recursive formula for Hermite polynomials by Berkowitz and Garner (14); using tensor representation for cumulants and Hermite polynomials by McCullagh (81, Chapter 5); using tensor representation for cumulants and involving multivariate Bell polynomials by Withers and Nadarajah (137); using vector moments and vector Hermite polynomials by Holmquist (61) and others. There exists various derivations for multivariate Edgeworth series (2, 39, 74, 81, 118, 137). There also exists multivariate GGC series, derived by McCullagh (81, Chapter 5), in tensor notations. But, as per the author's knowledge, there exists neither the multivariate GGC series nor the multivariate Edgeworth series in vector notations. For the ease of the readers in following and comparing the various representations; the existing representations of multivariate GGC series and multivariate GCA series are shown and compared in C.1.

Overall, to take advantages due to the recent advancement in representation, this article extends a specific derivation for univariate GGC series by Berberan-Santos (13) to multivariate; using only elementary calculus of several variables instead of Tensor calculus. As a by product, it also derives mutual relations between vector cumulants and vector moments of a random vector; integral form of the multivariate *pdf*; integral form of the multivariate vector Hermite polynomials and the multivariate GCA series. All the derived multivariate expressions are more elementary as using vector notations and more comprehensive as apparently more nearer to their counterparts for univariate; compare to their coordinatewise tensor notations. The intermediate theoretical results, in the article, are verified using suitable known examples.

Towards the aim of the article, the next Section 3.6 briefs some necessary background on the Kronecker product and a way to obtain vectorization of the higher order differentiations of a multivariate *pdf*. It also obtains the required multivariate Taylor series expansion using the derived notations. After the preliminary background, this article follows almost the same sequence for multivariate as that in (13) for univariate. The Section 3.7 uses the characteristic function and the generating functions to derive cumulants and moments of a random vector in vector notations with their mutual relationships. The Section 3.8 obtains multivariate *pdf* in terms of its vector cumulants. The expressions for derivatives of multivariate Gaussian density and vector Hermite polynomials are derived in Section 3.9. The Section 3.10 derives multivariate GCA series by representing an unknown *pdf* in terms of the Gaussian *pdf* as a reference. The Section 3.11 derives

[*] Somehow, the citation for (61) is not found in article (125).

GGC Expansion, representing an unknown *pdf* in terms of the a known reference *pdf.* The Section 3.12 derives an unknown characteristic function of a random vector in terms of a reference characteristic function. The Section 3.13 derives the same GGC expansion in a more compact way that summarizes the approach of the whole derivation. Finally, Section 3.14 concludes the article. For the sake of clarity; the calculation details, the proofs and the expressions for existing multivariate expansions are kept in appendix at the end of the article.

3.6 Vectorization of the higher order differentiations

The section briefs the Kronecker product and the way it can be applied to achieve vectorization of the higher order differentiations of a multivariate *pdf.* Based on it, the multivariate Taylor series is obtained in vector notations. More details can be found on the Kronecker Product in (80, Chapter 2), on achieving vectorization of the higher order differentiations in (105, 125) and on the commutation matrices in (80, Chapter 3, Section 7).

Definition 3.1 (Kronecker Product Operator $(\otimes)$)**.** The Kronecker Product Operator $(\otimes)$ between matrices $\mathbf{A}$ with size $p \times q$ and $\mathbf{B}$ with size $m \times n$ is defined as:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1q}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2q}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1}\mathbf{B} & a_{p2}\mathbf{B} & \cdots & a_{pq}\mathbf{B} \end{bmatrix} \tag{3.15}$$

The resultant matrix is of dimension $pm \times qn$. As a further example; let $\mathbf{A}$ is with size $p \times 1$ and $\mathbf{B}$ is with size $m \times 1$, then $\mathbf{A} \otimes \mathbf{B}'$ is[†] a matrix with size $p \times m$. $\mathbf{A} \otimes \mathbf{A}$ is symbolically represented as $\mathbf{A}^{\otimes 2}$ and has size $p^2 \times 1$. In general, $\mathbf{A} \otimes \mathbf{A} \otimes \ldots \otimes \mathbf{A}$ (n times) is symbolically represented as $\mathbf{A}^{\otimes n}$ and has size $p^n \times 1$.

Definition 3.2 (Jacobian Matrix)**.** Let $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_d)'$, $\boldsymbol{\lambda} \in \mathbb{R}^d$ and $\mathbf{f}(\boldsymbol{\lambda}) = (f_1(\lambda), f_2(\lambda), \ldots, f_m(\lambda))' \in \mathbb{R}^m$ be a differentiable m-component vector function. Then, Jacobian

[†] The symbol ' stands for Transpose of a matrix

matrix of $\mathbf{f}(\boldsymbol{\lambda})$ $(\mathbf{J}(\mathbf{f}))$ is an $m \times d$ matrix defined as under:

$$\mathbf{J}(\mathbf{f}(\boldsymbol{\lambda})) = \frac{d\mathbf{f}}{d\boldsymbol{\lambda}} = \left[\frac{\partial \mathbf{f}}{\partial \lambda_1}, \frac{\partial \mathbf{f}}{\partial \lambda_2}, \dots, \frac{\partial \mathbf{f}}{\partial \lambda_d}\right] = \begin{bmatrix} \frac{\partial f_1}{\partial \lambda_1} & \frac{\partial f_1}{\partial \lambda_2} & \cdots & \frac{\partial f_1}{\partial \lambda_d} \\ \frac{\partial f_2}{\partial \lambda_1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_m}{\partial \lambda_1} & \frac{\partial f_m}{\partial \lambda_2} & \cdots & \frac{\partial f_m}{\partial \lambda_d} \end{bmatrix} \tag{3.16}$$

Let the vector differential operator be defined as a column vector $\mathbf{D}_{\boldsymbol{\lambda}} = \left(\frac{\partial}{\partial \lambda_1}, \frac{\partial}{\partial \lambda_2}, \dots, \frac{\partial}{\partial \lambda_d}\right)'$, then the Jacobian matrix, in terms of the $\mathbf{D}_{\boldsymbol{\lambda}}$, can be re-written as:

$$\mathbf{J}(\mathbf{f}(\boldsymbol{\lambda})) = \mathbf{D}_{\boldsymbol{\lambda}}(\mathbf{f}) = \mathbf{f}(\boldsymbol{\lambda})\mathbf{D}'_{\boldsymbol{\lambda}} = (f_1(\boldsymbol{\lambda}), f_2(\boldsymbol{\lambda}), \dots, f_m(\boldsymbol{\lambda}))' \left(\frac{\partial}{\partial \lambda_1}, \frac{\partial}{\partial \lambda_2}, \dots, \frac{\partial}{\partial \lambda_d}\right) \tag{3.17}$$

This implies that to match the definition of differentiation from matrix calculus, the vector differential operator should be applied from the right to the left. This is same as the requirement to be satisfied on generalization of vector derivative to matrix derivative as discussed by Magnus (79). So, applying vector derivative operator from right to the left, has been kept as a rule throughout the chapter.

Definition 3.3 (The K-derivative Operator). Let $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_d)'$, $\boldsymbol{\lambda} \in \mathbb{R}^d$; the vector differential operator $\mathbf{D}_{\boldsymbol{\lambda}} = \left(\frac{\partial}{\partial \lambda_1}, \frac{\partial}{\partial \lambda_2}, \dots, \frac{\partial}{\partial \lambda_d}\right)'$ and a differentiable m-component vector function $\mathbf{f}(\boldsymbol{\lambda}) = (f_1(\lambda), f_2(\lambda), \dots, f_m(\lambda))' \in \mathbb{R}^m$. Then, the Kronecker product between $\mathbf{D}_{\boldsymbol{\lambda}}$ and $\mathbf{f}(\boldsymbol{\lambda})$ is given as under:

$$\mathbf{D}^{\otimes}_{\boldsymbol{\lambda}}\mathbf{f}(\boldsymbol{\lambda}) = \begin{bmatrix} f_1(\boldsymbol{\lambda}) \\ f_2(\boldsymbol{\lambda}) \\ \vdots \\ f_m(\boldsymbol{\lambda}) \end{bmatrix} \otimes \begin{bmatrix} \frac{\partial}{\partial \lambda_1} \\ \frac{\partial}{\partial \lambda_2} \\ \vdots \\ \frac{\partial}{\partial \lambda_d} \end{bmatrix} = Vec \begin{bmatrix} \frac{\partial f_1}{\partial \lambda_1} & \frac{\partial f_1}{\partial \lambda_2} & \cdots & \frac{\partial f_1}{\partial \lambda_d} \\ \frac{\partial f_2}{\partial \lambda_1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_m}{\partial \lambda_1} & \frac{\partial f_m}{\partial \lambda_2} & \cdots & \frac{\partial f_m}{\partial \lambda_d} \end{bmatrix}' \tag{3.18}$$

$$\Rightarrow \mathbf{D}^{\otimes}_{\boldsymbol{\lambda}}\mathbf{f}(\boldsymbol{\lambda}) = Vec\left(\frac{\partial \mathbf{f}}{\partial \boldsymbol{\lambda}'}\right)' = Vec\left(\frac{\partial}{\partial \boldsymbol{\lambda}}\mathbf{f}'\right) \tag{3.19}$$

where, the Vec operator converts $m \times d$ matrix into an $md \times 1$ column vector by stacking the columns one after an other. The operator $\mathbf{D}^{\otimes}_{\boldsymbol{\lambda}}$ is called Kronecker derivative operator or simply, K-derivative operator.

Thus, the Kronecker product with the vector differential operator, obtains vectorization of the transposed Jacobian of a vector function. Corresponding to the definition, the k^{th} order differentiation is given by:

$$\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k}\mathbf{f} = \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\left(\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k-1}\mathbf{f}\right) = [f_1(\lambda), f_2(\lambda), \ldots, f_m(\lambda)]' \otimes \left[\frac{\partial}{\partial\lambda_1}, \frac{\partial}{\partial\lambda_2}, \cdots, \frac{\partial}{\partial\lambda_d}\right]^{'\otimes k} \tag{3.20}$$

The $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k}\mathbf{f}$ is a column vector of dimension $md^k \times 1$. Some important properties of the K-derivative operator, those are useful in the further derivations, are listed in Appendix C.2.

3.6.1 Application of the K-derivative operator to the multivariate Taylor series

Let $\mathbf{x} = (X_1, X_2, ..., X_d)'$ be a d-dimensional column vector and $f(\mathbf{x})$ be the function of several variables differentiable in each variable. Using the defined K-derivative operator, the Taylor series for $f(\mathbf{x})$, expanding it at origin, is given as:

$$f(\mathbf{x}) = \sum_{m=0}^{m=\infty} \frac{1}{m!}\mathbf{c}(m,d)'\mathbf{x}^{\otimes m} \tag{3.21}$$

where, $\mathbf{c}(m,d)$ is the vector of dimension $d^m \times 1$ and given in terms of the derivative vector $\mathbf{D}_{\mathbf{x}} = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \cdots, \frac{\partial}{\partial x_d}\right)'$ as

$$\mathbf{c}(m,d) = \left(\mathbf{D}_{\mathbf{x}}^{\otimes m} f(\mathbf{x})\right)\Big|_{\mathbf{x}=\mathbf{0}} \tag{3.22}$$

The Taylor series expansion near $\boldsymbol{\lambda} = \mathbf{0}$, called the Maclaurian series, of some required functions based on the Equation (3.21) are derived in appendix C.3.

3.7 Moments, cumulants and characteristic function of a random vector

Let $\mathbf{x} = (X_1, X_2, ..., X_d)'$ be a d-dimensional random vector and $f(\mathbf{x})$ be its joint *pdf* differentiable in each variable.

The Characteristic function ($\mathcal{F}$) of $\mathbf{x}$ is defined as the expected value of $\mathrm{e}^{i\mathbf{x}'\boldsymbol{\lambda}}$, where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_d)'$, $\boldsymbol{\lambda} \in \mathbb{R}^d$. Also, both the characteristic function and the *pdf* are the Fourier Transform (F) of each other, in the sense they are dual.

$$\mathcal{F}_{\mathbf{x}}(\boldsymbol{\lambda}) = E\left\{\mathrm{e}^{i(\mathbf{x}'\boldsymbol{\lambda})}\right\} = \mathsf{F}(f(\mathbf{x})) = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} f(\mathbf{x})\mathrm{e}^{i(\mathbf{x}'\boldsymbol{\lambda})}d\mathbf{x} \tag{3.23}$$

Expanding $\mathrm{e}^{i\mathbf{x}'\boldsymbol{\lambda}}$ using its Maclaurian series in Equation (C.16) in appendix C.3, we get:

$$\mathcal{F}_{\mathbf{x}}(\boldsymbol{\lambda}) = \sum_{k=0}^{\infty} \mathbf{m}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!} \tag{3.24}$$

where, $\mathbf{m}(k,d)$ is the k^{th} order moment vector of dimension $d^k \times 1$ and given by

$$\mathbf{m}(k,d) = \int_{\mathbb{R}^d} \mathbf{x}^{\otimes k} f(\mathbf{x}) d\mathbf{x}$$

$$\text{Also, } f(\mathbf{x}) = \mathsf{F}^{-1}(\mathcal{F}(\lambda)) = \mathsf{F}^{-1}\left(\sum_{k=0}^{\infty} \mathbf{m}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right) \tag{3.25}$$

$$= \sum_{k=0}^{\infty} \frac{\mathbf{m}(k,d)'}{k!} \left(\frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} (i\boldsymbol{\lambda})^{\otimes k} \mathrm{e}^{-i\mathbf{x}'\boldsymbol{\lambda}} d\boldsymbol{\lambda}\right)$$

$$= \sum_{k=0}^{\infty} (-1)^k \frac{\mathbf{m}(k,d)'}{k!} \mathbf{D}^{(k)} \delta(\mathbf{x}) \tag{3.26}$$

$$(\because \text{Proof in Appendix C.4.1})$$

The Moment Generating Function (MGF) of $f(\mathbf{X})$ is given as

$$\mathbf{M}(\boldsymbol{\lambda}) = E\left\{\mathrm{e}^{\mathbf{x}'\boldsymbol{\lambda}}\right\} = \int_{\mathbb{R}^d} f(\mathbf{X}) \mathrm{e}^{\mathbf{x}'\boldsymbol{\lambda}} d\mathbf{X} \tag{3.27}$$

$$= \sum_{k=0}^{\infty} \mathbf{m}(k,d)' \frac{\boldsymbol{\lambda}^{\otimes k}}{k!} \quad (\because \text{Expanding } \mathrm{e}^{\mathbf{x}'\boldsymbol{\lambda}}) \tag{3.28}$$

Assuming $\mathbf{M}(\boldsymbol{\lambda})$ and $\mathcal{F}(\boldsymbol{\lambda})$ are expanded using Taylor series,

$$\mathbf{m}(k,d) = \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k} \mathbf{M}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}} = (-i)^k \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k} \mathcal{F}_{\mathbf{x}}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}} \tag{3.29}$$

The Cumulant Generating Function (CGF) of $f(\mathbf{X})$ is given by,

$$\mathbf{C}(\boldsymbol{\lambda}) = \ln \mathbf{M}(\boldsymbol{\lambda}) = \sum_{k=1}^{\infty} \mathbf{c}(k,d)' \frac{\boldsymbol{\lambda}^{\otimes k}}{k!} \tag{3.30}$$

where, $\mathbf{c}(k,d)$ is the k^{th} order cumulant vector of dimension $d^k \times 1$.
The Cumulant Generating Function (CGF) of $f(\mathbf{X})$ can also be defined using the characteristic function, as under:

$$\mathcal{C}(\boldsymbol{\lambda}) = \ln \mathcal{F}(\boldsymbol{\lambda}) = \sum_{k=1}^{\infty} \mathbf{c}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!} \tag{3.31}$$

Assuming $\mathbf{C}(\boldsymbol{\lambda})$ and $\mathcal{C}(\boldsymbol{\lambda})$ have been expanded using Taylor series,

$$\mathbf{c}(k,d) = \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k}\mathbf{C}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}} = (-i)^k \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes k}\mathcal{C}_{\mathbf{x}}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}} \tag{3.32}$$

Significantly, this section has derived the moments and the cumulants of a random vector in vector notations.

3.7.1 Relation between the cumulant vectors and the moment vectors

The relation between the moments and the cumulants is given by combining Equation (3.28) and Equation (3.30) as below:

$$\mathbf{M}(\boldsymbol{\lambda}) = \sum_{k=0}^{\infty} \mathbf{m}(k,d)' \frac{\boldsymbol{\lambda}^{\otimes k}}{k!} = \exp\left(\sum_{k=1}^{\infty} \mathbf{c}(k,d)' \frac{\boldsymbol{\lambda}^{\otimes k}}{k!}\right) \tag{3.33}$$

For $k = 1$, using Equation (3.29), we get:

$$\mathbf{m}(1,d) = \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes 1}\mathbf{M}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}}$$

Applying K-derivative $(\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes})$ to Equation (3.33),

$$\begin{aligned}\sum_{p=1}^{\infty} \mathbf{m}(p,d)' \frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)} \otimes \mathbf{I}_d\right)}{(p-1)!}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}} &= \sum_{p=1}^{\infty} \mathbf{c}(p,d)' \frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)} \otimes \mathbf{I}_d\right)}{(p-1)!} \\ &\quad \otimes \exp\left(\sum_{q=1}^{\infty} \mathbf{c}(q,d)' \frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}} \\ \Rightarrow \mathbf{m}(1,d) &= \mathbf{c}(1,d)\end{aligned}$$

Similarly, based on Equation (3.29), for any k taking k^{th} order K-derivative of above Equation (3.33) on both sides relates $\mathbf{m}(k,d)$ with $\mathbf{c}(k,d)$. For example, the cases for $k = 2$ and $k = 3$ are shown in Appendix C.4.2. Overall, the vector moments in terms of the vector cumulants can be summarized as under:

$$\begin{aligned}
\mathbf{m}(1,d) &= \mathbf{c}(1,d)\\
\mathbf{m}(2,d) &= \mathbf{c}(2,d) + \mathbf{c}(1,d)^{\otimes 2}\\
\mathbf{m}(3,d) &= \mathbf{c}(3,d) + 3\mathbf{c}(2,d)\otimes\mathbf{c}(1,d) + \mathbf{c}(1,d)^{\otimes 3}\\
\mathbf{m}(4,d) &= \mathbf{c}(4,d) + 4\mathbf{c}(3,d)\otimes\mathbf{c}(1,d) + 3\mathbf{c}(2,d)^{\otimes 2} + 6\mathbf{c}(2,d)\\
&\quad \otimes\mathbf{c}(1,d)^{\otimes 2} + \mathbf{c}(1,d)^{\otimes 4}\\
\mathbf{m}(5,d) &= \mathbf{c}(5,d) + 5\mathbf{c}(4,d)\otimes\mathbf{c}(1,d) + 10\mathbf{c}(3,d)\otimes\mathbf{c}(2,d)\\
&\quad + 10\mathbf{c}(3,d)\otimes\mathbf{c}(1,d)^{\otimes 2} + 15\mathbf{c}(2,d)^{\otimes 2}\otimes\mathbf{c}(1,d)\\
&\quad + 10\mathbf{c}(2,d)\otimes\mathbf{c}(1,d)^{\otimes 3} + \mathbf{c}(1,d)^{\otimes 5}\\
\mathbf{m}(6,d) &= \mathbf{c}(6,d) + 6\mathbf{c}(5,d)\otimes\mathbf{c}(1,d) + 15\mathbf{c}(4,d)\otimes\mathbf{c}(2,d)\\
&\quad + 15\mathbf{c}(4,d)\otimes\mathbf{c}(1,d)^{\otimes 2} + 10\mathbf{c}(3,d)^{\otimes 2} + 60\mathbf{c}(3,d)\otimes\mathbf{c}(2,d)\\
&\quad \otimes\mathbf{c}(1,d) + 20\mathbf{c}(3,d)\otimes\mathbf{c}(1,d)^{\otimes 2} + 15\mathbf{c}(2,d)^{\otimes 3}\\
&\quad + 45\mathbf{c}(2,d)^{\otimes 2}\otimes\mathbf{c}(1,d)^{\otimes 2} + 15\mathbf{c}(2,d)\otimes\mathbf{c}(1,d)^{\otimes 4} + \mathbf{c}(1,d)^{\otimes 6}
\end{aligned} \tag{3.34}$$

The above set of equations can be represented through more compact formulas as under. The Equation (3.35) gives generalized k^{th} order d-variate cumulant vector in terms of the moment vectors and the Equation (3.36) gives the vice-a-versa.

$$\mathbf{m}(k,d) = \sum_{p=0}^{k-1}\binom{k-1}{p}\mathbf{K}^{-1}_{\mathfrak{p}h\leftrightarrow l}\mathbf{c}(k-p,d)\otimes\mathbf{m}(p,d) \tag{3.35}$$

$$\mathbf{c}(k,d) = \mathbf{m}(k,d) - \sum_{p=1}^{k-1}\binom{k-1}{p}\mathbf{K}^{-1}_{\mathfrak{p}h\leftrightarrow l}\mathbf{c}(k-p,d)\otimes\mathbf{m}(p,d) \tag{3.36}$$

where, $\mathbf{K}^{-1}_{\mathfrak{p}h\leftrightarrow l}$ is a specific commutation matrix with corresponding dimensions that changes the place of the cumulants for Kronecker product such that the expression has decreasing order cumulants from left to the right, i.e. the higher order cumulant vector on the left and the lower order cumulant vector on the right. As the Kronecker products are non-commutative, without using the commutation matrices it would have been impossible to derive the compact formula. The derived multivariate expressions reduce to the following expressions for dimension $d = 1$ and are exactly same as those derived in (13).

$$\begin{aligned} c(k,1) &= m(k,1) - \sum_{p=1}^{k-1} \binom{k-1}{p} c(k-p,1) m(p,1) \\ \text{or more simply, } c_k &= m_k - \sum_{p=1}^{k-1} \binom{k-1}{p} c_{k-p} m_p \end{aligned} \tag{3.37}$$

Thus, the derived multivariate expressions in Equation (3.34) are elementary vector extensions to those for univariate.

3.8 Multivariate *pdf* expressed in terms of the cumulants

From Equation (3.25) and Equation (3.31), the multivariate *pdf* $f(\mathbf{x})$ can be written as:

$$f(\mathbf{x}) = \mathsf{F}^{-1}(\mathrm{e}^{C(\boldsymbol{\lambda})}) = \left(\frac{1}{2\pi}\right)^d \int_{\mathbb{R}^d} \exp\left(\sum_{k=1}^{\infty} \mathbf{c}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right) \exp\left(-i\mathbf{x}'\boldsymbol{\lambda}\right) d\boldsymbol{\lambda} \tag{3.38}$$

As *pdf* is a real function and Re$(e^{A+iB}e^{-iC}) = e^A \cos(B-C) = e^A \cos(C-B)$, the Equation (3.38) can be re-written as:

$$\begin{aligned} f(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} & \exp\left(\sum_{k=1}^{\infty} \frac{\mathbf{c}(2k,d)'}{2k!} (i\boldsymbol{\lambda})^{\otimes 2k}\right) \\ & \cos\left(\mathbf{x}'\boldsymbol{\lambda} + \sum_{k=1}^{\infty} \frac{\mathbf{c}(2k-1,d)'}{(2k-1)!} (i)^{2k} (\boldsymbol{\lambda})^{\otimes(2k-1)}\right) d\boldsymbol{\lambda} \end{aligned} \tag{3.39}$$

The integrand in this equation is an even function. So,

$$\begin{aligned} f(\mathbf{x}) = \frac{1}{(\pi)^d} \int_{(\mathbb{R}+)^d} & \exp\left(\sum_{k=1}^{\infty} \frac{\mathbf{c}(2k,d)'}{2k!} (i\boldsymbol{\lambda})^{\otimes 2k}\right) \\ & \cos\left(\mathbf{x}'\boldsymbol{\lambda} + \sum_{k=1}^{\infty} \frac{\mathbf{c}(2k-1,d)'}{(2k-1)!} (i)^{2k} (\boldsymbol{\lambda})^{\otimes(2k-1)}\right) d\boldsymbol{\lambda} \end{aligned} \tag{3.40}$$

where, $\mathbb{R}^+ = \{x \in \mathbb{R} : x \geq 0\}$. The Equation (3.39) and the Equation (3.40) give a multivariate *pdf* in terms of the cumulants. As they are derived using Taylor series expansion, the infinite order differentiability is an implicit assumption.

The equations can be verified using known *pdf* examples with finite number of moments and cumulants. Let say, the impulse delta density function has only the first order cumulant being non-zero and all other higher order cumulants are zero. Using this knowledge in Equation (3.39),

$$\begin{aligned} f(\mathbf{x}) &= \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \cos\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda} \right) d\boldsymbol{\lambda} \qquad (3.41) \\ &= \delta(\mathbf{x} - \mathbf{c}(1,d)) \quad \text{(shifted impulse delta function)} \end{aligned}$$

Let's take another example, the Gaussian density function has first two order cumulants nonzero and all other order cumulants are zero. Using this knowledge in Equation (3.39),

$$\begin{aligned} f(\mathbf{x}) &= \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2} \right) \cos\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda} \right) d\boldsymbol{\lambda} \\ &= G(\mathbf{x}) \quad \text{(See Appendix C.4.3 for proof)} \end{aligned} \qquad (3.42)$$

3.9 The multivariate Hermite polynomials in integral form

An interesting application of the integral form of multivariate *pdf* representation is achieved in this section. The multivariate Gaussian expressed as in Equation (3.42) is used to derive it's differentials and Hermite polynomials in a simple way. Taking k^{th} K-derivative of $G(\mathbf{x})$,

$$\begin{aligned} G^{(k)}(\mathbf{x}) := \mathbf{D}_{\mathbf{x}}^{\otimes k} G(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \boldsymbol{\lambda}^{\otimes k} \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2} \right) \\ \cos\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda} + \frac{k\pi}{2} \right) d\boldsymbol{\lambda} \end{aligned} \qquad (3.43)$$

The multivariate Hermite polynomials defined by Holmquist (61) are defined as under:

$$\mathbf{H}_k(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) = [G(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}})]^{-1} (-1)^k \left(\mathbf{C}_{\mathbf{x}} \mathbf{D}_x \right)^{\otimes k} G(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) \qquad (3.44)$$

$$\begin{aligned} \text{where, } G(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) &= |\mathbf{C}_{\mathbf{x}}|^{-1/2} (2\pi)^{-d/2} \exp\left(-\frac{1}{2} \mathbf{x}' \mathbf{C}_{\mathbf{x}}^{-1} \mathbf{x} \right) \\ &= |\mathbf{C}_{\mathbf{x}}|^{-1/2} (2\pi)^{-d/2} \exp\left(-\frac{1}{2} \left(Vec\, \mathbf{C}_{\mathbf{x}}^{-1} \right)' \mathbf{x}^{\otimes 2} \right) \qquad (3.45) \end{aligned}$$

This is equivalent to the 1-dimensional definition of Hermite polynomials[‡] by Rodrigues's formula in Equation (3.46), except the introduction of matrix $\mathbf{C}_{\mathbf{x}}$.

$$H_k(x) = [G(x)]^{-1} (-1)^k \frac{d^k}{dx^k} G(x) \text{ where, } G(x) = \frac{1}{2\pi} \mathrm{e}^{-\frac{1}{2}x^2} \qquad (3.46)$$

[‡] This is the 'probabilists' Hermite polynomials and not the 'physicists' Hermite polynomials used by Berberan-Santos (13).

Using Equation (3.43), the Equation (3.44) for multivariate Hermite polynomials is rewritten as:

$$\mathbf{H}_k(\mathbf{x};\mathbf{0},\mathbf{C}_\mathbf{x}) = (2\pi)^{d/2}\left|\mathbf{C}_\mathbf{x}\right|^{1/2}(-1)^k\left(\mathbf{C}_\mathbf{x}\right)^{\otimes k}\exp\left(\frac{1}{2}\left(Vec\,\mathbf{C}_\mathbf{x}^{-1}\right)'\mathbf{x}^{\otimes 2}\right)\frac{1}{(2\pi)^d}$$
$$\int_{\mathbb{R}^d}\boldsymbol{\lambda}^{\otimes k}\exp\left(-\frac{\mathbf{c}(2,d)'}{2}\boldsymbol{\lambda}^{\otimes 2}\right)\cos\left((\mathbf{x}-\mathbf{c}(1,d))'\boldsymbol{\lambda}+\frac{k\pi}{2}\right)d\boldsymbol{\lambda} \tag{3.47}$$

Taking $\mathbf{C}_\mathbf{x} = \mathbf{I}_d$, where $\mathbf{I}_d$ is $d \times d$ identity matrix; using the property $(Vec\,\mathbf{I}_d)'\mathbf{x}^{\otimes 2} = \mathbf{x}'\mathbf{x}$ and using change of variable as $\boldsymbol{\lambda}/\sqrt{2} = \mathbf{u}$ - the integral form of multivariate Hermite polynomials is obtained as under:

$$\mathbf{H}_k(\mathbf{x};\mathbf{0},\mathbf{I}_d) = (2)^{\frac{k+d+1}{2}}(\pi)^{-d/2}\exp\left(\frac{1}{2}\mathbf{x}'\mathbf{x}\right)$$
$$\int_{\mathbb{R}^d}\mathbf{u}^{\otimes k}\exp\left(-\mathbf{u}'\mathbf{u}\right)\cos\left(\sqrt{2}\mathbf{x}'\mathbf{u}-\frac{k\pi}{2}\right)d\mathbf{u} \tag{3.48}$$

The result in Equation (3.43) can also be obtained using Equation (3.42) and applying the derivative property of Fourier transform (F). The k^{th} derivative of $G(x)$ is given by,

$$G^{(k)}(\mathbf{x}) = \mathsf{F}^{-1}\left(\mathsf{F}(G^{(k)}(\mathbf{x}))\right) \tag{3.49}$$
$$= \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d}(i\boldsymbol{\lambda})^{\otimes k}\,\mathsf{F}(G(\mathbf{x}))\exp^{-i\mathbf{x}'\boldsymbol{\lambda}}d\boldsymbol{\lambda} \tag{3.50}$$
$$= \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d}\boldsymbol{\lambda}^{\otimes k}\exp\left(-\frac{\mathbf{c}(2,d)'}{2}\boldsymbol{\lambda}^{\otimes 2}\right)\cos\left((\mathbf{x}-\mathbf{c}(1,d))'\boldsymbol{\lambda}+\frac{k\pi}{2}\right)d\boldsymbol{\lambda} \tag{3.51}$$

3.10 Multivariate Gram-Charlier A series

Till now, the chapter has derived - an unknown *pdf* expressed in terms of its cumulants in Equation (3.39); the Gaussian density function expressed in terms of its cumulants in Equation (3.42) and the Hermite polynomials in Equation (3.48). Based on them, the multivariate Gram Charlier A series that expresses an unknown *pdf* using Gaussian density as a reference can be obtained. The expansion assumes first and second order cumulants being same for both the unknown *pdf* and the reference *pdf*. Using the expansion $\exp(A+B)\cos(C+D) = \exp(A)\exp(B)(\cos C\cos D -$

$\sin C \sin D)$, the Equation (3.39) can be re-written as:

$$\begin{aligned} f(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \exp\left(-\frac{\mathbf{c}(2,d)'}{2}\boldsymbol{\lambda}^{\otimes 2}\right) \exp\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right) \\ \left\{\cos((\mathbf{x}-\mathbf{c}(1,d))'\boldsymbol{\lambda}) \cos\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)\right. \\ \left.- \sin((\mathbf{x}-\mathbf{c}(1,d))'\boldsymbol{\lambda}) \sin\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)\right\} d\boldsymbol{\lambda} \qquad (3.52) \end{aligned}$$

Using the expansions in Appendix C.3, parts of the Equation (3.52) can be approximated upto maximum 6^{th}-order statistics as under:

$$\begin{aligned} \exp\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right) \cos\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right) \\ = 1 + \left(\frac{\mathbf{c}(4,d)'\boldsymbol{\lambda}^{\otimes 4}}{4!} - \frac{\mathbf{c}(6,d)'\boldsymbol{\lambda}^{\otimes 6}}{6!}\right) - \frac{1}{2}\left(\frac{\mathbf{c}(3,d)'\boldsymbol{\lambda}^{\otimes 3}}{3!}\right)^{\otimes 2} + \ldots \qquad (3.53) \\ \exp\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right) \sin\left(\sum_{k=2}^{\infty} \frac{\mathbf{c}(2k-1,d)'}{(2k-1)!}(-i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right) \\ = \frac{\mathbf{c}(3,d)'\boldsymbol{\lambda}^{\otimes 3}}{3!} - \frac{\mathbf{c}(5,d)'\boldsymbol{\lambda}^{\otimes 5}}{5!} + \ldots \qquad (3.54) \end{aligned}$$

Using above Equation (3.53) and Equation (3.54), the Equation (3.52) can be re-written as:

$$\begin{aligned} f(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \left\{\exp\left(-\frac{\mathbf{c}(2,d)'}{2}\boldsymbol{\lambda}^{\otimes 2}\right) \cos((\mathbf{x}-\mathbf{c}(1,d))'\boldsymbol{\lambda})\right. \\ \left.\left(1 + \frac{\mathbf{c}(4,d)'\boldsymbol{\lambda}^{\otimes 4}}{4!} - \frac{1}{6!}\left(\mathbf{c}(6,d) - 10\mathbf{c}(3,d)^{\otimes 2}\right)'\boldsymbol{\lambda}^{\otimes 6} + \ldots\right)\right\} d\boldsymbol{\lambda} \\ - \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \left\{\exp\left(-\frac{\mathbf{c}(2,d)'}{2}\boldsymbol{\lambda}^{\otimes 2}\right) \cos\left((\mathbf{x}-\mathbf{c}(1,d))'\boldsymbol{\lambda} - \frac{\pi}{2}\right)\right. \\ \left.\left(\frac{\mathbf{c}(3,d)'\boldsymbol{\lambda}^{\otimes 3}}{3!} - \frac{\mathbf{c}(5,d)'\boldsymbol{\lambda}^{\otimes 5}}{5!} + \ldots\right)\right\} d\boldsymbol{\lambda} \qquad (3.55) \end{aligned}$$

Using the Equation (3.43) for derivatives of Gaussian defined, the above Equation can be simplified as:

$$\begin{aligned} f(\mathbf{x}) = G(\mathbf{x}) - \frac{\mathbf{c}(3,d)'}{3!}G^{(3)}(\mathbf{x}) + \frac{\mathbf{c}(4,d)'}{4!}G^{(4)}(\mathbf{x}) - \frac{\mathbf{c}(5,d)'}{5!}G^{(5)}(\mathbf{x}) \\ + \frac{\mathbf{c}(6,d)' + 10\mathbf{c}(3,d)^{\otimes 2'}}{6!}G^{(6)}(\mathbf{x}) + \ldots \qquad (3.56) \end{aligned}$$

The Equation (3.56) is the Gram-Charlier A series expressed directly in terms of the cumulants and the derivatives of the Gaussian *pdf*. Usually, the GCA is represented in terms of the Hermite polynomials. So, the GCA expansion (Equation (3.56)) in terms of the Hermite polynomials; either using definition in Equation (3.44) or using $\mathbf{H}_k(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}^{-1})$ derived in Equation (3.47); can be re-written as:

$$
\begin{aligned}
f(\mathbf{x}) = G(\mathbf{x}) \Bigg[1 + \frac{\mathbf{c}(3,d)'}{3!}\left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 3}\mathbf{H}_3(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) + \frac{\mathbf{c}(4,d)'}{4!}\left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 4}\mathbf{H}_4(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) \\
+ \frac{\mathbf{c}(5,d)'}{5!}\left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 5}\mathbf{H}_5(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) + \frac{\mathbf{c}(6,d)' + 10\mathbf{c}(3,d)^{\otimes 2'}}{6!}\left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 6} \\
\mathbf{H}_6(\mathbf{x}; \mathbf{0}, \mathbf{C}_{\mathbf{x}}) + \ldots] \qquad (3.57)
\end{aligned}
$$

$$
\begin{aligned}
f(\mathbf{x}) = G(\mathbf{x}) \Bigg[1 + \frac{\mathbf{c}(3,d)'}{3!}\mathbf{H}_3(\mathbf{x}; \mathbf{0}, \mathbf{I}_d) + \frac{\mathbf{c}(4,d)'}{4!}\mathbf{H}_4(\mathbf{x}; \mathbf{0}, \mathbf{I}_d) + \frac{\mathbf{c}(5,d)'}{5!}\mathbf{H}_5(\mathbf{x}; \mathbf{0}, \mathbf{I}_d) \\
+ \frac{\mathbf{c}(6,d)' + 10\mathbf{c}(3,d)^{\otimes 2'}}{6!}\mathbf{H}_6(\mathbf{x}; \mathbf{0}, \mathbf{I}_d) + \ldots \Bigg] \qquad (3.58)
\end{aligned}
$$

Finally, the GCA series, in vector notations, can be expressed either using k^{th} order derivative of Gaussian $(G_k(\mathbf{x}))$ or using k^{th} order vector Hermite polynomials $(\mathbf{H}_k(\mathbf{x}))$ as under:

$$
\begin{aligned}
f(\mathbf{x}) &= \sum_{k=0}^{\infty}(-1)^k \frac{\mathbf{c}(k,d)'}{k!} G^{(k)}(\mathbf{x}) \qquad (3.59) \\
&= \sum_{k=0}^{\infty} \frac{\mathbf{c}(k,d)'}{k!}\mathbf{H}_k(\mathbf{x}; \mathbf{0}, \mathbf{I}_d) \qquad (3.60)
\end{aligned}
$$

3.11 Multivariate Generalized Gram-Charlier series

To derive the generalized Gram-Charlier series, an unknown *pdf* $f(\mathbf{x})$ need be represented in terms of any known reference *pdf* $\psi(\mathbf{x})$, where both the *pdf*s are represented in terms of their cumulants. Let the k^{th} order cumulant vector of the reference *pdf* $\psi(\mathbf{x})$ be $\mathbf{c}_r(k,d)$. Then, the k^{th} order cumulant difference vector $\boldsymbol{\delta}(k,d)$ is: $\boldsymbol{\delta}(k,d) = \mathbf{c}(k,d) - \mathbf{c}_r(k,d), \forall k$. Using $\boldsymbol{\delta}(k,d)$, the

Equation (3.39) can be re-written as under:

$$\begin{aligned} f(\mathbf{x}) &= \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d} \exp\left(\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)\exp\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right) \\ &\quad \cos\left(\mathbf{x}'\boldsymbol{\lambda}+\left(\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k-1,d)}{(2k-1)!}+\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k-1,d)}{(2k-1)!}\right)'(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)d\boldsymbol{\lambda} \qquad (3.61) \\ &= \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d} \exp\left(\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)\exp\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right) \\ &\quad \left\{\cos\left(\mathbf{x}'\boldsymbol{\lambda}+\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)\cos\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k-1,d)'}{(2k-1)!}\right.\right. \\ &\quad \left.(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right) - \sin\left(\mathbf{x}'\boldsymbol{\lambda}+\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right) \\ &\quad \left.\sin\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)\right\}d\boldsymbol{\lambda} \qquad (3.62) \end{aligned}$$

Using the expansions in Appendix C.3, parts of the Equation (3.62) can be approximated upto maximum 6^{th}-order statistics as under:

$$
\begin{aligned}
&\exp\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)\cos\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)\\
&\quad=1+\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)+\frac{1}{2}\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)^{\otimes 2}\\
&\qquad-\frac{1}{2}\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)^{\otimes 2}-\dots\\
&\quad=1+\left(-\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}+\frac{\boldsymbol{\delta}(4,d)'}{4!}\boldsymbol{\lambda}^{\otimes 4}-\frac{\boldsymbol{\delta}(6,d)'}{6!}\boldsymbol{\lambda}^{\otimes 6}\right)+\frac{1}{2}\left(-\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}\right)^{\otimes 2}\\
&\qquad-\frac{1}{2}\left(-\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}+\frac{\boldsymbol{\delta}(3,d)'}{3!}\boldsymbol{\lambda}^{\otimes 3}\right)^{\otimes 2}+\frac{1}{6}\left(-\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}\right)^{\otimes 3}-\frac{1}{2}\left(\left(-\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}\right.\right.\\
&\qquad\left.\left.+\frac{\boldsymbol{\delta}(4,d)'}{4!}\boldsymbol{\lambda}^{\otimes 4}\right)\otimes(\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda})^{\otimes 2}\right)-\frac{1}{4}\left(\left(-\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}\right)^{\otimes 2}\otimes(\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda})^{\otimes 2}\right)\\
&\qquad+\frac{1}{4!}\left(-\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}\right)^{\otimes 4}-\frac{1}{4!}\left(-\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}\right)^{\otimes 4}-\left(\frac{\boldsymbol{\delta}(2,d)'}{2!}\boldsymbol{\lambda}^{\otimes 2}\otimes\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}^{\otimes 4}\right)+\dots\\
&\quad=1-\frac{1}{2}\left(\boldsymbol{\delta}(1,d)^{\otimes 2}+\boldsymbol{\delta}(2,d)\right)'\boldsymbol{\lambda}^{\otimes 2}+\frac{1}{4!}\left(\boldsymbol{\delta}(1,d)^{\otimes 4}+6\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2}\right.\\
&\qquad\left.+3\boldsymbol{\delta}(2,d)^{\otimes 2}+4\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(1,d)+\boldsymbol{\delta}(4,d)\right)'\boldsymbol{\lambda}^{\otimes 4}-\frac{1}{6!}\left(\boldsymbol{\delta}(1,d)^{\otimes 6}+15\boldsymbol{\delta}(2,d)^{\otimes 3}\right.\\
&\qquad+10\boldsymbol{\delta}(3,d)^{\otimes 2}+15\boldsymbol{\delta}(4,d)\otimes\boldsymbol{\delta}(2,d)+15\boldsymbol{\delta}(4,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2}+20\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2}\\
&\qquad+15\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 4}+45\boldsymbol{\delta}(2,d)^{\otimes 2}\otimes\boldsymbol{\delta}(1,d)^{\otimes 2}+6\boldsymbol{\delta}(5,d)\otimes\boldsymbol{\delta}(1,d)\\
&\qquad\left.+60\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)+\boldsymbol{\delta}(6,d)\right)'\boldsymbol{\lambda}^{\otimes 6}+\dots
\end{aligned}
\tag{3.63}
$$

Similarly,

$$\begin{aligned}
&\exp\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)\sin\left(\sum_{k=1}^{\infty}\frac{\boldsymbol{\delta}(2k-1,d)'}{(2k-1)!}(-i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}\right)\\
&=\left(-\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}+\frac{\boldsymbol{\delta}(3,d)'}{3!}\boldsymbol{\lambda}^{\otimes 3}-\frac{\boldsymbol{\delta}(5,d)'}{5!}\boldsymbol{\lambda}^{\otimes 5}\right)+\left(\frac{\boldsymbol{\delta}(2,d)'\boldsymbol{\lambda}^{\otimes 2}}{2!}\otimes\frac{\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}}{1}\right.\\
&\left.+\frac{\boldsymbol{\delta}(3,d)'\boldsymbol{\lambda}^{\otimes 3}}{3!}\otimes\frac{\boldsymbol{\delta}(2,d)'\boldsymbol{\lambda}^{\otimes 2}}{2!}+\frac{\boldsymbol{\delta}(4,d)'\boldsymbol{\lambda}^{\otimes 4}}{4!}\otimes\frac{\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}^{\otimes 1}}{1!}\right)\\
&-\frac{1}{2}\left(\frac{\boldsymbol{\delta}(2,d)'\boldsymbol{\lambda}^{\otimes 2}}{2!}\right)^{\otimes 2}\otimes\frac{\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}^{\otimes 1}}{1!}-\left(-\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}\right)^{\otimes 3}+\ldots\\
&=\boldsymbol{\delta}(1,d)'\boldsymbol{\lambda}+\frac{1}{3!}\left(\boldsymbol{\delta}(3,d)+3\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)+\boldsymbol{\delta}(1,d)^{\otimes 3}\right)'\boldsymbol{\lambda}^{\otimes 3}-\frac{1}{5!}\left(\boldsymbol{\delta}(5,d)\right.\\
&+5\boldsymbol{\delta}(4,d)\otimes\boldsymbol{\delta}(1,d)+15\boldsymbol{\delta}(2,d)^{\otimes 2}\otimes\boldsymbol{\delta}(1,d)+10\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(2,d)\\
&\left.+10\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 3}+10\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2}+\boldsymbol{\delta}(1,d)^{\otimes 5}\right)'\boldsymbol{\lambda}^{\otimes 5}\ldots
\end{aligned}\tag{3.64}$$

Now, $\mathbf{D}_{\mathbf{x}}^{\otimes k}\psi(\mathbf{x})$ can be obtained by taking k^{th}-order K-derivative of Equation (3.39) as under:

$$\begin{aligned}
\psi^{(k)}(\mathbf{x})=\mathbf{D}_{\mathbf{x}}^{\otimes k}\psi(\mathbf{x})=\frac{1}{(2\pi)^d}\int_{\mathbb{R}^d}(\boldsymbol{\lambda})^{\otimes k}\exp\left(\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k,d)'}{2k!}(i\boldsymbol{\lambda})^{\otimes 2k}\right)\\
\cos\left(\mathbf{x}'\boldsymbol{\lambda}+\sum_{k=1}^{\infty}\frac{\mathbf{c}_r(2k-1,d)'}{(2k-1)!}(i)^{2k}(\boldsymbol{\lambda})^{\otimes(2k-1)}+\frac{k\pi}{2}\right)d\boldsymbol{\lambda}
\end{aligned}\tag{3.65}$$

The above Equation (3.65) with the previous results on expansions in Equation (3.63) and Equation (3.64) can be used to simplify the Equation (3.62). This derives the Generalized Gram-Charlier (GGC) series expressing an unknown *pdf* $f(\mathbf{x})$ in terms of the cumulant difference vectors $(\boldsymbol{\delta}(k,d))$ and derivatives of a reference *pdf* $\psi^{(k)}(\mathbf{x})$ as under:

$$f(\mathbf{x})=\sum_{k=0}^{\infty}(-1)^k\frac{\boldsymbol{\alpha}(k,d)'}{k!}\psi^{(k)}(\mathbf{x})\tag{3.66}$$

where,

$$\begin{aligned}
\boldsymbol{\alpha}(0,d) &= 1\\
\boldsymbol{\alpha}(1,d) &= \boldsymbol{\delta}(1,d)\\
\boldsymbol{\alpha}(2,d) &= \boldsymbol{\delta}(2,d) + \boldsymbol{\delta}(1,d)^{\otimes 2}\\
\boldsymbol{\alpha}(3,d) &= \boldsymbol{\delta}(3,d) + 3\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d) + \boldsymbol{\delta}(1,d)^{\otimes 3}\\
\boldsymbol{\alpha}(4,d) &= \boldsymbol{\delta}(4,d) + 4\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(1,d) + 3\boldsymbol{\delta}(2,d)^{\otimes 2} + 6\boldsymbol{\delta}(2,d)\\
&\quad \otimes \boldsymbol{\delta}(1,d)^{\otimes 2} + \boldsymbol{\delta}(1,d)^{\otimes 4}\\
\boldsymbol{\alpha}(5,d) &= \boldsymbol{\delta}(5,d) + 5\boldsymbol{\delta}(4,d)\otimes\boldsymbol{\delta}(1,d) + 10\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(2,d)\\
&\quad + 10\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2} + 15\boldsymbol{\delta}(2,d)^{\otimes 2}\otimes\boldsymbol{\delta}(1,d)\\
&\quad + 10\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 3} + \boldsymbol{\delta}(1,d)^{\otimes 5}\\
\boldsymbol{\alpha}(6,d) &= \boldsymbol{\delta}(6,d) + 6\boldsymbol{\delta}(5,d)\otimes\boldsymbol{\delta}(1,d) + 15\boldsymbol{\delta}(4,d)\otimes\boldsymbol{\delta}(2,d)\\
&\quad + 15\boldsymbol{\delta}(4,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2} + 10\boldsymbol{\delta}(3,d)^{\otimes 2} + 60\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(2,d)\\
&\quad \otimes\boldsymbol{\delta}(1,d) + 20\boldsymbol{\delta}(3,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 2} + 15\boldsymbol{\delta}(2,d)^{\otimes 3} + 45\boldsymbol{\delta}(2,d)^{\otimes 2}\\
&\quad \otimes\boldsymbol{\delta}(1,d)^{\otimes 2} + 15\boldsymbol{\delta}(2,d)\otimes\boldsymbol{\delta}(1,d)^{\otimes 4} + \boldsymbol{\delta}(1,d)^{\otimes 6}
\end{aligned} \tag{3.67}$$

The above set of equations (3.67) has exact resemblance with that expressing moments in terms of the cumulants in Section 3.7.1. This must happen, as Equation (3.66) for GGC expansion with $\delta(\mathbf{x})$ as a reference *pdf* is matching Equation (3.26). This matching proves that $\boldsymbol{\alpha}(k,d)$ is related in same way to $\boldsymbol{\delta}(k,d)$, as $\mathbf{m}(k,d)$ to $\mathbf{c}(k,d)$. That is,:

$$\sum_{k=0}^{\infty}\boldsymbol{\alpha}(k,d)'\frac{\boldsymbol{\lambda}^{\otimes k}}{k!} = \exp\left(\sum_{k=1}^{\infty}\boldsymbol{\delta}(k,d)'\frac{\boldsymbol{\lambda}^{\otimes k}}{k!}\right) \tag{3.68}$$

Further, the $\boldsymbol{\alpha}(k,d)$ in Equation (3.66) recursively can be obtained in terms of the cumulant difference vector $\boldsymbol{\delta}(k,d)$ as under:

$$\boldsymbol{\alpha}(k,d) = \sum_{p=0}^{k-1}\binom{k-1}{p}\mathbf{K}^{-1}_{\mathfrak{p}h\leftrightarrow l}\boldsymbol{\delta}(k-p,d)\otimes\boldsymbol{\alpha}(p,d) \tag{3.69}$$

where, $\mathbf{K}^{-1}_{\mathfrak{p}h\leftrightarrow l}$ is a specific commutation matrix; as described previously; to change the order of the cumulants for Kronecker product such that the expression has decreasing order cumulants from left to the right.

The verification of the derived GGC can be obtained by taking Gaussian density as a reference *pdf*. With Gaussian density as a reference, $\boldsymbol{\delta}(1,d) = \mathbf{0}$ and $\boldsymbol{\delta}(2,d) = \mathbf{0}$. So, the coefficients

$\boldsymbol{\alpha}(k,d)$ in Equation (3.66) can be derived as under:

$$\begin{aligned}
\boldsymbol{\alpha}(0,d) &= 1\\
\boldsymbol{\alpha}(1,d) &= \mathbf{0}\\
\boldsymbol{\alpha}(2,d) &= \mathbf{0}\\
\boldsymbol{\alpha}(3,d) &= \boldsymbol{\delta}(3,d) = \mathbf{c}(3,d)\\
\boldsymbol{\alpha}(4,d) &= \boldsymbol{\delta}(4,d) = \mathbf{c}(4,d)\\
\boldsymbol{\alpha}(5,d) &= \boldsymbol{\delta}(5,d) = \mathbf{c}(5,d)\\
\boldsymbol{\alpha}(6,d) &= \boldsymbol{\delta}(6,d) + 10\boldsymbol{\delta}(3,d)^{\otimes 2} = \mathbf{c}(6,d) + 10\mathbf{c}(3,d)^{\otimes 2}
\end{aligned} \tag{3.70}$$

Thus, the GGC series is derived and verified using known examples.

3.12 Characteristic function of an unknown random vector in terms of a reference characteristic function

The GGC derived as in Equation (3.66) can be used to give the characteristic function of an unknown *pdf*, in terms of the characteristic function of a reference *pdf*. For that taking Fourier transform (F) of Equation (3.66), we get:

$$\mathcal{F}_{\mathbf{x}}(\boldsymbol{\lambda}) = \sum_{k=0}^{\infty}(-1)^k \frac{\boldsymbol{\alpha}(k,d)'}{k!}\mathsf{F}(\psi^{(k)}(\mathbf{x})) \tag{3.71}$$

$$\mathcal{F}_{\mathbf{x}}(\boldsymbol{\lambda}) = \left[\sum_{k=0}^{\infty}\boldsymbol{\alpha}(k,d)'\frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right]\mathsf{F}(\psi(\mathbf{x})) \quad (\because \text{differentiation property of } \mathsf{F}\,) \tag{3.72}$$

$$= \exp\left[\sum_{k=1}^{\infty}\boldsymbol{\delta}(k,d)'\frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right]\mathcal{F}_r(\mathbf{x}) \quad (\because \text{ Equation (3.68)}) \tag{3.73}$$

where, $\mathcal{F}_r$ is the characteristic function of the reference *pdf*.

3.13 Compact derivation for the Generalized Gram-Charlier expansion

The compact derivation of Equation (3.66) follows as under:

$$\mathcal{F}_{\mathbf{x}}(\boldsymbol{\lambda}) = \exp\left[\sum_{k=1}^{\infty} \mathbf{c}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right] \quad (\because \text{ definition in Equation (3.31) }) \tag{3.74}$$

$$= \exp\left[\sum_{k=1}^{\infty} \boldsymbol{\delta}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right] \exp\left[\sum_{k=1}^{\infty} \mathbf{c}_r(k,d) \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right] \tag{3.75}$$

$$(\because \boldsymbol{\delta}(k,d) = \mathbf{c}(k,d) - \mathbf{c}_r(k,d))$$

$$= \left[\sum_{k=0}^{\infty} \boldsymbol{\alpha}(k,d)' \frac{(i\boldsymbol{\lambda})^{\otimes k}}{k!}\right] \mathsf{F}(\psi(\mathbf{x})) \tag{3.76}$$

Taking inverse Fourier transform of the above equation brings

$$f(\mathbf{x}) = \sum_{k=0}^{\infty} \boldsymbol{\alpha}(k,d)' \frac{(-1)^k}{k!} \boldsymbol{\delta}^{(k)}(\mathbf{x}) * \psi(\mathbf{x}) \tag{3.77}$$

$$\text{or } f(\mathbf{x}) = \sum_{k=0}^{\infty} \boldsymbol{\alpha}(k,d)' \frac{(-1)^k}{k!} \psi^{(k)}(\mathbf{x}) \tag{3.78}$$

where, $*$ indicates convolution. Thus, the Equation (3.66) is obtained in a more compact way.

3.14 Conclusion

The chapter has derived multivariate Generalized Gram-Charlier (GGC) expansion in Equation (3.66); combined with Equation (3.69); that expresses an unknown multivariate *pdf* in terms of vector cumulants and vector derivatives of a reference *pdf.* The multivariate Gram-Charlier A series is derived in Equation (3.59) and Equation (3.60) representing an unknown multivariate *pdf* in terms of its vector cumulants and vector Hermite polynomials. There has been also derived compact formulas for obtaining multivariate vector moments from vector cumulants in Equation (3.35) and vise-a-verse in Equation (3.36); the integral form of multivariate *pdf* representation in Equation (3.39) and the integral form of multivariate vector Hermite polynomials in Equation (3.47), as well, in Equation (3.48). The expressions are derived using only elementary calculus of several variables in vector notations through Kronecker product based derivative operator. Thus, they are more transparent and more comprehensive compare to their corresponding multi-linear matrix representations or tensor representations.

3.15 AMISE in vector notations for bandwidth selection multivariate KDE

The derivation for bandwidth parameter selection in multivariate KDE, satisfying AMISE between the estimated multivariate PDF and the actual multivariate PDF, can be found in (136). For the reasons discussed in Section 3.1, this chapter re-derives multivariate AMISE in terms of the vector notations. The conventional higher order derivatives of a multivariate PDF necessitates matrix or tensor notations. Instead, this chapter achieves vectorization using the Kronecker product of the multivariate PDF with the vector differential operator $\mathbf{D_x} = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \cdots, \frac{\partial}{\partial x_d}\right)'$.

$$\text{MISE}(f(\mathbf{x}), \hat{f(\mathbf{x})}) = \int_{\mathbb{R}^d} \text{Bias}^2(\hat{f(\mathbf{x})})d\mathbf{x} + \int_{\mathbb{R}^d} \text{Var}(\hat{f(\mathbf{x})})d\mathbf{x} \tag{3.79}$$

The bias and variance estimations, using the Taylor Series expansion in Equation (3.21), are derived as under.

$$\begin{aligned} E\{\hat{f(\mathbf{x})}\} &= \frac{1}{N}\sum_{i=1}^{N} E\left\{\mathcal{K}_{\mathbf{H}}(\mathbf{x}-\mathbf{x}_i)\right\} && (3.80)\\ &= \int_{\mathbb{R}^d} \mathcal{K}_{\mathbf{H}}(\mathbf{u}-\mathbf{x})f(\mathbf{u})d\mathbf{u} \\ &= \int_{\mathbb{R}^d} \mathcal{K}(\mathbf{z})f(\mathbf{x}+\mathbf{Hz})d\mathbf{z} \text{ ($\because$ substituting } z = \mathbf{u}-\mathbf{x}\text{)} \\ &= \int_{\mathbb{R}^d} \mathcal{K}(\mathbf{z})\left(f(\mathbf{x}) + \mathbf{k}(1,d)'(\mathbf{Hz}) + \frac{1}{2}\mathbf{k}(2,d)'(\mathbf{Hz})^{\otimes 2} + O(\text{tr}(\mathbf{H}^{\otimes 2}))\right)d\mathbf{z} && (3.81) \end{aligned}$$

where, $\mathbf{k}(i,d) = \mathbf{D}_{\mathbf{x}}^{\otimes i} f(\mathbf{x}) \mid_{\mathbf{x}=\mathbf{x}}$
Assuming the kernel with symmetric, bounded and uncorrelated PDF the following properties are satisfied:

$$\int_{\mathbb{R}^d} \mathbf{u}\mathcal{K}(\mathbf{u})d\mathbf{u} = \mathbf{0} \tag{3.82}$$

$$\int_{\mathbb{R}^d} \mathbf{u}\otimes\mathbf{u}\mathcal{K}(\mathbf{u})d\mathbf{u} = \mathbf{m}_{\mathcal{K}}(2,d) = \mu_2(\mathcal{K})\text{Vec}(\mathbf{I}_{(d\times d)}) \tag{3.83}$$

$$= \mu_2(\mathcal{K})\boldsymbol{\delta}_2 \quad \text{where, } \boldsymbol{\delta}_2 = \text{Vec}(\mathbf{I}_{(d\times d)}) \tag{3.84}$$

Here, $\mathbf{m}_{\mathcal{K}}(2,d)$ is the second order moment vector of the d-variate kernel with d^2 components and $\boldsymbol{\delta}_2$ is vector of size $(d^2 \times 1)$, used as an indicator function with only specific d values being 1. The second order moment of each component is constant with value $\mu_2(\mathcal{K})$ and all cross-moments are zero. Using these properties, the bias can be written as:

$$\Rightarrow \text{Bias}(f(\hat{\mathbf{x}})) = E\{f(\hat{\mathbf{x}})\} - f(\mathbf{x}) = \frac{1}{2}\mathbf{k}(2,d)' \left(\mathbf{H}^{\otimes 2}\mathbf{m}_{\mathcal{K}}(2,d)\right) + O(\mathbf{H}^{\otimes 2}) \tag{3.85}$$

Similarly, the variance of the estimation can be approximated as under:

$$\begin{aligned}
\text{Var}(\hat{f}(\mathbf{x})) &= \text{Var}\left(\frac{1}{N}\sum_{i=1}^{N}\mathcal{K}_{\mathbf{H}}(\mathbf{x}-\mathbf{x}_i)\right) \\
&= \frac{1}{N\text{det}(\mathbf{H})}\int_{\mathbb{R}^d}\mathcal{K}^2(z)f(\mathbf{x}+\mathbf{Hz})d\mathbf{z} - \frac{1}{N}E^2\left\{\hat{f}(\mathbf{x})\right\} \\
&= \frac{1}{N\text{det}(\mathbf{H})}\int_{\mathbb{R}^d}\mathcal{K}^2(z)\left(f(\mathbf{x}) + \mathbf{k}(1,d)'\mathbf{Hz} + \frac{1}{2}\mathbf{k}(2,d)'(\mathbf{Hz})^{\otimes 2} + O(\text{tr}(\mathbf{H}^{\otimes 2}))\right)d\mathbf{z} \\
&\quad - \frac{1}{N}E^2\left\{\hat{f}(\mathbf{x})\right\} \quad (\because \text{ using Taylor's expansion}) \\
\Rightarrow \text{Var}(\hat{f}(\mathbf{x})) &= \frac{1}{N\text{det}(\mathbf{H})}f(\mathbf{x})\int_{\mathbb{R}^d}\mathcal{K}^2(\mathbf{z})d\mathbf{z} + O\left(\frac{\det\mathbf{H}}{N}\right) \quad (\because \text{assuming large } N\text{, small } h)
\end{aligned} \tag{3.86}$$

Combining equations (3.79), (3.85) and (3.86); we get:

$$\text{MISE}(\hat{f}(\mathbf{x})) = \frac{1}{4}\int_{\mathbb{R}^d}\left(\mathbf{k}(2,d)'\mathbf{H}^{\otimes 2}\mathbf{m}_{\mathcal{K}}(2,d)\right)^2 d\mathbf{z} + \frac{R(\mathcal{K})}{N\text{det}(\mathbf{H})} + O\left(\text{tr}\left(\mathbf{H}^{\otimes 4}\right)\right) + O\left(\frac{\text{det}(\mathbf{H})}{N}\right)$$

where, $R(\mathcal{K}) = \int_{\mathbb{R}^d}\mathcal{K}^2(z)dz$. An asymptotic large sample approximation AMISE is obtained, assuming $\lim_{N\to\infty}\text{det}(\mathbf{H}) = 0$ and $\lim_{N\to\infty}N\text{det}(\mathbf{H}) = \infty$ i.e. $\text{det}(\mathbf{H})$ reduces to 0 at a rate slower than $1/N$.

$$\begin{aligned}
\text{AMISE}(\hat{f}(\mathbf{x})) &= \frac{1}{4}\int_{\mathbb{R}^d}\left(\mathbf{k}(2,d)'\mathbf{H}^{\otimes 2}\mathbf{m}_{\mathcal{K}}(2,d)\right)^2 d\mathbf{z} + \frac{1}{N\text{det}(\mathbf{H})}R(\mathcal{K}) \\
&= \frac{1}{4}\mu_2^2(\mathcal{K})\int_{\mathbb{R}^d}\left(\mathbf{k}(2,d)'\left(\mathbf{H}^{\otimes 2}\boldsymbol{\delta}_2\right)\right)^2 d\mathbf{z} + \frac{1}{N\text{det}(\mathbf{H})}R(\mathcal{K})(\because \text{ using Equation(3.84)})
\end{aligned} \tag{3.87}$$

The Equation (**??**) interprets that a small det($\mathbf{H}$) increases estimation variance, whereas, a larger det($\mathbf{H}$) increases estimation bias.

3.15.1 AMISE for $\mathbf{H} \in \mathcal{S}$

To simplify further the bandwidth estimation, let us assume $\mathbf{H} \in \mathcal{S}$, where $\mathcal{S} \subseteq \mathcal{D}$ with constant diagonal. Accordingly, let $\mathbf{H} = h\mathbf{H}_0$ i.e. same bandwidth in all directions. With this condition and an added assumption of $\mathbf{H}_0 = \mathbf{I}_d$ (an identity matrix with dimension $d \times d$), the AMISE can be given as:

$$\begin{aligned}\text{AMISE}(\hat{f}(\mathbf{x})) &= \frac{h^4}{4}\mu_2^2(\mathcal{K})\left(\mathbf{I}_d^{\otimes 4}\boldsymbol{\delta}_2^{\otimes 2}\right)'\mathbf{R}(f''(\mathbf{x})) + \frac{1}{Nh^d}R(\mathcal{K}) \\ &= \frac{h^4}{4}\mu_2^2(\mathcal{K})R(f''(\mathbf{x})) + \frac{1}{Nh^d}R(\mathcal{K}) \end{aligned} \tag{3.88}$$

Taking derivative of AMISE$(\hat{f}(\mathbf{x}))$ in Equation (3.88) with respect to h and comparing it to zero gives the optimal bandwidth parameter minimizing the AMISE. It is:

$$\frac{d}{dh}AMISE(f(\hat{x})) = h^3\mu_2^2(\mathcal{K})R(f''(\mathbf{x})) - \frac{d}{Nh^{(d+1)}}R(\mathcal{K}) = 0$$

$$\Rightarrow h_{AMISE} = \left(d\frac{R(\mathcal{K})}{\mu_2^2(\mathcal{K})R(f''(\mathbf{x}))N}\right)^{\frac{1}{d+4}} \tag{3.89}$$

$$= (CN)^{-\frac{1}{d+4}} \quad \text{where, } C = \frac{\mu_2^2(\mathcal{K})R(f''(\mathbf{x}))}{dR(\mathcal{K})} \tag{3.90}$$

Thus, the optimal bandwidth parameter depends upon some of the kernel parameters, number of samples and the second order derivative of the actual PDF being estimated.

By comparing Equation (**??**) for $h_i, \mathbf{H} \in \mathcal{D}$ with the Equation (3.89) for $h, \mathbf{H} \in \mathcal{S}$, we get a notion for $R(f''(\mathbf{x}))$ as under:

$$R(f''(\mathbf{x})) = d\left[\left(\mathbf{R}_i(f''(\mathbf{x}))\right)^{-(d+4)}\left(\prod_{p=1}^{d}\mathbf{R}_p(f''(\mathbf{x}))\right)\right]^{-1/4} \tag{3.91}$$

3.15.2 *Rule-of-Thumb* for multivariate KDE

Let us apply the *Rule-of-Thumb* for bandwidth estimation i.e. estimate the bandwidth assuming the unknown PDF as a multivariate Gaussian. For a multiplicative Gaussian kernel, $\mu_2(\mathcal{K}) = 1$, $R(\mathcal{K}) = 2^{-d}\pi^{-d/2}$ and $R(f''(\mathbf{x})) = \frac{d(d+2)}{2^{(d+2)}\pi^{d/2}\sigma^{(d+4)}}$. So, the bandwidth parameter h; for $\mathbf{H} \in \mathcal{S}, \mathbf{H} = h\mathbf{I}_d$; is obtained as under:

$$h_{ROT} = \left(\frac{4}{(2+d)N}\right)^{\frac{1}{4+d}}\sigma \tag{3.92}$$

where, σ is the standard deviation, assumed same in all directions.

3.16 Extended Rule-of-Thumb for multivariate PDF

The ExROT expressions in vector notation for bandwidth selection in multivariate KDE satisfying AMISE criteria requires estimation of either $\mathbf{R}(\mathbf{k}(2,d))$ or $\mathbf{R}_i(\mathbf{k}(2,d))$ or $R(\mathbf{k}(2,d))$ that in turns require estimation of $\mathbf{k}(2,d)$. This can be achieved by twice application of the vector derivative operator with Kronecker product to the Equation (3.66) for GGC Series. The $\mathbf{k}(2,d)$ is derived as under:

$$\mathbf{k}(2,d) = \mathbf{D}_{\mathbf{x}}^{\otimes 2} f(\mathbf{x}) = \sum_{k=0}^{\infty} (-1)^k \frac{(\boldsymbol{\alpha}(k,d) \otimes \mathbf{I}_{d^2})'}{k!} \psi^{(k+2)}(\mathbf{x}) \tag{3.93}$$

Taking Gaussian PDF as a reference PDF; i.e. $\psi(\mathbf{x}) = G(\mathbf{x})$; in Equation (3.93) or directly taking twice differentiation of Equation (3.56) for GCA Series and also using Equation (**??**) for kth order vector Hermite polynomial $\mathbf{H}_k$[§] the $\mathbf{k}(2,d)$ is re-written as under:

$$\mathbf{k}(2,d) = \mathbf{D}_{\mathbf{x}}^{\otimes 2} f(\mathbf{x}) = \sum_{k=0}^{\infty} (-1)^k \frac{(\mathbf{c}(k,d) \otimes \mathbf{I}_{d^2})'}{k!} G^{(k+2)}(\mathbf{x}) \tag{3.94}$$

$$\approx \frac{(1 \otimes \mathbf{I}_{d^2})'}{1} G^{(2)}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) - \frac{(\mathbf{c}(3,d) \otimes \mathbf{I}_{d^2})'}{3!} G^{(5)}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) + \frac{(\mathbf{c}(4,d) \otimes \mathbf{I}_{d^2})'}{4!} G^{(6)}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) \tag{3.95}$$

$$\mathbf{k}(2,d) = \left[\sum_{k=0}^{\infty} \frac{(\mathbf{c}(k,d) \otimes \mathbf{I}_{d^2})'}{k!} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes(k+2)} \mathbf{H}_{k+2}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) \right] G(\mathbf{x}) \tag{3.96}$$

$$\begin{aligned} &\approx (2\pi)^{-d/2} \left|\mathbf{C}_{\mathbf{x}}\right|^{-1/2} \exp\left(-\frac{1}{2} \left(\mathrm{Vec}\mathbf{C}_{\mathbf{x}}^{-1}\right)' \mathbf{x} \otimes \mathbf{x}\right) \left[\frac{(1 \otimes \mathbf{I}_{d^2})'}{1} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 2} \mathbf{H}_2(\mathbf{x}) \right. \\ &\left. - \frac{(\mathbf{c}(3,d) \otimes \mathbf{I}_{d^2})'}{3!} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 5} \mathbf{H}_5(\mathbf{x}) + \frac{(\mathbf{c}(4,d) \otimes \mathbf{I}_{d^2})'}{4!} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 6} \mathbf{H}_6(\mathbf{x}) \right] \end{aligned} \tag{3.97}$$

$$\begin{aligned} \mathbf{R}(f''(\mathbf{x})) \approx & \int_{\mathbb{R}^d} \left[\left\{ \frac{(1 \otimes \mathbf{I}_{d^2})'}{1} G^{(2)}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) - \frac{(\mathbf{c}(3,d) \otimes \mathbf{I}_{d^2})'}{3!} G^{(5)}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) \right. \right. \\ & \left. \left. + \frac{(\mathbf{c}(4,d) \otimes \mathbf{I}_{d^2})'}{4!} G^{(6)}(\mathbf{x}; \mathbf{C}_{\mathbf{x}}) \right\} \text{ o } \delta_2 \right]^2 d\mathbf{z} \end{aligned} \tag{3.98}$$

$$\begin{aligned} R(f''(\mathbf{x})) \approx & \sum_{i=1}^{d^4} \left[\int_{\mathbb{R}^d} G^2(\mathbf{x}) \left\{ \left(\frac{(1 \otimes \mathbf{I}_{d^2})'}{1} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 2} \mathbf{H}_2(\mathbf{x}) - \frac{(\mathbf{c}(3,d) \otimes \mathbf{I}_{d^2})'}{3!} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 5} \mathbf{H}_5(\mathbf{x}) \right. \right. \right. \\ & \left. \left. \left. + \frac{(\mathbf{c}(4,d) \otimes \mathbf{I}_{d^2})'}{4!} \left(\mathbf{C}_{\mathbf{x}}^{-1}\right)^{\otimes 6} \mathbf{H}_6(\mathbf{x}) \right) \text{ o } \boldsymbol{\delta}_2 \right\}^{\otimes 2} d\mathbf{x} \right]_i \end{aligned} \tag{3.99}$$

[§] The symbol $\mathbf{H}_k$ for kth order Hermite polynomial need not be confused with the notation $\mathbf{H}$ for bandwidth matrix.

3.16.1 ExROT for $\mathbf{H} \in \mathcal{S}$

The assumption $\mathbf{H} \in \mathcal{S}$, requires estimation of $R(f''(\mathbf{x}))$ for the selection of bandwidth parameter h, which is obtained as under:

$$
\begin{aligned}
R(f''(\mathbf{x})) &= \int_{\mathbb{R}^d} \left[\sum_{i=1}^{d} \frac{\partial^2 f(\mathbf{x})}{\partial x_i^2} \right]^2 \\
&\approx \int_{\mathbb{R}^d} \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \left[G(\mathbf{x}) - \frac{\mathbf{c}(3,d)'}{3!} G^{(3)}(\mathbf{x}) + \frac{\mathbf{c}(4,d)'}{4!} G^{(4)}(\mathbf{x}) \right] \right]^2 d\mathbf{x} \qquad (3.100) \\
&= \int_{\mathbb{R}^d} \left(\left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} G(\mathbf{x}) \right]^2 + \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \frac{\mathbf{c}(3,d)' G^{(3)}(\mathbf{x})}{3!} \right]^2 + \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \frac{\mathbf{c}(4,d)' G^{(4)}(\mathbf{x})}{4!} \right]^2 \right. \\
&\quad -2 \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} G(\mathbf{x}) \right] \left[\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} \frac{\mathbf{c}(3,d)' G^{(3)}(\mathbf{x})}{3!} \right] \\
&\quad -2 \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \frac{\mathbf{c}(3,d)' G^{(3)}(\mathbf{x})}{3!} \right] \left[\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} \frac{\mathbf{c}(4,d)' G^{(4)}(\mathbf{x})}{4!} \right] \\
&\quad \left. +2 \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} G(\mathbf{x}) \right] \left[\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} \frac{\mathbf{c}(4,d)' G^{(4)}(\mathbf{x})}{4!} \right] \right) d\mathbf{x} \qquad (3.101) \\
&= Q_1 + Q_2 + Q_3 - Q_4 - Q_5 + Q_6 \qquad (3.102)
\end{aligned}
$$

where, the $Q_i, i = 1:6$ are the symbols for corresponding terms simplified as under.

ExROT based on 3rd and 4th order cross-cumulants assumed zero

The simplification is obtained knowing $G(\mathbf{x}) = \prod_{i=1}^{d} G(x_i)$ and assuming all third order and fourth order cross-moments to be zero. Also, there have been used the symbol $c_i(3,d)$ as the third order cumulant (i.e. skewness) of x_i, $\mathbf{c}(3, d, \text{mean}) = \frac{1}{d} \sum_{i=1}^{d} c_i(3,d)$ as the mean of the skewness and $\mathbf{c}(4, d, \text{mean}) = \frac{1}{d}\sum_{i=1}^{d} c_i(4, d)$ as the mean of the kurtosis.

$$Q_1 = \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}G(\mathbf{x})\right]^2 d\mathbf{x} = \frac{4!!d + 2!!2!!d(d-1)}{2^{d+2}\pi^{d/2}\sigma^{d+4}} = \frac{d(d+2)}{2^{d+2}\pi^{d/2}\sigma^{d+4}} \tag{3.103}$$

$$Q_2 = \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{\mathbf{c}(3,d)'G^{(3)}(\mathbf{x})}{3!}\right]^2 d\mathbf{x} \tag{3.104}$$

$$= \int_{\mathbb{R}^d}\left\{\left[\sum_{i=1}^{d}\frac{c_i(3,d)}{3!}\frac{\partial^5}{\partial x_i^5}G(\mathbf{x})\right]^2 + \left[\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\sum_{k=1,k\neq j}^{d}\frac{c_k(3,d)(\mathbf{x})}{3!}\frac{\partial^3}{\partial x_k^3}G(\mathbf{x})\right]^2\right.$$
$$\left. +2\left[\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{c_i(3,d)c_j(3,d)}{3!3!}\frac{\partial^5}{\partial x_i^5}G(\mathbf{x})\left(\frac{\partial^2}{\partial x_j^2}\sum_{k=1,k\neq j}^{d}\frac{\partial^3}{\partial x_k^3}G(\mathbf{x})\right)\right]\right\} d\mathbf{x} \tag{3.105}$$

$$= \int_{\mathbb{R}^d}\left\{\sum_{i=1}^{d}\left[\frac{c_i(3,d)}{3!}\mathbf{H}_5(x_i)G(\mathbf{x})\right]^2 + \left[\sum_{j=1}^{d}\sum_{k=1,k\neq j}^{d}\frac{c_k(3,d)(\mathbf{x})}{3!}\mathbf{H}_2(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right]^2\right.$$
$$\left. +2\left[\sum_{i=1}^{d}\frac{c_i(3,d)c_j(3,d)}{3!3!}\mathbf{H}_5(x_i)G(\mathbf{x})\left(\sum_{j=1}^{d}\sum_{k=1,k\neq j}^{d}\mathbf{H}_2(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right)\right]\right\} d\mathbf{x} \tag{3.106}$$

$$= \frac{c_i^2(3,d)}{(3!)^2}\frac{10!!d}{2^{d+5}\pi^{d/2}\sigma^{d+10}} + \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\frac{4!!6!!d(d-1) + 2!!6!!2!!d(d-1)(d-2)}{2^{d+5}\pi^{d/2}\sigma^{d+10}} \tag{3.107}$$

$$= \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\left[\frac{15d(d^2+62)}{2^{d+5}\pi^{d/2}\sigma^{d+10}}\right] \quad (\because \text{ assuming } c_i(3,d) = \mathbf{c}(3,d,\text{mean})) \tag{3.108}$$

$$Q_3 = \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{\mathbf{c}(4,d)'G^{(4!)}(\mathbf{x})}{4}\right]^2 d\mathbf{x} \tag{3.109}$$

$$= \int_{\mathbb{R}^d}\left\{\left[\sum_{i=1}^{d}\frac{c_i(4,d)}{4!}\mathbf{H}_6(x_i)G(\mathbf{x})\right]^2 + \left[\sum_{j=1}^{d}\sum_{k=1,k\neq j}^{d}\frac{c_k(4,d)(\mathbf{x})}{4!}\mathbf{H}_2(x_j)\mathbf{H}_4(x_k)G(\mathbf{x})\right]^2\right.$$
$$\left. +2\left[\sum_{i=1}^{d}\frac{c_i(4,d)c_j(4,d)}{4!4!}\mathbf{H}_6(x_i)G(\mathbf{x})\left(\sum_{j=1}^{d}\sum_{k=1,k\neq j}^{d}\mathbf{H}_2(x_j)\mathbf{H}_4(x_k)G(\mathbf{x})\right)\right]\right\} d\mathbf{x} \tag{3.110}$$

$$= \frac{c^2(4,d,mean)}{(4!)^2}\left\{\frac{(12!!d + 6!!6!!d(d-1)) + 4!!8!!d(d-1) + 4!!4!!4!!d(d-1)(d-2)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right.$$
$$+\frac{2!!8!!2!!d(d-1)(d-2) + 2!!4!!2!!4!!d(d-1)(d-2)(d-3)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}$$
$$\left.+2\frac{8!!4!!d(d-1) + 6!!2!!4!!d(d-1)(d-2) + 10!!2!!d(d-1))}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right\} \tag{3.111}$$

$$= \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2}\left[\frac{3d(3d^3 + 56d^2 + 516d + 2890)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right] \tag{3.112}$$

$$Q_4 = \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} G(\mathbf{x})\right]\left[\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} \frac{\mathbf{c}(3,d)' G^{(3)}(\mathbf{x})}{3!}\right] d\mathbf{x} \tag{3.113}$$

$$\begin{aligned}
&= \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d} \frac{c_i(3,d)}{3!} \mathbf{H}_2(x_i)\mathbf{H}_5(x_i) G^2(\mathbf{x})\right.\\
&\quad + \sum_{i=1}^{d} \sum_{j=1, j\neq i}^{d} \frac{c_j(3,d)(\mathbf{x})}{3!} \mathbf{H}_2(x_i)\mathbf{H}_2(x_i)\mathbf{H}_3(x_j) G^2(\mathbf{x})\\
&\quad + \sum_{i=1}^{d} \sum_{j=1, j\neq i}^{d} \frac{c_k(3,d)(\mathbf{x})}{3!} \mathbf{H}_2(x_i)\mathbf{H}_2(x_j)\mathbf{H}_3(x_i) G^2(\mathbf{x})\\
&\quad \left. + \sum_{i=1}^{d} \sum_{j=1, j\neq i}^{d} \sum_{k=1, k\neq i,j}^{d} \frac{c_i(3,d)(\mathbf{x})}{3!} \mathbf{H}_2(x_i)\mathbf{H}_2(x_j)\mathbf{H}_3(x_k) G^2(\mathbf{x})\right] d\mathbf{x}
\end{aligned} \tag{3.114}$$

$$= 0 \tag{3.115}$$

$$Q_5 = \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \frac{\mathbf{c}(3,d)' G^{(3)}(\mathbf{x})}{3!}\right]\left[\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} \frac{\mathbf{c}(4,d)' G^{(4)}(\mathbf{x})}{4!}\right] d\mathbf{x} \tag{3.116}$$

$$\begin{aligned}
&= \int_{\mathbb{R}^d} \left\{ 2\left[\sum_{i=1}^{d} \frac{c_i(3,d)}{3!} \mathbf{H}_5(x_i) G(\mathbf{x}) + \sum_{j=1}^{d} \sum_{j=1, j\neq i}^{d} \frac{c_j(3,d)}{3!} \mathbf{H}_2(x_i)\mathbf{H}_3(x_j) G(\mathbf{x})\right]\right.\\
&\quad \left.\left[\sum_{k=1}^{d} \frac{c_k(4,d)}{4!} \mathbf{H}_6(x_k) G(\mathbf{x}) + \sum_{l=1, l\neq k}^{d} \frac{c_l(4,d)}{4!} \mathbf{H}_2(x_k)\mathbf{H}_4(x_l) G(\mathbf{x})\right]\right\} d\mathbf{x}
\end{aligned} \tag{3.117}$$

$$= 0 \tag{3.118}$$

$$Q_6 = \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} G(\mathbf{x})\right]\left[\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} \frac{\mathbf{c}(4,d)' G^{(4)}(\mathbf{x})}{4!}\right] d\mathbf{x} \tag{3.119}$$

$$\begin{aligned}
&= \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d} \mathbf{H}_2(x_i) G(\mathbf{x})\right]\left[\sum_{i=1}^{d} \frac{c_i(4,d)}{4!} \mathbf{H}_6(x_i) G(\mathbf{x})\right.\\
&\quad \left. + \sum_{i=1}^{d} \sum_{j=1, j\neq i}^{d} \frac{c_j(4,d)}{4!} \mathbf{H}_2(x_i)\mathbf{H}_4(x_j) G(\mathbf{x})\right] d\mathbf{x}
\end{aligned} \tag{3.120}$$

$$= 2\frac{\mathbf{c}(4,d,\text{mean})}{(4!)} \frac{(2!!)(6!!)d^2 + 4!!4!!d(d-1) + 2!!2!!4!!d^2(d-1)}{2^{d+4}\pi^{d/2}\sigma^{d+8}} \tag{3.121}$$

$$= \frac{\mathbf{c}(4,d,\text{mean})}{(4!)} \frac{6d(d^2+7d-3)}{2^{d+4}\pi^{d/2}\sigma^{d+8}} \tag{3.122}$$

Combining above simplifications, the formula for $R(f''(\mathbf{x}))$ can be derived as under:

$$R(f''(\mathbf{x})) = \frac{d(d+2)}{2^{d+2}\pi^{d/2}\sigma^{d+4}} + \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\left[\frac{15d(d^2+62)}{2^{d+5}\pi^{d/2}\sigma^{d+10}}\right] + \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2} \left[\frac{3d(3d^3+56d^2+516d+2890)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right] + \frac{\mathbf{c}(4,d,\text{mean})}{(4!)}\left[\frac{6d(d^2+7d-3)}{2^{d+4}\pi^{d/2}\sigma^{d+8}}\right] \quad (3.123)$$

With $R(\mathcal{K}) = 2^{-d}\pi^{-d/2}$, $\mu_2(\mathcal{K}) = 1$ and using Equation (3.90); the bandwidth parameter using GCA based ExROT (h_i(ExROT)) can be given as under:

$$h_{\text{ExROT}} = [CN]^{-\frac{1}{d+4}} \quad (3.124)$$

$$\text{where, } C = \frac{(d+2)}{2^2\sigma^{d+4}}\left[1 + \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\frac{15(d^2+62)}{2^3(d+2)\sigma^6} + \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2} \frac{3(3d^3+56d^2+516d+2890)}{2^4(d+2)\sigma^8} + \frac{\mathbf{c}(4,d,\text{mean})}{4!}\frac{6(d^2+7d-3)}{2^2(d+2)\sigma^4}\right] \quad (3.125)$$

3.17 Performance Analysis of ExROT in Multivariate

There has been performed two experiments to understand the performance of ExROT derived in multivariate KDE. For both the experiments, 12 normal mixture densities are selected for performance test, as used by (135) and by (43). The normal mixture equations make it possible to calculate the IMSE error between the expected probability at a point and that estimated using the estimated kernel bandwidth parameter. The densities are reported in the following Table 3.5. The first experiment analyzes the performance against varying type of density with fixing the number of samples to 50000. The second experiment analyzes the performance against varying number of samples for all the various densities. There are six different classes of bandwidth selection rules.

1. The class $\mathcal{Z}_1 = \{h_1^2\mathbf{I} : h_1 > 0\}$. There is only one bandwidth value estimated form the data.
2. The class $\mathcal{Z}_2 = diag(h_1^2, h_2^2) : h_1, h_2 > 0\}$. For 2-d signals, there are two different bandwidth values estimated along each dimension.
3. The class
$$\mathcal{Z}_2 = \left\{\begin{bmatrix} h_1 & h_{12} \\ h_{21} & h_2 \end{bmatrix} : h_1, h_2 > 0\right\}$$
.There are three different bandwidth parameter values estimated.
4. The class, with 'scaling', $\mathcal{C}_2 = h^2\mathbf{D} : h > 0$; where, $\mathbf{D}$ is a scaling matrix.
5. The class, with 'sphering', $\mathcal{C}_3 = h^2\mathbf{C} : h > 0$; where, $\mathbf{C}$ is a covariance matrix.

6. The 'hybrid' class
$$\mathcal{Y} = \left\{ \left[\begin{array}{cc} h_1 & \rho_{12}h_1h_2 \\ \rho_{12}h_1h_2 & h_2 \end{array} \right] : h_1, h_2 > 0 \right\}$$
.

For both the experiments, there were 7 types of bandwidth estimators implemented. The Silverman's ROT (136) is used with four different classes: $\mathcal{Z}_1, \mathcal{Z}_2, \mathcal{C}_2, \mathcal{C}_3$. The ExROT, derived in this chapter, is used with three different classes: $\mathcal{Z}_1, \mathcal{C}_2, \mathcal{C}_3$.

Table 3.5: Bivariate Normal Mixture Densities for the Performance Test of Bandwidth Selection Rules in multivariate Kernel Density Estimations

Density	$w_1N(\mu_{11}, \mu_{12}, \sigma^2_{11}, \sigma^2_{12}, \rho_1) + \ldots + w_kN(\mu_{k1}, \mu_{k2}, \sigma^2_{k1}, \sigma^2_{k2}, \rho_k)$
(1) Uncorrelated Normal	$N\left(0, 0, \frac{1}{4}, 1, 0\right)$
(2) Correlated normal	$N\left(0, 0, 1, 1, \frac{7}{10}\right)$
(3) Skewed	$\frac{1}{5}N\left(0, 0, 1, 1, 0\right) + \frac{1}{5}N\left(\frac{1}{2}, \frac{1}{2}, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, 0\right) + \frac{3}{5}N\left(\frac{13}{12}, \frac{13}{12}, \left(\frac{5}{9}\right)^2, \left(\frac{5}{9}\right)^2, 0\right)$
(4) Kurtotic	$\frac{2}{3}N\left(0, 0, 1, 4, \frac{1}{2}\right) + \frac{1}{3}N\left(0, 0, \left(\frac{2}{3}\right)^2, \left(\frac{1}{3}\right)^2, -\frac{1}{2}\right)$
(5) BimodalI	$\frac{1}{2}N\left(-1, 0, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, 0\right) + \frac{1}{2}N\left(1, 0, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, 0\right)$
(6) BimodalII	$\frac{1}{2}N\left(-\frac{3}{2}, 0, \left(\frac{1}{4}\right)^2, 1, 0\right) + \frac{1}{2}N\left(\frac{3}{2}, 0, \left(\frac{1}{4}\right)^2, 1, 0\right)$
(7) BimodalIII	$\frac{1}{2}N\left(-1, 1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, \frac{3}{5}\right) + \frac{1}{2}N\left(1, -1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, \frac{3}{5}\right)$
(8) BimodalIV	$\frac{1}{2}N\left(1, -1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, \frac{7}{10}\right) + \frac{1}{2}N\left(-1, 1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, 0\right)$
(9) TrimodalI	$\frac{9}{20}N\left(-\frac{6}{5}, \frac{6}{5}, \left(\frac{3}{5}\right)^2, \left(\frac{3}{5}\right)^2, \frac{3}{10}\right) + \frac{9}{20}N\left(\frac{6}{5}, -\frac{6}{5}, \left(\frac{3}{5}\right)^2, \left(\frac{3}{5}\right)^2, -\frac{3}{5}\right) + \frac{1}{10}N\left(0, 0, \left(\frac{1}{4}\right)^2, \left(\frac{1}{4}\right)^2, \frac{1}{5}\right)$
(10) TrimodalII	$\frac{1}{3}N\left(-\frac{6}{5}, 0, \left(\frac{3}{5}\right)^2, \left(\frac{3}{5}\right)^2, \frac{7}{10}\right) + \frac{1}{3}N\left(\frac{6}{5}, 0, \left(\frac{3}{5}\right)^2, \left(\frac{3}{5}\right)^2, \frac{7}{10}\right) + \frac{1}{3}N\left(0, 0, \left(\frac{3}{5}\right)^2, \left(\frac{3}{5}\right)^2, -\frac{7}{10}\right)$
(11) TrimodalIII	$\frac{3}{7}N\left(-1, 0, \left(\frac{3}{5}\right)^2, \left(\frac{7}{10}\right)^2, \frac{3}{5}\right) + \frac{3}{7}N\left(1, \frac{2\sqrt{3}}{3}, \left(\frac{3}{5}\right)^2, \left(\frac{7}{10}\right)^2, 0\right) + \frac{1}{7}N\left(1, -\frac{2\sqrt{3}}{3}, \left(\frac{3}{5}\right)^2, \left(\frac{7}{10}\right)^2, 0\right)$
(12) Quadrimodal	$\frac{1}{8}N\left(-1, 1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, \frac{2}{5}\right) + \frac{3}{8}N\left(-1, -1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, \frac{3}{5}\right) + \frac{1}{8}N\left(1, -1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, \frac{7}{10}\right)$ $+\frac{3}{8}N\left(1, 1, \left(\frac{2}{3}\right)^2, \left(\frac{2}{3}\right)^2, -\frac{1}{2}\right)$

Results of the first experiment are tabularized in Table 3.6. The table entries show mean of the IMSE error in 100 trials. The results show that for all the type of densities, ExROT out-performs ROT in it's respective class. The bold face values indicate best entry in the corresponding row. The ExROT with class $\mathcal{Z}_1$ implementation, out-performance other bandwidth estimators for densities 5 and 10 those are equi-variant. The 2-d density 5 is uncorrelated, but 10 is correlated. The ExROT with class $\mathcal{C}_2$ implementation out-performance other classes, when the test densities are having unequal variances and uncorrelated; such as, for density types 1, 3, and 6. But, the same estimators also out-performs 4, 7 and 8. For the remaining density types 2, 9, 11 and 12; the ExROT with class $\mathcal{C}_3$ implementation out-performs other classes.

Results of the second experiment are tabularized in Table 3.7 to Table 3.12. The entries in Table 3.7 to Table 3.9 show the bandwidth parameter estimated. The entries in Table 3.10 to Table 3.12 show the IMSE values due to the estimated bandwidth parameter. The IMSE error in Table 3.10 to Table 3.12 show that for a specific class type, *ExROT* out-performs *ROT* in all cases, for any *pdf* and any number of samples. The bold face values indicate best entry in the row. Also, there has been shown in bold faced italics two values, one from *ROT* and another from *ExROT*.

Table 3.6: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) against varying 2-d distributions using 50000 samples. The results show the mean IMSE (Integrated Mean square Error) of the 100 trials. The 2-d densities used are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	$rot(Z_1)$	$exrot(Z_1)$	$rot(C_2)$	$exrot(C_2)$	$rot(Z_2)$	$rot(C_3)$	$exrot(C_3)$
1	0.1307	0.0286	0.1282	**0.0255**	0.1282	0.1282	**0.0255**
2	0.1854	0.0305	0.1852	0.0305	0.1852	0.1078	**0.0217**
3	0.1392	0.0371	0.1896	**0.0287**	0.1896	0.1838	0.0289
4	0.3774	0.1428	0.3218	**0.1103**	0.3218	0.3226	0.1191
5	0.1519	**0.0261**	0.1424	0.0280	0.1424	0.1424	0.0280
6	0.6822	0.3279	0.6509	**0.2752**	0.6509	0.6509	**0.2752**
7	0.3829	0.0877	0.3256	**0.0625**	0.3256	0.3622	0.0849
8	0.3656	0.0883	0.3127	**0.0634**	0.3127	0.3394	0.0864
9	0.4593	0.1815	0.3993	0.1307	0.3993	0.3280	**0.1241**
10	0.2621	**0.0479**	0.2680	0.0570	0.2680	0.2644	0.0555
11	0.2511	0.0519	0.2271	0.0433	0.2271	0.2095	**0.0373**
12	0.2550	0.0638	0.2206	0.0462	0.2206	0.2167	**0.0446**

That from *ROT* denotes the best performance for that density from *ROT*; mostly using maximum number of 50000 samples. Corresponding to that performance of *ROT* at 50000 samples; there is one entry from *ExROT* that indicates maximum number of samples required to achieve atleast slightly better than that performance. The performance of ExROT, equivalent to the performance of ROT at 50000, is achieved for type 1 and type 4 densities using 500 samples; for type 3 density using 1000 samples; for type 9 density using 2000 samples and for all other remaining 8 density types just 200 samples are sufficient. To decide the best of the used estimators, mean and median of all the experiments are taken and reported in the last two rows of Table 3.12. With respect to both mean and median of the IMSE performance criteria, the ExROT with $\mathcal{C}_2$ is the best.

3.18 ExROT for Bandwidth Selection in Kernel Density Derivative Estimator

In general, the r^{th} derivative of multivariate density $f(\mathbf{x})$ using a v^{th} order kernel is estimated as under:

$$\hat{f}^{(r)}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{K}_{v,\mathbf{H}}^{(r)}(\mathbf{x}-\mathbf{x}_i) \qquad (3.126)$$

Usually a product kernel is used with whitening in all directions and $\mathbf{H} = h\mathbf{I}_{d\times d}$ as defined under:

$$\mathcal{K}^{(r)}_{v,\mathbf{H}}(\mathbf{x}-\mathbf{x}_i) = \prod_{j=1}^{d} \frac{1}{h}\mathcal{K}^{(r_j)}_{v}\left(\frac{x_{ji}-x_j}{h}\right) \tag{3.127}$$

where, $r = (r_1, r_2, \ldots, r_d)^T$ and the $(r_j)^{th}$ derivative corresponds to the fact that

$$f^{(r)}(x) = \frac{\partial^r f(x)}{\partial^{r_1}x_1\partial^{r_2}x_2\ldots\partial^{r_d}x_d}$$

The AMISE for density derivative is derived as in (56); but in vector notations similar to that for density estimation in Section 3.15. as under:

$$\begin{aligned} E\{\hat{f}^{(r)}(\mathbf{x})\} &= \int_{\mathbb{R}^d}\mathcal{K}(\mathbf{z})\left(f^{(r)}(\mathbf{x}) + \mathbf{k}(1+r,d)'(\mathbf{Hz}\otimes\mathbf{I}_d^{\otimes r})\right. \\ &\qquad \left.+\frac{1}{2}\mathbf{k}(2+r,d)'(\mathbf{Hz}\otimes\mathbf{I}_d^{\otimes r})^{\otimes 2} + O(\text{tr}(\mathbf{H}^{\otimes 2}))\right)d\mathbf{z} \qquad (3.128) \\ \Rightarrow \text{Bias}(\hat{f}^{(r)}(\mathbf{x})) &= E\{\hat{f}^{(r)}(\mathbf{x})\} - f^{(r)}(\mathbf{x}) \\ &= \frac{1}{v!}\mathbf{k}(v+r,d)'\left(\mathbf{H}^{\otimes v}\mathbf{m}_{\mathcal{K}^r}(v,d)\otimes\mathbf{I}_d^{\otimes 2r}\right) + O(\mathbf{H}^{\otimes 2}) \qquad (3.129) \\ \text{Var}(\hat{f}(\mathbf{x})) &= \frac{1}{N\text{det}(\mathbf{H})}f^{(r)}(\mathbf{x})\int_{\mathbb{R}^d}\left(\mathcal{K}^{(r)}(\mathbf{z})\right)^2 d\mathbf{z} + O\left(\frac{\det\mathbf{H}}{N}\right) \\ &\quad (\because \text{ assuming large } N, \text{ small } h) \qquad (3.130) \end{aligned}$$

Now, assuming $\mathbf{H}\in\mathcal{S}$ we get simplified AMISE criteria and the taking derivative with respect to h, we get the bandwidth parameter as under:

$$\text{AMISE}\left\{\hat{f}^{(r)}(\mathbf{x})\right\} = \frac{\mu_v(\mathcal{K}_v)}{(v!)^2}\int h^{2v}\left(\mathbf{k}(r+v,d)'\left(\mathbf{I}_d^{\otimes v}\boldsymbol{\delta}_v\otimes\mathbf{I}_d^{\otimes 2r}\right)\right)^2 d\mathbf{x} + \frac{1}{Nh^{d+2r}}R(\mathcal{K}) \tag{3.131}$$

$$h_{AMISE} = \left(\frac{(v!)^2}{2v}\frac{(d+2r)R(\mathcal{K})}{\mu_v^2(\mathcal{K})R(\nabla^{(v)}f^{(r)}(\mathbf{x}))N}\right)^{\frac{1}{d+2v+2r}} \tag{3.132}$$

$$\text{or } h_{AMISE} = [CN]^{-\frac{1}{d+2v+2r}} \quad \text{where, } C = \frac{\mu_v^2(\mathcal{K})R(\nabla^{(v)}f^{(r)}(\mathbf{x})}{(d+2r)R(\mathcal{K})} \tag{3.133}$$

With this, the ExROT for gradient density can be derived and needs $R(\mathbf{k}(r+v,d))$ definition. With Gaussian kernel, the first order derivative of 1-dimensional density; i.e., $v = 2$, $r = 1$ and

$d = 1$; the required parameter $R(\mathbf{k}(r+v,d))$ can be derived as under.

$$\begin{aligned}
R(f^{(3)}(x)) &= \int_{\infty}^{\infty} \frac{1}{2\pi\sigma}\exp\left(-(z)^2\right)\left[-\frac{1}{\sigma^3}H_3(z) + \frac{k_3}{3!\sigma^6}H_6(z) - \frac{k_4}{4!\sigma^7}H_7(z)\right]^2 dz \\
&= \frac{1}{2\sqrt{\pi}\sigma}\left[\frac{1}{\sigma^6}\frac{15}{8} + \frac{1}{\sigma^{12}}\left(\frac{k_3}{3!}\right)^2\frac{10395}{64} + \frac{1}{\sigma^{14}}\left(\frac{k_4}{4!}\right)^2\frac{135135}{2^7} + \frac{1}{\sigma^9}\left(\frac{k_4}{4!}\right)\frac{945}{32}\right] \\
\Rightarrow h_{GC} &= \sigma(CN)^{-\frac{1}{7}} \qquad (3.134) \\
\text{where, } C &= \frac{1.875}{\sigma^6} + 4.5117\frac{k_3^2}{\sigma^{12}} + 1.8329\frac{k_4^2}{\sigma^{14}} + 2.4609\frac{k_4}{\sigma^6} \qquad (3.135)
\end{aligned}$$

Similarly, let for example, with Gaussian kernel (i.e. $v = 2$) and *i.i.d.* components; the bandwidth parameter h_{GC} for first of derivative (i.e. $r = 1$) of a d-dimensional density $f(\mathbf{x})$ be derived. The required roughness $R(\nabla^2 f^1(\mathbf{x}))$ is obtained as under:

$$R(\nabla^2 f^{(1)}(\mathbf{x})) = \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\frac{\partial^2 f^{(1)}(\mathbf{x})}{\partial x_i^2}\right]^2 \qquad (3.136)$$

$$\approx \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\left[G^{(1)}(\mathbf{x}) - \frac{(\mathbf{c}(3,d)\otimes\mathbf{I}_d)'}{3!}G^{(4)}(\mathbf{x}) + \frac{(\mathbf{c}(4,d)\otimes\mathbf{I}_d)'}{4!}G^{(5)}(\mathbf{x})\right]\right]^2 d\mathbf{x} \qquad (3.137)$$

$$\begin{aligned}
= \int_{\mathbb{R}^d}\Bigg(&\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}G^{(1)}(\mathbf{x})\right]^2 + \left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{(\mathbf{c}(3,d)\otimes\mathbf{I}_d)'G^{(4)}(\mathbf{x})}{3!}\right]^2 \\
&+ \left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{(\mathbf{c}(4,d)\otimes\mathbf{I}_d)'G^{(5)}(\mathbf{x})}{4!}\right]^2 \\
&-2\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}G^{(1)}(\mathbf{x})\right]\left[\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\frac{(\mathbf{c}(3,d)\otimes\mathbf{I}_d)'G^{(4)}(\mathbf{x})}{3!}\right] \\
&-2\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{(\mathbf{c}(3,d)\otimes\mathbf{I}_d)'G^{(4)}(\mathbf{x})}{3!}\right]\left[\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\frac{(\mathbf{c}(4,d)\otimes\mathbf{I}_d)'G^{(5)}(\mathbf{x})}{4!}\right] \\
&+2\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}G^{(1)}(\mathbf{x})\right]\left[\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\frac{(\mathbf{c}(4,d)\otimes\mathbf{I}_d)'G^{(5)}(\mathbf{x})}{4!}\right]\Bigg) d\mathbf{x} \qquad (3.138) \\
= Q_1 + Q_2 + Q_3 - Q_4 - Q_5 + Q_6 \qquad (3.139)
\end{aligned}$$

where, $Q_i, i = 1:6$ are the symbols for corresponding terms.

The simplification of these terms is obtained knowing $G(\mathbf{x}) = \prod_{i=1}^{d} G(x_i)$ and assuming all third order and fourth order cross-moments to be zero. Also, there have been used the symbol

$c_i(3,d)$ as the third order cumulant (i.e. skewness) of x_i, $\mathbf{c}(3,d,\text{mean}) = \frac{1}{d}\sum_{i=1}^{d} c_i(3,d)$ as the mean of the skewness and $\mathbf{c}(4,d,\text{mean}) = \frac{1}{d}\sum_{i=1}^{d} c_i(4,d)$ as the mean of the kurtosis.

$$
\begin{aligned}
Q_1 &= \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}G^{(1)}(\mathbf{x})\right]^2 d\mathbf{x} = \int_{\mathbb{R}^d}\left[\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{\partial}{\partial x_j}G(\mathbf{x})\right]^2 d\mathbf{x} && (3.140)\\
&= \int_{\mathbb{R}^d}\left\{\left[\sum_{i=1}^{d}\frac{\partial^3}{\partial x_i^3}G(\mathbf{x})\right]^2 + \left[\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\sum_{j=1,j\neq i}^{d}\frac{\partial}{\partial x_j}G(\mathbf{x})\right]^2\right.\\
&\quad\left. +2\left[\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{\partial^3}{\partial x_i^3}G(\mathbf{x})\left(\frac{\partial^2}{\partial x_j^2}\sum_{k=1,k\neq j}^{d}\frac{\partial}{\partial x_k}G(\mathbf{x})\right)\right]\right\} d\mathbf{x} && (3.141)\\
&= \int_{\mathbb{R}^d}\left\{\sum_{i=1}^{d}\left[\mathbf{H}_3(x_i)G(\mathbf{x})\right]^2 + \left[\sum_{j=1}^{d}\sum_{k=1,k\neq j}^{d}\mathbf{H}_2(x_j)\mathbf{H}_1(x_k)G(\mathbf{x})\right]^2\right.\\
&\quad\left. +2\left[\sum_{i=1}^{d}\mathbf{H}_3(x_i)G(\mathbf{x})\left(\sum_{j=1}^{d}\sum_{k=1,k\neq j}^{d}\mathbf{H}_2(x_j)\mathbf{H}_1(x_k)G(\mathbf{x})\right)\right]\right\} d\mathbf{x} && (3.142)\\
&= \frac{6!!d}{2^{d+3}\pi^{d/2}\sigma^{d+6}} + \frac{4!!2!!d(d-1) + 2!!2!!2!!d(d-1)(d-2) + 2 * 4!!2!!d(d-1)}{2^{d+3}\pi^{d/2}\sigma^{d+6}} && (3.143)\\
&= \frac{d(d^2+6d+8)}{2^{d+3}\pi^{d/2}\sigma^{d+6}} && (3.144)
\end{aligned}
$$

$$Q_2 = \int_{\mathbb{R}^d} \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \frac{(\mathbf{c}(3,d)\otimes \mathbf{I}_d)'\, G^{(4)}(\mathbf{x})}{3!}\right]^2 d\mathbf{x} \tag{3.145}$$

$$= \int_{\mathbb{R}^d} \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \sum_{j=1}^{d} \frac{\partial}{\partial x_j} \frac{(\mathbf{c}(3,d))'\, G^{(3)}(\mathbf{x})}{3!}\right]^2 d\mathbf{x} \tag{3.146}$$

$$\begin{aligned}
= \int_{\mathbb{R}^d} \Bigg\{ & \sum_{i=1}^{d} \left[\frac{c_i(3,d)}{3!}\mathbf{H}_6(x_i)G(\mathbf{x})\right]^2 + \left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d} \frac{c_j(3,d)(\mathbf{x})}{3!}\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x})\right]^2 \\
&+ \left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d} \frac{c_k(3,d)(\mathbf{x})}{3!}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right]^2 \\
&+ \left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d} \frac{c_j(3,d)(\mathbf{x})}{3!}\mathbf{H}_5(x_i)\mathbf{H}_1(x_j)G(\mathbf{x})\right]^2 \\
&+2\left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d} \frac{c_i(3,d)c_j(3,d)}{3!3!}\mathbf{H}_6(x_i)G(\mathbf{x})\left(\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x})\right)\right] \\
&+2\left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d} \frac{c_i(3,d)c_k(3,d)}{3!3!}\mathbf{H}_6(x_i)G(\mathbf{x})\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right] \\
&+2\left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d} \frac{c_i(3,d)c_j(3,d)}{3!3!}\mathbf{H}_6(x_i)G(\mathbf{x})\left(\mathbf{H}_5(x_i)\mathbf{H}_1(x_j)G(\mathbf{x})\right)\right] \\
&+2\left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d} \frac{c_j(3,d)c_k(3,d)}{3!3!}\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x})\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right] \\
&+2\left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d} \frac{c_j(3,d)c_k(3,d)}{3!3!}\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x})\mathbf{H}_5(x_i)\mathbf{H}_1(x_j)G(\mathbf{x})\right] \\
&+2\left[\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d} \frac{c_j(3,d)c_k(3,d)}{3!3!}\mathbf{H}_5(x_i)\mathbf{H}_1(x_j)G(\mathbf{x})\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right]\Bigg\} d\mathbf{x}
\end{aligned} \tag{3.147}$$

$$\begin{aligned}
= \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2} & \left[\frac{12!!d + 6!!6!!d(d-1)}{2^{d+6}\pi^{d/2}\sigma^{d+12}} + \frac{6!!6!!d(d-1)}{2^{d+6}\pi^{d/2}\sigma^{d+12}} + \frac{4!!2!!6!!d(d-1)(d-2)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right. \\
&\left. + \frac{10!!2!!d(d-1)}{2^{d+6}\pi^{d/2}\sigma^{d+12}} + \frac{2*8!!4!!d(d-1)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right] \quad (\because \text{assuming } c_i(3,d) = \mathbf{c}(3,d,\text{mean}))
\end{aligned} \tag{3.148}$$

$$= \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\left[\frac{5d(9d^2+374d+1528)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right] \tag{3.149}$$

$$Q_3 = \int_{\mathbb{R}^d} \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \frac{(\mathbf{c}(4,d) \otimes \mathbf{I}_d)' \, G^{(5)}(\mathbf{x})}{4} \right]^2 d\mathbf{x} \tag{3.150}$$

$$= \int_{\mathbb{R}^d} \left[\sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} \sum_{j=1}^{d} \frac{\partial}{\partial x_j} \frac{(\mathbf{c}(4,d))' \, G^{(4)}(\mathbf{x})}{4!} \right]^2 d\mathbf{x} \tag{3.151}$$

$$\begin{aligned}
&= \int_{\mathbb{R}^d} \Bigg\{ \left[\sum_{i=1}^{d} \frac{c_i(4,d)}{4!} \mathbf{H}_7(x_i) G(\mathbf{x}) \right]^2 + \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \frac{c_j(4,d)(\mathbf{x})}{4!} \mathbf{H}_3(x_i) \mathbf{H}_4(x_j) G(\mathbf{x}) \right]^2 \\
&+ \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \sum_{k=1, k \neq i,j}^{d} \frac{c_k(4,d)(\mathbf{x})}{4!} \mathbf{H}_2(x_i) \mathbf{H}_1(x_j) \mathbf{H}_4(x_k) G(\mathbf{x}) \right]^2 \\
&+ \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \frac{c_j(4,d)(\mathbf{x})}{4!} \mathbf{H}_6(x_i) \mathbf{H}_1(x_j) G(\mathbf{x}) \right]^2 \\
&+2 \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \frac{c_i(4,d) c_j(4,d)}{4!4!} \mathbf{H}_7(x_i) G(\mathbf{x}) \left(\mathbf{H}_3(x_i) \mathbf{H}_4(x_j) G(\mathbf{x}) \right) \right] \\
&+2 \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \sum_{k=1, k \neq i,j}^{d} \frac{c_i(4,d) c_k(4,d)}{4!4!} \mathbf{H}_7(x_i) G(\mathbf{x}) \mathbf{H}_2(x_i) \mathbf{H}_1(x_j) \mathbf{H}_4(x_k) G(\mathbf{x}) \right] \\
&+2 \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \frac{c_i(4,d) c_j(4,d)}{4!4!} \mathbf{H}_7(x_i) G(\mathbf{x}) \left(\mathbf{H}_6(x_i) \mathbf{H}_1(x_j) G(\mathbf{x}) \right) \right] \\
&+2 \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \sum_{k=1, k \neq i,j}^{d} \frac{c_j(4,d) c_k(4,d)}{4!4!} \mathbf{H}_3(x_i) \mathbf{H}_4(x_j) G(\mathbf{x}) \mathbf{H}_2(x_i) \mathbf{H}_1(x_j) \mathbf{H}_4(x_k) G(\mathbf{x}) \right] \\
&+2 \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \frac{c_i(4,d) c_j(4,d)}{4!4!} \mathbf{H}_3(x_i) \mathbf{H}_4(x_j) G(\mathbf{x}) \left(\mathbf{H}_6(x_i) \mathbf{H}_1(x_j) G(\mathbf{x}) \right) \right] \\
&+2 \left[\sum_{i=1}^{d} \sum_{j=1, j \neq i}^{d} \sum_{k=1, k \neq i,j}^{d} \frac{c_j(4,d) c_k(4,d)}{4!4!} \mathbf{H}_6(x_i) \mathbf{H}_1(x_j) G(\mathbf{x}) \mathbf{H}_2(x_i) \mathbf{H}_1(x_j) \mathbf{H}_4(x_k) G(\mathbf{x}) \right] \Bigg\} d\mathbf{x}
\end{aligned} \tag{3.152}$$

$$\begin{aligned}
&= \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2} \Bigg[\frac{14!!d + (6!!8!!d(d-1) + 6!!4!!4!!d(d-1)(d-2))}{2^{d+7} \pi^{d/2} \sigma^{d+14}} \\
&+ \frac{(4!!2!!8!!d(d-1)(d-2) + 2!!2!!4!!2!!4!!d(d-1)(d-2)(d-3)(d-4))}{2^{d+7} \pi^{d/2} \sigma^{d+14}} \\
&+ \frac{(12!!2!!d(d-1) + 6!!2!!6!!d(d-1)(d-2)) + 2 * 10!!4!!d(d-1)}{2^{d+7} \pi^{d/2} \sigma^{d+14}} \\
&+ \frac{2 * 8!!2!!4!!d(d-1)(d-2)}{2^{d+7} \pi^{d/2} \sigma^{d+14}} \Bigg] \quad (\because \text{assuming } c_i(4,d) = \mathbf{c}(4,d,\text{mean}))
\end{aligned} \tag{3.153}$$

$$= \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2} \left[\frac{9d(d^4 - 10d^3 + 180d^2 + 1475d + 13369)}{2^{d+7} \pi^{d/2} \sigma^{d+14}} \right] \tag{3.154}$$

$$
\begin{aligned}
Q_4 &= \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d}\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{\partial}{\partial x_j}G(\mathbf{x})\right]\left[\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\sum_{k=1}^{d}\frac{\partial}{\partial x_k}\frac{\mathbf{c}(3,d)'G^{(3)}(\mathbf{x})}{3!}\right]d\mathbf{x} && (3.155)\\
&= \int_{\mathbb{R}^d}\Bigg\{2\left[\sum_{i=1}^{d}\mathbf{H}_3(x_i)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)G(\mathbf{x})\right]\\
&\quad\left[\sum_{i=1}^{d}\frac{c_i(3,d)}{3!}\mathbf{H}_6(x_i)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_2(x_i)\mathbf{H}_4(x_j)G(\mathbf{x})\right.\\
&\quad+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_5(x_i)\mathbf{H}_1(x_j)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x})\\
&\quad\left.+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\right]\Bigg\}d\mathbf{x} && (3.156)\\
&= 0 && (3.157)
\end{aligned}
$$

$$
\begin{aligned}
Q_5 &= \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{\partial}{\partial x_j}\frac{\mathbf{c}(3,d)'G^{(3)}(\mathbf{x})}{3!}\right]\left[\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\frac{\partial}{\partial x_j}\frac{\mathbf{c}(4,d)'G^{(4)}(\mathbf{x})}{4!}\right]d\mathbf{x} && (3.158)\\
&= \int_{\mathbb{R}^d}\Bigg\{2\left[\sum_{i=1}^{d}\frac{c_i(3,d)}{3!}\mathbf{H}_6(x_i)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_2(x_i)\mathbf{H}_4(x_j)G(\mathbf{x})\right.\\
&\quad+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_3(x_k)G(\mathbf{x})\\
&\quad\left.+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_5(x_i)\mathbf{H}_1(x_j)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(3,d)}{3!}\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x})\right]\\
&\quad\left[\sum_{i=1}^{d}\frac{c_i(4,d)}{4!}\mathbf{H}_7(x_i)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(4,d)}{4!}\mathbf{H}_2(x_i)\mathbf{H}_5(x_j)G(\mathbf{x})\right.\\
&\quad+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(4,d)}{4!}\mathbf{H}_6(x_i)\mathbf{H}_1(x_j)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_k(4,d)}{4!}\mathbf{H}_3(x_i)\mathbf{H}_4(x_j)G(\mathbf{x})\\
&\quad\left.+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d}\frac{c_k(4,d)}{4!}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_4(x_k)G(\mathbf{x})\right]\Bigg\}d\mathbf{x} && (3.159)\\
&= 0 && (3.160)
\end{aligned}
$$

$$Q_6 = \int_{\mathbb{R}^d} 2\left[\sum_{i=1}^{d}\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}\frac{\partial}{\partial x_j}G(\mathbf{x})\right]\left[\sum_{j=1}^{d}\frac{\partial^2}{\partial x_j^2}\sum_{k=1}^{d}\frac{\partial}{\partial x_k}\frac{\mathbf{c}(4,d)'G^{(4)}(\mathbf{x})}{4!}\right]d\mathbf{x} \tag{3.161}$$

$$\begin{aligned}
&= \int_{\mathbb{R}^d}\Bigg\{2\left[\sum_{i=1}^{d}\mathbf{H}_3(x_i)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)G(\mathbf{x})\right] \\
&\left[\sum_{i=1}^{d}\frac{c_i(4,d)}{4!}\mathbf{H}_7(x_i)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_j(4,d)}{4!}\mathbf{H}_2(x_i)\mathbf{H}_5(x_j)G(\mathbf{x})\right. \\
&+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_j(4,d)}{4!}\mathbf{H}_6(x_i)\mathbf{H}_1(x_j)G(\mathbf{x}) + \sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\frac{c_j(4,d)}{4!}\mathbf{H}_3(x_i)\mathbf{H}_3(x_j)G(\mathbf{x}) \\
&\left.+\sum_{i=1}^{d}\sum_{j=1,j\neq i}^{d}\sum_{k=1,k\neq i,j}^{d}\frac{c_k(4,d)}{4!}\mathbf{H}_2(x_i)\mathbf{H}_1(x_j)\mathbf{H}_4(x_k)G(\mathbf{x})\right]\Bigg\}\,d\mathbf{x}
\end{aligned} \tag{3.162}$$

$$= \frac{\mathbf{c}(4,d,mean)}{4!}\frac{2*10!!d + 4!!6!!d(d-1) + 4!!2!!4!!d(d-1)(d-2) + 8!!2!!d(d-1)}{2^{d+5}\pi^{d/2}\sigma^{d+10}} \tag{3.163}$$

$$= \frac{\mathbf{c}(4,d,mean)}{4!}\frac{3d(3d^2+41d+586)}{2^{d+5}\pi^{d/2}\sigma^{d+10}} \tag{3.164}$$

Combining above simplifications, the formula for $R(\nabla^2 f^{(1)}(\mathbf{x}))$ can be derived as under:

$$\begin{aligned}
R(f^{(3)}(\mathbf{x})) = {} & \frac{d(d+4)(d+2)}{2^{d+3}\pi^{d/2}\sigma^{d+6}} + \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\left[\frac{5d(9d^2+374d+1528)}{2^{d+6}\pi^{d/2}\sigma^{d+12}}\right] \\
& + \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2}\left[\frac{9d(d^4-10d^3+180d^2+1475d+13369)}{2^{d+7}\pi^{d/2}\sigma^{d+14}}\right] \\
& + \frac{\mathbf{c}(4,d,\text{mean})}{(4!)}\left[\frac{3d(3d^2+41d+586)}{2^{d+5}\pi^{d/2}\sigma^{d+10}}\right]
\end{aligned} \tag{3.165}$$

With $R(\mathcal{K}) = 2^{-d}\pi^{-d/2}$, $\mu_2(\mathcal{K}) = 1$ and using Equation (3.133); the bandwidth parameter using GCA based ExROT (h_i(ExROT)) can be given as under:

$$h_{\text{ExROT}} = [CN]^{-\frac{1}{d+6}} \quad \text{where, } C = \frac{\mu_2^2(\mathcal{K})R(\nabla^{(2)}f^{(1)}(\mathbf{x}))}{(d+2)R(\mathcal{K})} \tag{3.166}$$

$$\begin{aligned}
C = \frac{d(d+4)}{2^3\sigma^{d+6}}\Bigg[1 & + \frac{\mathbf{c}^2(3,d,\text{mean})}{(3!)^2}\frac{5(9d^2+374d+1528)}{2^3\sigma^6(d+4)(d+2)} \\
& + \frac{\mathbf{c}^2(4,d,\text{mean})}{(4!)^2}\frac{9(d^4-10d^3+180d^2+1475d+13369)}{2^4\sigma^8(d+4)(d+2)} \\
& + \frac{\mathbf{c}(4,d,\text{mean})}{(4!)}\frac{3(3d^2+41d+586)}{2^2\sigma^4(d+4)(d+2)}\Bigg]
\end{aligned} \tag{3.167}$$

3.19 Conclusion and Future directions

The chapter addresses the issue of bandwidth selection in KDE for both - univariate and multivariate. There has been proposed Gram-Charlier A Series based *Extended Rule-of-Thumb* (ExROT) on the assumption that the density being estimated is near Gaussian. The performance analysis of ExROT is done using standard test set for univariate density estimation. The results show that in all nongaussian unimodal density estimation cases - skewed or kurtotic or with outlier - ExROT has performed better than ROT. This is achieved at computation comparable to ROT and too small compare to the ϵ-exact solve-the-equation plug-in rule. The ExROT has also outperformed ROT in some of the skewed multimodal density estimation - skewed bimodal, claw, Asymmetric claw. The Gram-Charlier A Series based ExROT for bandwidth selection is also obtained for multivariate KDE and multivariate density derivative estimations.

The chapter serves as a particular demonstration to a more generalized class of bandwidth selection rules based on PDF approximations through infinite series. The PDF approximation through infinite series expansion is a well established area and there exist many such approximations based on various reference PDFs. As the first results are encouraging, many such rules can be developed.

Table 3.7: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) using varying number of samples. The results show mean bandwidth of 100 trials. The 2-d distributions are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	**nsam**	$hrot(Z_1)$	$hexrot(Z_1)$	$hrot(C_2)$	$hexrot(C_2)$	$hrot(Z_2)$		$hrot(C_3)$	$hexrot(C_3)$
1	100	0.8854	0.2179	1.4736	0.4571	0.7405	1.4310	1.4736	0.4569
	200	0.8230	0.2196	1.3128	0.4091	0.6540	1.3143	1.3128	0.4091
	500	0.7088	0.2061	1.1269	0.3540	0.5619	1.1314	1.1269	0.3540
	1000	0.6276	0.1882	1.0040	0.3157	0.5013	1.0041	1.0040	0.3157
	2000	0.5591	0.1717	0.8944	0.2815	0.4463	0.8948	0.8944	0.2815
	5000	0.4814	0.1499	0.7678	0.2418	0.3838	0.7693	0.7678	0.2418
	10000	0.4276	0.1338	0.6840	0.2154	0.3417	0.6843	0.6840	0.2154
	20000	0.3806	0.1194	0.6094	0.1919	0.3048	0.6090	0.6094	0.1919
	50000	0.3267	0.1027	0.5231	0.1647	0.2615	0.5229	0.5231	0.1647
2	100	1.4807	0.4571	1.4736	0.4572	1.4723	1.4729	1.4736	0.4585
	200	1.3284	0.4134	1.3128	0.4091	1.3143	1.3234	1.3128	0.4098
	500	1.1209	0.3507	1.1269	0.3527	1.1264	1.1205	1.1269	0.3532
	1000	0.9989	0.3143	1.0040	0.3159	1.0030	0.9994	1.0040	0.3159
	2000	0.8962	0.2820	0.8944	0.2814	0.8952	0.8952	0.8944	0.2814
	5000	0.7675	0.2417	0.7678	0.2418	0.7670	0.7681	0.7678	0.2417
	10000	0.6840	0.2153	0.6840	0.2153	0.6841	0.6838	0.6840	0.2154
	20000	0.6095	0.1920	0.6094	0.1919	0.6095	0.6094	0.6094	0.1919
	50000	0.5238	0.1650	0.5231	0.1647	0.5234	0.5234	0.5231	0.1647
3	100	0.9625	0.1940	1.4736	0.4147	1.1813	1.1931	1.4736	0.4173
	200	0.8715	0.1748	1.3128	0.3671	1.0757	1.0605	1.3128	0.3684
	500	0.7458	0.1470	1.1269	0.3147	0.9121	0.9202	1.1269	0.3160
	1000	0.6678	0.1316	1.0040	0.2802	0.8145	0.8225	1.0040	0.2814
	2000	0.5946	0.1167	0.8944	0.2495	0.7287	0.7295	0.8944	0.2504
	5000	0.5114	0.1003	0.7678	0.2140	0.6270	0.6261	0.7678	0.2148
	10000	0.4561	0.0896	0.6840	0.1907	0.5584	0.5587	0.6840	0.1915
	20000	0.4051	0.0793	0.6094	0.1699	0.4970	0.4967	0.6094	0.1706
	50000	0.3481	0.0682	0.5231	0.1459	0.4267	0.4267	0.5231	0.1464
4	100	2.6376	0.8273	1.4736	0.4312	1.3349	2.4390	1.4736	0.4372
	200	2.2746	0.7130	1.3128	0.3842	1.1840	2.1328	1.3128	0.3889
	500	2.0053	0.6291	1.1269	0.3297	1.0206	1.8632	1.1269	0.3333
	1000	1.7563	0.5509	1.0040	0.2947	0.9070	1.6437	1.0040	0.2982
	2000	1.5715	0.4930	0.8944	0.2620	0.8082	1.4687	0.8944	0.2653
	5000	1.3502	0.4236	0.7678	0.2250	0.6924	1.2624	0.7678	0.2278
	10000	1.2030	0.3774	0.6840	0.2005	0.6174	1.1245	0.6840	0.2030
	20000	1.0731	0.3366	0.6094	0.1787	0.5499	1.0027	0.6094	0.1809
	50000	0.9207	0.2889	0.5231	0.1533	0.4722	0.8604	0.5231	0.1552
5	100	1.3906	0.4221	1.4736	0.4558	1.7676	0.9798	1.4736	0.4558
	200	1.2365	0.3787	1.3128	0.4083	1.5734	0.8749	1.3128	0.4082
	500	1.0702	0.3303	1.1269	0.3515	1.3577	0.7532	1.1269	0.3515
	1000	0.9491	0.2936	1.0040	0.3137	1.2060	0.6714	1.0040	0.3137
	2000	0.8445	0.2611	0.8944	0.2794	1.0744	0.5966	0.8944	0.2794
	5000	0.7257	0.2247	0.7678	0.2400	0.9231	0.5119	0.7678	0.2400
	10000	0.6459	0.2000	0.6840	0.2139	0.8219	0.4560	0.6840	0.2139
	20000	0.5753	0.1782	0.6094	0.1906	0.7323	0.4060	0.6094	0.1906
	50000	0.4942	0.1531	0.5231	0.1636	0.6289	0.3486	0.5231	0.1636

Table 3.8: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) using varying number of samples. The results show mean bandwidth of 100 trials. The 2-d distributions are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	**nsam**	$hrot(Z_1)$	$hexrot(Z_1)$	$hrot(C_2)$	$hexrot(C_2)$	$hrot(Z_2)$		$hrot(C_3)$	$hexrot(C_3)$
6	100	2.4476	0.7734	1.4736	0.4376	2.2501	1.4627	1.4736	0.4377
	200	2.1851	0.6904	1.3128	0.3906	2.0006	1.3155	1.3128	0.3907
	500	1.8710	0.5911	1.1269	0.3356	1.7190	1.1228	1.1269	0.3356
	1000	1.6686	0.5272	1.0040	0.2993	1.5265	1.0098	1.0040	0.2993
	2000	1.4813	0.4680	0.8944	0.2667	1.3601	0.8943	0.8944	0.2667
	5000	1.2708	0.4015	0.7678	0.2290	1.1673	0.7673	0.7678	0.2290
	10000	1.1336	0.3582	0.6840	0.2040	1.0402	0.6845	0.6840	0.2040
	20000	1.0095	0.3190	0.6094	0.1818	0.9267	0.6094	0.6094	0.1818
	50000	0.8663	0.2737	0.5231	0.1560	0.7955	0.5230	0.5231	0.1560
7	100	2.1401	0.6777	1.4736	0.4559	1.7799	1.7657	1.4736	0.4626
	200	1.9179	0.6075	1.3128	0.4069	1.5883	1.5829	1.3128	0.4142
	500	1.6350	0.5180	1.1269	0.3496	1.3540	1.3603	1.1269	0.3565
	1000	1.4534	0.4605	1.0040	0.3117	1.2086	1.2070	1.0040	0.3180
	2000	1.2930	0.4097	0.8944	0.2777	1.0739	1.0768	0.8944	0.2834
	5000	1.1101	0.3517	0.7678	0.2384	0.9233	0.9230	0.7678	0.2434
	10000	0.9882	0.3131	0.6840	0.2125	0.8226	0.8217	0.6840	0.2168
	20000	0.8798	0.2788	0.6094	0.1893	0.7323	0.7321	0.6094	0.1932
	50000	0.7556	0.2394	0.5231	0.1625	0.6289	0.6284	0.5231	0.1658
8	100	2.1418	0.6783	1.4736	0.4563	1.7749	1.7730	1.4736	0.4594
	200	1.8791	0.5954	1.3128	0.4065	1.5660	1.5729	1.3128	0.4121
	500	1.6260	0.5152	1.1269	0.3498	1.3548	1.3517	1.1269	0.3548
	1000	1.4519	0.4600	1.0040	0.3117	1.2069	1.2075	1.0040	0.3164
	2000	1.2913	0.4092	0.8944	0.2777	1.0732	1.0760	0.8944	0.2821
	5000	1.1084	0.3512	0.7678	0.2384	0.9233	0.9216	0.7678	0.2421
	10000	0.9885	0.3132	0.6840	0.2124	0.8221	0.8224	0.6840	0.2158
	20000	0.8801	0.2789	0.6094	0.1893	0.7324	0.7322	0.6094	0.1922
	50000	0.7558	0.2395	0.5231	0.1625	0.6287	0.6287	0.5231	0.1650
9	100	2.4181	0.7652	1.4736	0.4520	1.8801	1.8917	1.4736	0.4476
	200	2.1370	0.6764	1.3128	0.4028	1.6790	1.6693	1.3128	0.3991
	500	1.8323	0.5799	1.1269	0.3462	1.4388	1.4346	1.1269	0.3438
	1000	1.6291	0.5156	1.0040	0.3087	1.2790	1.2786	1.0040	0.3061
	2000	1.4575	0.4613	0.8944	0.2750	1.1401	1.1433	0.8944	0.2731
	5000	1.2482	0.3951	0.7678	0.2361	0.9783	0.9795	0.7678	0.2344
	10000	1.1130	0.3523	0.6840	0.2103	0.8725	0.8725	0.6840	0.2088
	20000	0.9918	0.3139	0.6094	0.1874	0.7772	0.7775	0.6094	0.1861
	50000	0.8507	0.2692	0.5231	0.1609	0.6671	0.6670	0.5231	0.1597
10	100	1.2232	0.3566	1.4736	0.4575	1.6797	0.8804	1.4736	0.4563
	200	1.1137	0.3331	1.3128	0.4101	1.5139	0.7922	1.3128	0.4090
	500	0.9457	0.2850	1.1269	0.3536	1.2948	0.6734	1.1269	0.3525
	1000	0.8489	0.2575	1.0040	0.3152	1.1581	0.6023	1.0040	0.3143
	2000	0.7529	0.2282	0.8944	0.2808	1.0293	0.5359	0.8944	0.2800
	5000	0.6440	0.1953	0.7678	0.2412	0.8814	0.4603	0.7678	0.2405
	10000	0.5747	0.1745	0.6840	0.2149	0.7860	0.4104	0.6840	0.2143
	20000	0.5119	0.1554	0.6094	0.1915	0.7002	0.3655	0.6094	0.1909
	50000	0.4393	0.1334	0.5231	0.1644	0.6008	0.3139	0.5231	0.1639

Table 3.9: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) using varying number of samples. The results show mean bandwidth of 100 trials. The 2-d distributions are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	**nsam**	$hrot(Z_1)$	$hexrot(Z_1)$	$hrot(C_2)$	$hexrot(C_2)$	$hrot(Z_2)$		$hrot(C_3)$	$hexrot(C_3)$
11	100	1.8372	0.5796	1.4736	0.4561	1.7098	1.5747	1.4736	0.4504
	200	1.6442	0.5194	1.3128	0.4077	1.5242	1.4100	1.3128	0.4029
	500	1.4036	0.4437	1.1269	0.3506	1.3069	1.2057	1.1269	0.3469
	1000	1.2480	0.3945	1.0040	0.3125	1.1630	1.0736	1.0040	0.3084
	2000	1.1071	0.3500	0.8944	0.2785	1.0331	0.9555	0.8944	0.2751
	5000	0.9526	0.3012	0.7678	0.2392	0.8879	0.8212	0.7678	0.2362
	10000	0.8482	0.2682	0.6840	0.2130	0.7917	0.7304	0.6840	0.2104
	20000	0.7565	0.2392	0.6094	0.1898	0.7051	0.6518	0.6094	0.1875
	50000	0.6488	0.2051	0.5231	0.1629	0.6052	0.5590	0.5231	0.1609
12	100	2.1622	0.6847	1.4736	0.4551	1.7824	1.7836	1.4736	0.4573
	200	1.8991	0.6016	1.3128	0.4067	1.5790	1.5769	1.3128	0.4089
	500	1.6313	0.5168	1.1269	0.3498	1.3513	1.3596	1.1269	0.3516
	1000	1.4536	0.4606	1.0040	0.3116	1.2084	1.2074	1.0040	0.3134
	2000	1.2953	0.4104	0.8944	0.2777	1.0763	1.0763	0.8944	0.2793
	5000	1.1101	0.3517	0.7678	0.2385	0.9234	0.9229	0.7678	0.2398
	10000	0.9892	0.3134	0.6840	0.2125	0.8222	0.8230	0.6840	0.2136
	20000	0.8803	0.2789	0.6094	0.1893	0.7327	0.7320	0.6094	0.1903
	50000	0.7556	0.2394	0.5231	0.1625	0.6286	0.6287	0.5231	0.1634

Table 3.10: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) using varying number of samples. The results show the mean IMSE (Integrated Mean Square Error) error of 100 trials. The 2-d distributions are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	**nsam**	$rot(Z_1)$	$exrot(Z_1)$	$rot(C_2)$	$exrot(C_2)$	$rot(Z_2)$	$rot(C_3)$	$exrot(C_3)$
1	100	0.4394	0.2811	0.4693	**0.1891**	0.4693	0.4685	0.1909
	200	0.4170	0.1962	0.4310	**0.1535**	0.4310	0.4307	0.1543
	500	0.3642	0.1362	0.3729	**0.1160**	0.3729	0.3728	0.1162
	1000	0.3198	***0.1041***	0.3270	**0.0901**	0.3270	0.3270	0.0903
	2000	0.2798	0.0816	0.2840	**0.0731**	0.2840	0.2840	0.0731
	5000	0.2324	0.0616	0.2332	**0.0558**	0.2332	0.2332	0.0558
	10000	0.1966	0.0485	0.1961	**0.0435**	0.1961	0.1961	0.0435
	20000	0.1660	0.0392	0.1646	**0.0351**	0.1646	0.1646	0.0351
	50000	0.1307	0.0286	***0.1282***	**0.0255**	0.1282	0.1282	0.0255
2	100	0.4703	0.1927	0.4724	0.1883	0.4724	0.3951	**0.1581**
	200	0.4460	***0.1612***	0.4456	0.1589	0.4456	0.3610	**0.1314**
	500	0.4004	0.1197	0.4014	0.1197	0.4014	0.3100	**0.0937**
	1000	0.3679	0.1017	0.3689	0.1017	0.3689	0.2742	**0.0776**
	2000	0.3360	0.0814	0.3358	0.0812	0.3358	0.2375	**0.0614**
	5000	0.2903	0.0624	0.2904	0.0623	0.2904	0.1943	**0.0456**
	10000	0.2565	0.0496	0.2566	0.0496	0.2566	0.1636	**0.0363**
	20000	0.2248	0.0410	0.2247	0.0409	0.2247	0.1378	**0.0293**
	50000	0.1854	0.0305	***0.1852***	0.0305	0.1852	0.1078	**0.0217**
3	100	0.4611	0.2949	0.5332	**0.1966**	0.5332	0.5230	0.1984
	200	0.4281	0.2252	0.4999	**0.1558**	0.4999	0.4893	0.1565
	500	0.3733	0.1753	0.4475	**0.1232**	0.4475	0.4370	0.1240
	1000	0.3346	***0.1345***	0.4079	**0.1009**	0.4079	0.3978	0.1009
	2000	0.2931	0.1086	0.3653	**0.0806**	0.3653	0.3552	0.0811
	5000	0.2434	0.0809	0.3114	**0.0603**	0.3114	0.3022	0.0604
	10000	0.2087	0.0630	0.2721	**0.0481**	0.2721	0.2639	0.0483
	20000	0.1752	0.0509	0.2337	**0.0378**	0.2337	0.2265	0.0379
	50000	***0.1392***	0.0371	0.1896	**0.0287**	0.1896	0.1838	0.0289
4	100	0.5491	0.3582	0.4963	**0.3038**	0.4963	0.4860	0.3105
	200	0.5320	0.3267	0.4758	**0.2775**	0.4758	0.4657	0.2850
	500	0.5161	***0.3008***	0.4532	**0.2494**	0.4532	0.4432	0.2582
	1000	0.4962	0.2718	0.4321	**0.2248**	0.4321	0.4232	0.2342
	2000	0.4781	0.2481	0.4125	**0.2024**	0.4125	0.4048	0.2124
	5000	0.4515	0.2161	0.3864	**0.1739**	0.3864	0.3809	0.1840
	10000	0.4301	0.1934	0.3669	**0.1542**	0.3669	0.3633	0.1641
	20000	0.4079	0.1706	0.3475	**0.1341**	0.3475	0.3457	0.1436
	50000	0.3774	0.1428	***0.3218***	**0.1103**	0.3218	0.3226	0.1191
5	100	0.4100	0.1678	0.3791	**0.1699**	0.3791	0.3786	0.1709
	200	0.3817	0.1369	0.3485	**0.1402**	0.3485	0.3483	***0.1407***
	500	0.3455	0.1081	0.3094	**0.1111**	0.3094	0.3093	0.1112
	1000	0.3138	0.0874	0.2791	**0.0908**	0.2791	0.2791	0.0909
	2000	0.2825	0.0704	0.2502	**0.0741**	0.2502	0.2502	0.0742
	5000	0.2430	0.0526	0.2163	**0.0559**	0.2163	0.2163	0.0559
	10000	0.2134	0.0421	0.1922	**0.0450**	0.1922	0.1922	0.0450
	20000	0.1854	0.0338	0.1696	**0.0365**	0.1696	0.1697	0.0365
	50000	0.1519	0.0261	***0.1424***	**0.0280**	0.1424	0.1424	0.0280

Table 3.11: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) using varying number of samples. The results show the mean IMSE (Integrated Mean Square Error) error of 100 trials. The 2-d distributions are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	**nsam**	$rot(Z_1)$	$exrot(Z_1)$	$rot(C_2)$	$exrot(C_2)$	$rot(Z_2)$	$rot(C_3)$	$exrot(C_3)$
6	100	0.8052	0.6559	0.7819	**0.6123**	0.7819	0.7818	**0.6123**
	200	0.7978	***0.6271***	0.7746	**0.5804**	0.7746	0.7745	**0.5804**
	500	0.7867	0.5827	0.7637	**0.5341**	0.7637	0.7637	**0.5341**
	1000	0.7779	0.5499	0.7552	**0.4986**	0.7552	0.7551	**0.4986**
	2000	0.7673	0.5123	0.7441	**0.4604**	0.7441	0.7441	**0.4604**
	5000	0.7504	0.4618	0.7255	**0.4082**	0.7255	0.7255	**0.4082**
	10000	0.7343	0.4234	0.7073	**0.3689**	0.7073	0.7073	**0.3689**
	20000	0.7143	0.3824	0.6851	**0.3280**	0.6851	0.6851	**0.3280**
	50000	0.6822	0.3279	***0.6509***	**0.2752**	0.6509	0.6509	**0.2752**
7	100	0.5749	0.3537	0.5535	**0.2942**	0.5535	0.5376	0.3398
	200	0.5629	***0.3197***	0.5397	**0.2596**	0.5397	0.5247	0.3050
	500	0.5433	0.2665	0.5172	**0.2097**	0.5172	0.5058	0.2557
	1000	0.5274	0.2315	0.4978	**0.1789**	0.4978	0.4920	0.2224
	2000	0.5095	0.1991	0.4751	**0.1512**	0.4751	0.4768	0.1914
	5000	0.4817	0.1610	0.4393	**0.1198**	0.4393	0.4525	0.1551
	10000	0.4561	0.1352	0.4078	**0.0990**	0.4078	0.4297	0.1303
	20000	0.4265	0.1124	0.3732	**0.0813**	0.3732	0.4028	0.1088
	50000	0.3829	0.0877	***0.3256***	**0.0625**	0.3256	0.3622	0.0849
8	100	0.5582	0.3390	0.5351	**0.2850**	0.5351	0.5096	0.3224
	200	0.5422	***0.2994***	0.5180	**0.2465**	0.5180	0.4926	0.2870
	500	0.5233	0.2593	0.4957	**0.2080**	0.4957	0.4738	0.2481
	1000	0.5068	0.2259	0.4758	**0.1769**	0.4758	0.4589	0.2165
	2000	0.4877	0.1947	0.4528	**0.1497**	0.4528	0.4428	0.1877
	5000	0.4592	0.1585	0.4180	**0.1192**	0.4180	0.4193	0.1535
	10000	0.4345	0.1342	0.3886	**0.0992**	0.3886	0.3986	0.1305
	20000	0.4063	0.1125	0.3566	**0.0820**	0.3566	0.3748	0.1097
	50000	0.3656	0.0883	***0.3127***	**0.0634**	0.3127	0.3394	0.0864
9	100	0.6318	0.4363	0.6043	0.3702	0.6043	0.5261	**0.3199**
	200	0.6189	0.4032	0.5884	0.3357	0.5884	0.5013	**0.2907**
	500	0.6006	0.3641	0.5654	0.2966	0.5654	0.4734	**0.2609**
	1000	0.5846	0.3345	0.5455	0.2688	0.5455	0.4519	**0.2380**
	2000	0.5678	***0.3059***	0.5241	0.2412	0.5241	0.4303	**0.2166**
	5000	0.5411	0.2673	0.4916	0.2060	0.4916	0.4012	**0.1881**
	10000	0.5190	0.2404	0.4654	0.1818	0.4654	0.3795	**0.1672**
	20000	0.4945	0.2142	0.4376	0.1587	0.4376	0.3570	**0.1482**
	50000	0.4593	0.1815	0.3993	0.1307	0.3993	***0.3280***	**0.1241**
10	100	0.5286	**0.2452**	0.5095	0.2627	0.5095	0.5085	0.2597
	200	0.5078	**0.2152**	0.4847	***0.2322***	0.4847	0.4840	0.2287
	500	0.4680	**0.1716**	0.4452	0.1894	0.4452	0.4449	0.1860
	1000	0.4413	**0.1472**	0.4193	0.1642	0.4193	0.4189	0.1606
	2000	0.4099	**0.1206**	0.3920	0.1371	0.3920	0.3913	0.1339
	5000	0.3680	**0.0937**	0.3575	0.1083	0.3575	0.3561	0.1057
	10000	0.3368	**0.0768**	0.3316	0.0896	0.3316	0.3295	0.0874
	20000	0.3047	**0.0632**	0.3048	0.0744	0.3048	0.3019	0.0725
	50000	***0.2621***	**0.0479**	0.2680	0.0570	0.2680	0.2644	0.0555

Table 3.12: Performance comparison of the bandwidth selection methods for Kernel Density Estimation (KDE) using varying number of samples. The results show the mean IMSE (Integrated Mean Square Error) error of 100 trials. The 2-d distributions are: 1. Uncorrealted Normal 2. Correlated Normal 3. Skewed 4. Kurtotic 5. Bimodal I 6. Bimodal II 7. Bimodal III 8. Bimodal IV 9. Trimodal I 10. Trimodal II 11. Trimodal III 12. Quadrimodal

PDF_i	**nsam**	$rot(Z_1)$	$exrot(Z_1)$	$rot(C_2)$	$exrot(C_2)$	$rot(Z_2)$	$rot(C_3)$	$exrot(C_3)$
11	100	0.4509	0.2337	0.4319	0.2118	0.4319	0.4203	**0.1963**
	200	0.4329	***0.2091***	0.4128	0.1869	0.4128	0.4003	**0.1710**
	500	0.4057	0.1733	0.3852	0.1529	0.3852	0.3712	**0.1390**
	1000	0.3844	0.1477	0.3635	0.1289	0.3635	0.3478	**0.1155**
	2000	0.3623	0.1227	0.3412	0.1055	0.3412	0.3247	**0.0933**
	5000	0.3336	0.0989	0.3112	0.0843	0.3112	0.2935	**0.0741**
	10000	0.3098	0.0818	0.2866	0.0692	0.2866	0.2687	**0.0604**
	20000	0.2855	0.0680	0.2616	0.0571	0.2616	0.2436	**0.0497**
	50000	0.2511	0.0519	***0.2271***	0.0433	0.2271	0.2095	**0.0373**
12	100	0.4209	0.2425	0.3927	0.2081	0.3927	0.3803	**0.2067**
	200	0.4015	***0.2162***	0.3734	0.1822	0.3734	0.3602	**0.1806**
	500	0.3787	0.1873	0.3504	0.1533	0.3504	0.3381	**0.1513**
	1000	0.3609	0.1637	0.3326	0.1305	0.3326	0.3213	**0.1283**
	2000	0.3433	0.1424	0.3148	0.1112	0.3148	0.3053	**0.1089**
	5000	0.3194	0.1156	0.2898	0.0881	0.2898	0.2828	**0.0857**
	10000	0.3012	0.0982	0.2702	0.0736	0.2702	0.2646	**0.0713**
	20000	0.2818	0.0814	0.2492	0.0599	0.2492	0.2447	**0.0579**
	50000	0.2550	0.0638	***0.2206***	0.0462	0.2206	0.2167	**0.0446**
mean		0.4207	0.1905	0.4023	**0.1628**	0.4023	0.3844	0.1644
median		0.4100	0.1611	0.3875	**0.1324**	0.3875	0.3738	0.1365

Chapter 4

Near Independence and BSS

This chapter aims BSS of near-independent sources for linear, instantaneous mixtures. To achieve that, it does theoretical and empirical study on the optimization landscape due to various contrasts against varying source distributions. It derives a Search for Rotation based ICA (SRICA) algorithm using search based global optimization technique. Using SRICA, it verifies the previously derived contrasts and compares separability of various contrasts. Based on the empirical results, it provides discussion on the use of ICA for BSS in linear, instantaneous mixtures.

4.1 Introduction

The ICA model assumes the sources being separated, as mutually the *m.i.p.* with respect to a given contrast function. The *m.i.p.* sources assure global optimal in the optimization landscape due to used contrast. As already discussed and defined in Chapter 1.5; the sources producing either shift of global optima or addition of spurious local optima or both with respect to the used contrast qualify to be near-independent for that contrast. It is to be noted that the near-independence is not a characteristic of sources alone, but it is the characteristic of sources exhibited in the presence of a specific contrast. The BSS of such near-independent sources with given contrast function is defined as 'near-independent BSS' (nIBSS). The nIBSS study should focus on the optimization landscape violations due to either source distribution and contrast duos or other reasons and their consequences on separation quality. The nIBSS study do not denounce use of approximations and approximate solutions. But, more precise solution is an everlasting hunger. Overall, there are following motivations to study the nIBSS.

1. At lower dimensions, a slight shift in global optima may allow atleast an approximate solution. With increasing dimension, cumulative slight shifts in pairwise optima, may cause the actual solution much far than the global optimal. So, nIBSS study is needed when large

scale.

2. There are two already known phenomena that affect the optimization landscape of ICA contrasts. There exists spurious local minima for entropy based contrast (17, 90, 91, 92, 132, 134). There happens to be a shift of global for kurtosis based contrasts with lack of number of samples (83, 108, 109). There is needed a study to find other phenomena or circumstances that bring optimization landscape violations. Also, this motivates to search for a contrast that do not have local minima, atleast for multimodal distributions.

3. There is atleast one more reason that produces a shift of global optima. The existing ICA algorithms use contrasts derived through varying independence definitions, their interpretations and approximations. The various interpretations of independence definition include either non-Gaussianity based or entropy based or pairwise independence based or nonlinear decorrelation based interpretations. The approximations are derived in terms of higher order cumulants or moments using truncated version of either a Gram-Charlier expansion or an Edgeworth expansion of PDF or entropy. The various contrasts may agree for the exact independence condition, but it is highly unlikely that they will match for the 'degree of independence'. So, *m.i.p.* with respect to one of the contrsts do not assure *m.i.p.* with respect to the others. In practice, before an actual BSS application through ICA, it is rarely assured either the sources are *m.i.p.* or there are not other possible signals separated from the same mixture that are more independent than the actual sources with respect to the used contrast. The scenario is - the ICA mathematics, many times, uses properties available through ideal independence; the ICA definition allows the components to be *m.i.p.* and the real world ICA applications deal with the sources that are neither independent nor *m.i.p.*. For example, as proved in (33), pairwise independence among random variables is equivalent to independence of the random vector. But, isn't it worth to question whether pairwise *m.i.p.* random variables imply *m.i.p.* random vector? Overall, whether solution within acceptable range or not, there must be atleast a study on the separation quality and possible remedy due to various approximations together.

4. Independent Component Analysis (ICA) is an established tool for both Component Analysis (CA) (16, 42) and Blind source Separation (BSS) (25, 35). As a tool for CA, ICA claims to express the real nature of data by finding components that are *m.i.p.* with respect to the used contrast. If the contrast changes, the ICs get changed for the same data. For CA applications, the solution is still useful with some change in the amount of redundancy removal. Compare to that, the goal for BSS is to get back the actual sources irrespective of the used contrast. So, with change in estimated ICs, the estimated sources get affected. Historically, ICA has been introduced as a way for BSS in linear instantaneous mixing. So, conventionally, in linear

> instantaneous mixing system ICA is considered equivalent to BSS though both CA and BSS have different goals. But, whether this equivalence should be considered even in large scale also? To answer this, it would be interesting to study on how the estimated sources are affected due to change in the contrast in lower dimensions as well in higher dimensions.

The require study needs comparision of separated sources through various contrasts against varying distributions, against varying number of samples and against varying number of sources. The sources separated depend both upon the used contrast and the used optimization technique. So, to have comparision of separated sources due to the contrast only, there is needed an ICA algorithm that permits use of various contrasts with a same optimization technique.

There are existing many algebraic (21, 22, 26, 28, 33) and neural net techniques (11, 18, 63, 64, 72, 77) for linear ICA. The algebraic techniques are based on the uncorrelatedness of higher order statistics, similar to the algebraic techniques for PCA based on the uncorrelatedness of second order statistics. With approximated contrasts, approximate solutions are expected. In case of a neural net techniques, the nonlinearity used for learning has to be a function of PDF of the components to be estimated. In the absence of this knowledge, family of densities e.g. superGaussian or subGaussian; is used as an approximation to select the nonlinearity. This requires to have some prior knowledge of densities to be estimated and so violates the blind assumption. Also, gradient based optimization methods have poor global convergence. Overall, an ICA algorithm - allowing use of varying independence measures, assuring global solution and being truly blind - is required and can not be obtained through gradient based optimization technique. This leads to have an ICA algorithm based on global search techniques.

With above motivations, the chapter contents are divided into three parts. The next Section 4.2 briefs conventional BSS contrasts. The Section 4.3 defines the concept of near-independence and does empirical study of the optimization landscape due to various ICA contrasts.The next Section 4.4 studies the theoretical extrema analysis for Shannon entropy and kurtosis based contrasts. Then, the Section **??** concludes the extrema analysis. The Section 4.7 derives the Search for Rotation based ICA (SRICA) algorithm based on the global search based optimization techniques as a solution to the needed ICA algorithm. The contrasts derived in previous chapters are verified using the SRICA in Section 4.8. The Section **??** provides comparision of various contrasts based on the separated sources and finally the Section **??** provides discussion based on the empirical results. Finally, the Section 4.9 concludes the chapter.

4.2 The Conventional Optimization Criteria or Contrasts for ICA

Let $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ be a random vector. Then, the variable $y_i, i = 1 : n$ are independent if and only if $p_{\mathbf{y}}(y_1, y_2, \ldots, y_n) = \prod_{i=1}^{n} p_{y_i}(y_i)$ (36, 84), where $p(\mathbf{y})$ is the joint distribution and $\prod_{i=1}^{n} p(y_i)$ is the product of marginal distributions. Accordingly, the contrasts based on an asymmetric distance measure Kullback-Leibler Divergence (KLD) between two distributions, $p(\mathbf{y})$ and $\prod_{i=1}^{n} p(y_i)$, and the mutual information (25, 33) are most widely used.

$$\Phi^{kld}(\mathbf{y}) := -KLD\left(p(\mathbf{y}), \prod_{i=1}^{n} p(y_i)\right) = -\int_{\mathbf{y}} p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod_{i=1}^{n} p(y_i)} d\mathbf{y} \tag{4.1}$$

where, $d\mathbf{y} = dy_1 dy_2 ... dy_n$
The equation 4.1 also interprets KLD, as a measure of mutual information $I(\mathbf{y})$ among the random variables y_is.

$$\begin{aligned}
\Phi^{mikld}(\mathbf{y}) &:= -I(y_1, y_2, ..., y_n) = -KLD\left(p(\mathbf{y}), \prod_{i=1}^{n} p(y_i)\right) && (4.2)\\
&= -\left(\sum_{i=1}^{n} H(y_i) - H(\mathbf{y})\right) && (4.3)\\
&= -\left(\sum_{i=1}^{n} H(y_i) - H(\mathbf{z}) - \log(|R|)\right) \quad (\because \mathbf{y} = \mathbf{Rz}) && (4.4)\\
\Phi^{hyi}(\mathbf{y}) &:= -\sum_{i=1}^{n} H(y_i) && (4.5)
\end{aligned}$$

where, $H(\cdot)$ denotes Shannon's Entropy (36). The $\Phi^{kld}(\mathbf{y})$ and $\Phi^{mi}(\mathbf{y})$ are zero when both the distributions are same or the y_is are mutually independent. The other independence measures used for the experiments in the thesis are defined as under.

$$\begin{aligned}
\Phi^{jskld}(\mathbf{y}) &:= -\left(KLD(\mathbf{y}, \mathbf{M}) + KLD(y_i, \mathbf{M})\right) && (4.6)\\
\Phi^{k4}(\mathbf{y}) &:= \sum_{i=1}^{n} |kurt(y_i) - 3| && (4.7)
\end{aligned}$$

where, $M = \frac{1}{2}(p(\mathbf{y}) + \prod_{i=1}^{n} p(y_i))$; $kurt(\cdot)$ denotes kurtosis and $|\cdot|$ denotes the absolute value.

4.3 Empirical Extrema Analysis of Conventional Contrasts

The goal, in this section, is to empirically verify the existence of spurious optima of various independence measures for varying distributions. This type of results have been obtained for entropy measure applied to multimodal distributions, in the articles (17, 90, 91, 92, 132, 134). The current section, achieves this for 21 distributions (shown in fig. 4.1) and for three independence measures Φ^{hyi}, Φ^{k4} and Φ^{radH}; where Φ^{radH} is the contrast derived by (75) that is based on sum of marginal entropies and spacing estimates of entropy. The first 18 types (a to r) of distributions are suggested by (10) and two more skewed types of distributions were added to test the performance of the ICA algorithms against skewed sources. The s type is a GGD with skewness $\mu_3 = -0.25$ (left skewed) and kurtosis $k_4 = 3.75$ and the t type is a GGD with skewness $\mu_3 = 0.75$ (right skewed) and kurtosis $k_4 = 0$. Both the distributions are generated using Power Method with parameters $b = 0.75031534111078$, $c = -0.02734119591845$, $d = 0.07699282409939$ for s type and $b = 1.11251460048528$, $c = 0.17363001955694$ and $d = -.05033444870926$ for t type. The u type is a Rayleigh distribution with $\beta = 1$ and so the corresponding $\mu_3 = 0.631$ and $k_4 = 0.245$. All 21 distributions are shown in the Figure 4.1.

Assuming the sources s_is to be zero-mean and univariant, $\mathbf{y} = \mathbf{G}\mathbf{s}$ from the previous equation The variance of y_is is restricted to be 1, as equal to that of s_is. So, $g_{i1}^2 + g_{i2}^2 = 1$. The matrix G can be represented as

$$G = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

Accordingly, by varying θ for $\mathbf{G}(\theta)$, all possible estimated sources can be generated.

There have been obtained plots of independence measures versus angle θ for all the distributions. The optima for maximum independence should be obtained at $\theta = 0$, as corresponding to this point estimated sources y_is are equal to the actual sources s_is. The results were interesting for the multi-modal densities and the skewed densities. Accordingly, figure 4.2 shows the plots of independence measure versus θ for symmetric, multi-modal densities from f to i. Similarly, figure 4.3 shows the plots of independence measure versus θ for skewed multi-modal density j; skewed uni-modal densities s, t and a Gaussian density u. In both the figures - the plots in the first column are for $\Phi^{hyi}(\mathbf{y})$; the plots in the second column are for $\Phi^{k4}(\mathbf{y})$ and the plots in the third column are for $\Phi^{radH}(\mathbf{y})$. The following observations can be made.

- The density u (Gaussian) has both multiple minima and shift of the global. That is as expected. So, discussion here is for other densities.

- The densities - g, i, j, n and t - for which dissatisfactory performances were obtained, as in table **??**, show existence of spurious optima except density n. The $\Phi^{hyi}(\mathbf{y})$ measure shows

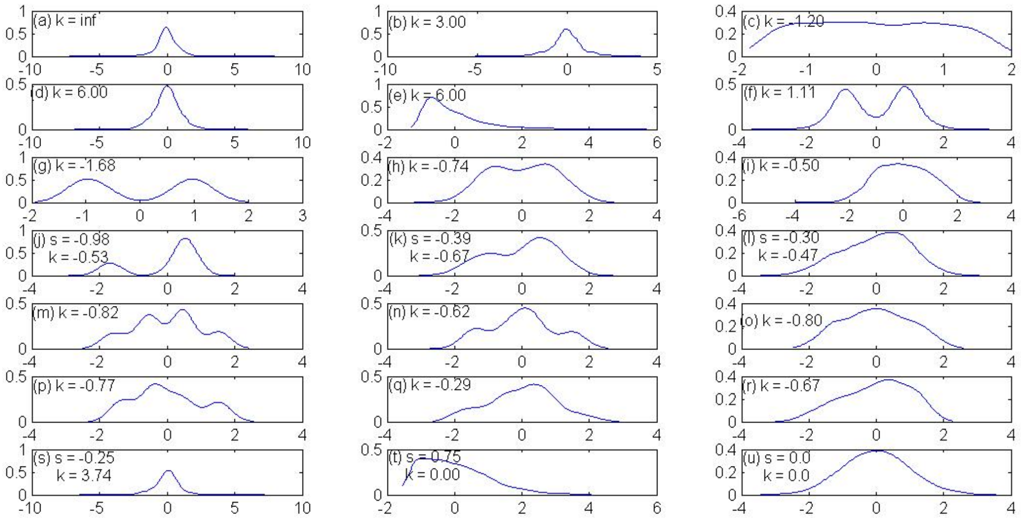

Figure 4.1: Probability density functions of sources with their kurtosis: (a) Student with 3 degrees of freedom; (b) double exponential; (c) uniform; (d) Student with 5 degrees of freedom; (e) exponential; (f) mixture of two double exponentials; (g)-(h)-(i) symmetric mixtures of two Gaussians: multimodal, transitional and unimodal; (j)-(k)-(l) nonsymmetric mixtures of two Gaussians, multimodal, transitional and unimodal; (m)-(n)-(o) symmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (p)-(q)-(r) nonsymmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (s) left skewed Generalized Gaussian Distribution(GGD); (t) right skewed GGD; (u) Gaussian distribution

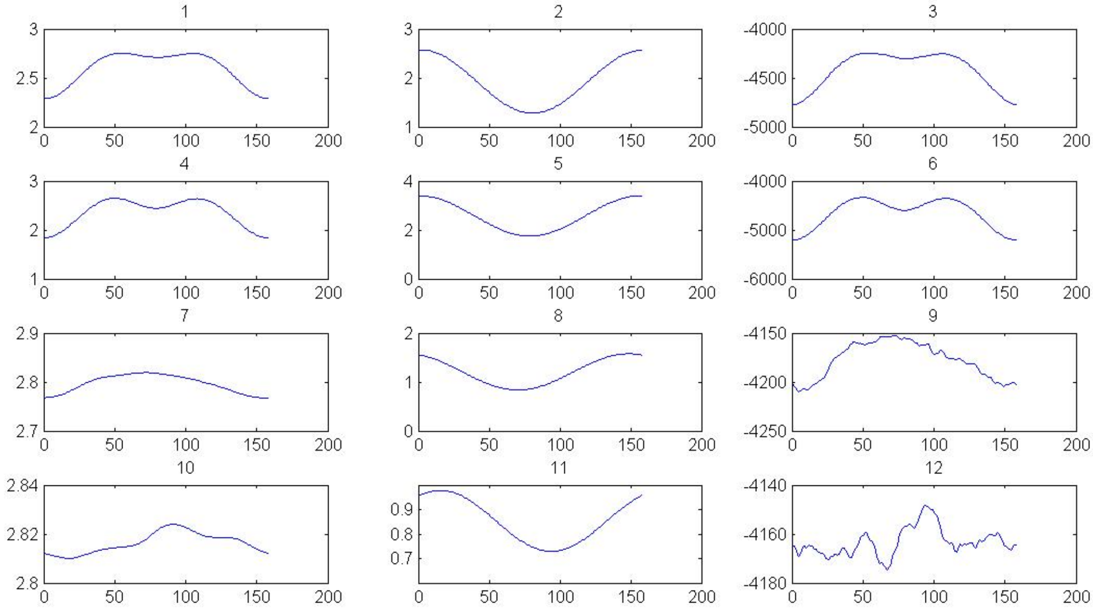

Figure 4.2: The plots of independence measures versus theta for some symmetric, multi-modal distributions:- plots (1)-(4)-(7)-(10): $\Phi^{hyi}(\mathbf{y})$ for distributions f, g, h, i; plots (2)-(5)-(8)-(11): $\Phi^{k4}(\mathbf{y})$ for distributions f, g, h, i; plots (3)-(6)-(9)-(12): $\Phi^{radH}(\mathbf{y})$ for distributions f, g, h, i

spurious minima for densities f, g, i and j. The density i has also slight shift of the global optima. The $\Phi^{radH}(\mathbf{y})$ measure shows many local minima and requires smoothening, as discussed in the article (75). But, obvious spurious minima, which can not be avoided through smoothening, are obtained for densities f, g, i and j. Out of them, the densities i show shift in global also. The observations justify the values obtained for API and the independence measure, in the previous tables.

- The $\Phi^{k4}(\mathbf{y})$ measure does not show existence of spurious maxima for any densities. Instead, it shows shift of the global maxima for densities i and t.

- It should be noted that an added local optima makes the optimization landscape more difficult to optimize but the shifted global optima makes it almost impossible to find the actual solution without any additional information.

- The observations need further mathematical analysis to find the reasons for the observations.

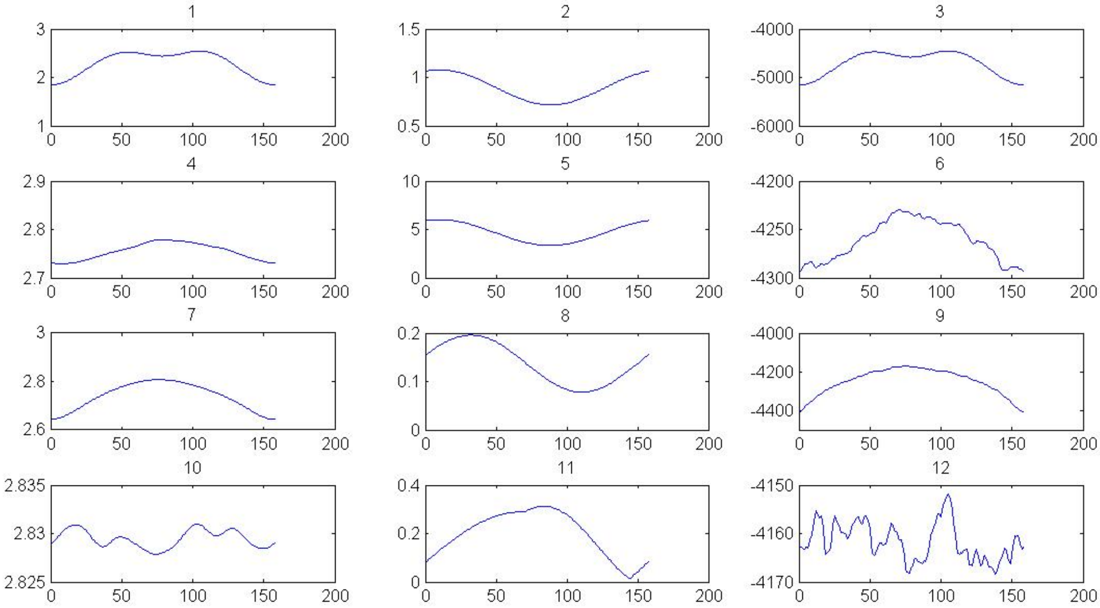

Figure 4.3: The plots of independence measures versus theta for skewed distributions and Normal distribution:- plots (1)-(4)-(7)-(10): $\Phi^{hyi}(\mathbf{y})$ for distributions j, s, t, u; plots (2)-(5)-(8)-(11): $\Phi^{k4}(\mathbf{y})$ for distributions j, s, t, u; plots (3)-(6)-(9)-(12): $\Phi^{radH}(\mathbf{y})$ for distributions j, s, t, u

4.4 Theoretical Extrema Analysis of Conventional BSS Contrasts against Varying Distributions

The extrema analysis of an entropy measure for symmetric and near Gaussian sources have been done in the articles (17, 132, 134). Here, similar analysis of both entropy based and kurtosis based independence measures have been done. The sources are assumed to be independent, near Gaussian, symmetric or non-symmetric and identical or un-identical

The ICA problem, to estimate the orthogonal matrix $\mathbf{R}$, can be expressed as,

$$arg \min_{\mathbf{R}} \Phi(\mathbf{R}) \; s.t. \; \mathbf{R}\mathbf{R}^{\mathbf{T}} = 1 \tag{4.8}$$

We are interested to verify whether a spurious optima of the cost function $\Phi(\mathbf{R})$ exists or not. At the stationary points, including a spurious optima, the gradient must be zero. Taking gradient of the cost function $\Phi(\mathbf{R})$ on the Stiefel manifold (∇_m),

$$\nabla_m \Phi(\mathbf{R}) = \nabla\Phi(\mathbf{R}) - \mathbf{R}\nabla\Phi(\mathbf{R})^T\mathbf{R} \tag{4.9}$$

where, $\nabla\Phi(\mathbf{R})$ is the gradient of the cost function on the Euclidean space.

$$\nabla_m\Phi(\mathbf{R}) = 0 \Rightarrow \nabla\Phi(\mathbf{R})\mathbf{R}^T = \mathbf{R}\nabla\Phi(\mathbf{R})^T \tag{4.10}$$

4.4.1 Extrema Analysis of Entropy based Contrasts

We are interested in finding the independence measure for the possible estimated sources. Without loss of generality, let, the two sources $\mathrm{s} = (s_1, s_2)^T$ are zero-mean, uni-variant and the mixing matrix $\mathbf{A}$ is identity. Then, the estimated sources y_is can be represented, using $\mathbf{y} = \mathbf{RAs}$, as

$$\begin{aligned} y_1 &= r_{11}s_1 + r_{12}s_2 \\ y_2 &= -r_{21}s_1 + r_{22}s_2 \end{aligned}$$

The pdf of any near Gaussian signals can be expressed using Gram-Charlier series with two correction terms to a Gaussian pdf. For example, pdf of y_i is represented as under,

$$f_{y_i}(u) = g(u)\left(1 + \frac{k_{3,y_i}}{6}H_3(u) + \frac{k_{4,y_i}}{24}H_4(u)\right)\ i = 1,2 \tag{4.11}$$

where, $g(u)$ is a zero-mean, uni-variant Gaussian pdf; $H_3(u)$ and $H_4(u)$ are, consecutively, the third and fourth order Chebyshev-Hermite polynomials; k_{3,y_i} and k_{4,y_i} are the third and fourth order cumulants for signal y_i. The cumulants can be calculated as under.

$$\begin{aligned} \mu_{3,y_i} &= E\{y_i^3\} = E\{(r_{i1}s_1 + r_{i2}s_2)^3\} \\ &= r_{i1}^3\mu_{3,s_1} + r_{i2}^3\mu_{3,s_2} \\ &\quad (\because \text{ zero mean, uncorrelated sources }) \\ k_{3,y_i} &= \mu_{3,y_i} \end{aligned} \tag{4.12}$$

$$\begin{aligned} \mu_{4,y_i} &= E\{y_i^4\} = E\{(r_{i1}s_1 + r_{i2}s_2)^4\} \\ &= r_{i1}^4\mu_{3,s_1} + 6r_{i1}^2r_{i2}^2 + r_{i2}^4\mu_{3,s_2} \\ &\quad (\because \text{ zero mean, univariant, uncorrelated sources}) \\ k_{4,y_i} &= \mu_{4,y_i} - 3 \end{aligned} \tag{4.13}$$

The learning rule in equation (4.10) can be rewritten for the Φ^{hyi} measure as,

$$\begin{aligned} \nabla h(\mathbf{r_1})\mathbf{r_2}^T &= \nabla h(\mathbf{r_2})\mathbf{r_1}^T \\ \Rightarrow r_{21}\frac{\partial h(\mathbf{r_1})}{\partial r_{11}} + r_{22}\frac{\partial h(\mathbf{r_1})}{\partial r_{12}} &= r_{11}\frac{\partial h(\mathbf{r_2})}{\partial r_{21}} + r_{12}\frac{\partial h(\mathbf{r_2})}{\partial r_{22}} \end{aligned} \tag{4.14}$$

The corresponding derivatives with $h(\mathbf{y_i}) = \int_{-\infty}^{\infty} f_{(y_i)} \log f_{(y_i)} dy_i$ can be expressed as,

$$\begin{aligned} \frac{\partial h(\mathbf{r_i})}{\partial r_{ij}} &= -\int_{-\infty}^{\infty} (1 + \log f_{y_i}(u)) \frac{\partial f_{y_i}}{\partial r_{ij}} du \\ \frac{\partial f_{y_i}(u)}{\partial r_{ij}} &= g(u) \left[\frac{1}{2} \left(r_{ij}^2 \mu_{3,s_i} \right) H_3(u) \right. \\ &\quad \left. + \frac{1}{6} \left(r_{ij}^3 \mu_{4,s_i} + 3 r_{ij} r_{i1} r_{i2} \right) H_4(u) \right] \end{aligned} \tag{4.15}$$

Rewriting equation (4.14),

$$\begin{aligned} &\int_{\infty}^{\infty} (1 + \log f_{y_1}(u)) \left[r_{21}\frac{\partial f_{y_1}(u)}{\partial r_{11}} + r_{22}\frac{\partial f_{y_1}(u)}{\partial r_{12}} \right] du \\ &\quad = \int_{\infty}^{\infty} (1 + \log f_{y_2}(u)) \left[r_{11}\frac{\partial f_{y_2}(u)}{\partial r_{21}} + r_{12}\frac{\partial f_{y_2}(u)}{\partial r_{22}} \right] du \\ &\int_{-\infty}^{\infty} g(u) \left(1 + \log f_{y_1}(u)\right) D_1(u, r) du \\ &\quad = \int_{-\infty}^{\infty} g(u) \left(1 + \log f_{y_2}(u)\right) D_2(u, r) du \end{aligned} \tag{4.16}$$

where,

$$\begin{aligned} D_1(u, r) &= \frac{1}{g(u)} \left[r_{21}\frac{\partial f_{y_1}(u)}{\partial r_{11}} + r_{22}\frac{\partial f_{y_1}(u)}{\partial r_{12}} \right] \\ &= c_{3,y_1} H_3(u) + c_{4,y_1} H_4(u) && (4.17) \\ c_{3,y_1} &= \frac{1}{2} \left(r_{11}^2 r_{21} \mu_{3,s_1} + r_{12}^2 r_{22} \mu_{3,s_2} \right) && (4.18) \\ c_{4,y_1} &= \frac{1}{6} \left(r_{11}^3 r_{21} \mu_{4,s_1} + r_{12}^3 r_{22} \mu_{4,s_2} \right) \\ &\quad + \frac{1}{2} \left(r_{11} r_{12}^2 r_{21} + r_{11}^2 r_{12} r_{22} \right) && (4.19) \end{aligned}$$

Similarly,

$$\begin{aligned}
D_2(u,r) &= \frac{1}{g(u)}\left[r_{11}\frac{\partial f_{y_2}(u)}{\partial r_{21}} + r_{12}\frac{\partial f_{y_2}(u)}{\partial r_{22}}\right] \\
&= c_{3,y_2}H_3(u) + c_{4,y_2}H_4(u) && (4.20) \\
c_{3,y_2} &= \frac{1}{2}\left(r_{11}r_{21}^2\mu_{3,s_1} + r_{12}r_{22}^2\mu_{3,s_2}\right) && (4.21) \\
c_{4,y_2} &= \frac{1}{6}\left(r_{11}r_{21}^3\mu_{4,s_1} + r_{12}r_{22}^3\mu_{4,s_2}\right) \\
&+ \frac{1}{2}\left(r_{11}r_{21}r_{22}^2 + r_{12}r_{21}^2r_{22}\right) && (4.22)
\end{aligned}$$

Using this for equation (4.16)

$$\begin{aligned}
&\int_{\infty}^{\infty}\left(1+\log f_{y_1}(u)\right)\left(c_{3,y_1}H_3(u) + c_{4,y_1}H_4(u)\right)g(u)du \\
&= \int_{\infty}^{\infty}\left(1+\log f_{y_2}(u)\right)\left(c_{3,y_2}H_3(u) + c_{4,y_2}H_4(u)\right)g(u)du && (4.23)
\end{aligned}$$

Now, expanding f_{y_i} as in equation (4.11) and using Taylor series expansion $\log(1+\epsilon) = \epsilon - \frac{\epsilon^2}{2} + \ldots$

$$\begin{aligned}
\log f_{y_i}(u) &= -\frac{1}{2}\log 2\pi - \frac{u^2}{2}\log e \\
&+ \log\left(1 + \frac{k_{3,s_i}}{6}H_3(u) + \frac{k_{4,s_i}}{24}H_4(u)\right) && (4.24) \\
&= -\frac{1}{2}\log 2\pi - \frac{u^2}{2} + \left(\frac{k_{3,s_i}}{6}H_3(u) + \frac{k_{4,s_i}}{24}H_4(u)\right) \\
&- \frac{1}{2}\left(\frac{k_{3,s_i}}{6}H_3(u) + \frac{k_{4,s_i}}{24}H_4(u)\right)^2 && (4.25)
\end{aligned}$$

Also,

$$
\begin{aligned}
\int_{-\infty}^{\infty} g(u)H_3(u)du &= 0 \\
\int_{-\infty}^{\infty} u^2 g(u)H_3(u)du &= 0 \\
\int_{-\infty}^{\infty} g(u)H_4(u)du &= 0 \\
\int_{-\infty}^{\infty} u^2 g(u)H_3(u)du &= 0 \\
\int_{-\infty}^{\infty} g(u)H_m(u)H_n(u)du &= 0; m \neq n \\
&= n!; m = n \\
\int g(u)D_i(u)du &= -\left(c_{3,y_i}H_2(u) + c_{4,y_i}H_3(u)\right)
\end{aligned}
$$

The left side of the equation (4.23) can be re-written using expansion for pdf in equation (4.25),

$$
\begin{aligned}
\int_{-\infty}^{\infty} & \log\left(1 + X_i(u)\right)\left(c_{3,y_i}H_3(u) + c_{4,y_i}H_4(u)\right)g(u)du \\
=& \int_{-\infty}^{\infty}\left(\frac{k_{3,s_i}}{6}c_{3,y_i}H_3(u)^2 + \frac{k_{4,s_i}}{24}c_{4,y_i}H_4(u)^2\right. \\
& - \frac{1}{2}\frac{k_{3,s_i}^2}{36}\frac{k_{4,s_i}}{24}H_3(u)^2H_4(u) - \frac{1}{2}\left(\frac{k_{4,s_i}}{24}\right)^3 H_4(u)^3 \\
& \left. - \frac{1}{2}2\frac{k_{3,s_i}^2}{36}\frac{k_{4,s_i}}{24}H_3(u)^2H_4(u)\right)du \\
=& \frac{k_{3,s_i}}{6}c_{3,y_i}(6) + \frac{k_{4,s_i}}{24}c_{4,y_i}(24) \\
& - \frac{1}{2}\left(\frac{k_{3,s_i}^2}{36}\frac{k_{4,s_i}}{24}(3)(216) + \left(\frac{k_{4,s_i}}{6}\right)^3(1728)\right) \\
=& k_{3,s_i}c_{3,y_i} + k_{4,s_i}c_{4,y_i} \\
& - \frac{1}{16}k_{4,s_i}\left(6k_{3,s_i}^2 - k_{4,s_i}^2\right) \qquad (4.26)
\end{aligned}
$$

Accordingly, the equation (4.24) can be write as,

$$
\begin{aligned}
k_{3,s_1}c_{3,y_1} + k_{4,s_1}c_{4,y_1} - \frac{1}{16}k_{4,s_1}\left(6k_{3,s_1}^2 - k_{4,s_1}^2\right) \\
= k_{3,s_2}c_{3,y_2} + k_{4,s_2}c_{4,y_2} - \frac{1}{16}k_{4,s_2}\left(6k_{3,s_2}^2 - k_{4,s_2}^2\right) \qquad (4.27)
\end{aligned}
$$

Case-I:- Near Gaussian, *i.i.d.* sources

The sources are *i.i.d.* implies $\mu_{3,s_1} = \mu_{3,s_2}$ and $\mu_{4,s_1} = \mu_{4,s_2}$.

Case-I(a) Let the sources are symmetrical also. Accordingly, $k_{3,s_1} = k_{3,s_2} = 0$, $k_{4,s_1} = k_{4,s_2} = k_4 \neq 0$ The equation (4.27) reduces to,

$$c_{4,y_1} = c_{4,y_2} \tag{4.28}$$

$$\begin{aligned}
\Rightarrow & \frac{1}{6}\left(-\cos^3\theta\sin\theta\mu_{4,s_1} + \sin^3\theta\cos\theta\mu_{4,s_2}\right) \\
& + \frac{1}{2}\left(-\cos\theta\sin^3\theta + \sin\theta\cos^3\theta\right) \\
& - \frac{1}{6}\left(-\cos\theta\sin^3\theta\mu_{4,s_1} + \sin\theta\cos^3\theta\mu_{4,s_2}\right) \\
& - \frac{1}{2}\left(-\cos^3\theta\sin\theta + \sin^3\theta\cos\theta\right) = 0 \\
\Rightarrow & -\frac{1}{6}\sin\theta\cos\theta\cos 2\theta\mu_{4,s_1} - \frac{1}{6}\sin\theta\cos\theta\cos 2\theta\mu_{4,s_2} \\
& + \sin\theta\cos\theta\cos 2\theta = 0
\end{aligned}$$

$$\Rightarrow \sin\theta\cos\theta\cos 2\theta(\mu_4 - 3) = 0 \ (\because \mu_{4,s_1} = \mu_{4,s_2} = \mu_4) \tag{4.29}$$

The ICA allows permutation and reflection of the solution. So, $\theta \in [0, \frac{\Pi}{2})$. The equation (4.29) proves that $\theta = 0, \frac{\pi}{2}, \frac{\pi}{4}$ and $\mu_4 = 3$ are the stationary points. Among them, $\theta = 0, pi/2$ are the required minima and $\theta = pi/4$ corresponds to the global maxima. There do not exist a spurious local minima of an entropy measure for symmetric, near Gaussian and *i.i.d.* sources. This also justifies that the when the kurtosis is near zero, almost for all θ the gradient is near zero. This brings a possibility of a spurious local or global minima due to randomization. This was empirically observed in the plot for density i, in figure 4.2.

Case-I(b) Let the sources are asymmetric, with zero kurtosis and *i.i.d.*. This case corresponds to the skewed Normal distribution. Accordingly, $k_{3,s_1} = k_{3,s_2} = k_3 \neq 0$, $k_{4,s_1} = k_{4,s_2} = 0$ The equation (4.27) reduces to,

$$c_{3,y_1} = c_{3,y_2} \tag{4.30}$$

$$\begin{aligned}
\Rightarrow & \frac{1}{2}\left(-\cos^2\theta\sin\theta\mu_{3,s_1} + \sin^2\theta\cos\theta\mu_{3,s_2}\right) \\
& - \frac{1}{2}\left(\cos\theta\sin^2\theta\mu_{3,s_1} + \sin\theta\cos^2\theta\mu_{3,s_2}\right) = 0
\end{aligned}$$

$$\Rightarrow 2\sin\theta\cos\theta\cos\theta\mu_3 = 0 \ (\because \mu_{3,s_1} = \mu_{3,s_2} = \mu_3) \tag{4.31}$$

The equation 4.31 proves that $\theta = 0$, $\theta = \frac{\pi}{2}$ and $\mu_3 = 0$ are the stationary points. The conditions $\mu_3 = 0$ and $k_4 = 0$ implies Gaussianity. In that case, for all θ the gradient is zero. This justifies that the ICA solution can not be obtained for mixture of more than one Gaussians. But, in case of $\mu_3 \neq 0$, the gradient follows the shape of the $\sin\theta\cos\theta\cos\theta$. So, there do not exist a spurious local minima of an entropy measure for asymmetric, zero-kurtosis and *i.i.d.* sources. The empirical results for t type distributions justify this.

Case-I(c) Let the sources are asymmetric and *i.i.d.*. Accordingly, $k_{3,s_1} = k_{3,s_2} = k_3 \neq 0$, $k_{4,s_1} = k_{4,s_2} = k_4 \neq 0$. The equation (4.27) reduces to,

$$\begin{aligned}
& k_{3,s_1}c_{3,y_1} + k_{4,s_1}c_{4,y_1} = k_{3,s_2}c_{3,y_2} + k_{4,s_2}c_{4,y_2} && (4.32)\\
\Rightarrow & k_3(c_{3,y_1} - c_{3,y_2}) + k_4(c_{4,y_1} - c_{4,y_2}) = 0 \\
\Rightarrow & \text{Either } (c_{3,y_1} - c_{3,y_2} = 0 \text{ and } c_{4,y_1} - c_{4,y_2} = 0) \\
& \text{or } \left(\frac{c_{3,y_1} - c_{3,y_2}}{c_{4,y_1} - c_{4,y_2}} = -\frac{k_4}{k_3}\right) && (4.33)
\end{aligned}$$

This proves that other than $\theta = 0, \frac{\pi}{2}$, there is only one other stationary point satisfying $\frac{\cos 2\theta}{\cos\theta} = -\frac{\mu_3}{\mu_4 - 3} = -\frac{k_3}{k_4}$. This corresponds to a shifted maxima. There is no spurious minima of entropy measure for non-symmetric, near Gaussian *i.i.d.* sources.

Case-II:- Near Gaussian, un-identical but independent sources

The un-identical sources imply, $\mu_{3,s_1} \neq \mu_{3,s_2}$ and $\mu_{4,s_1} \neq \mu_{4,s_2}$. Let $\mu_{3,s_1} - \mu_{3,s_2} = a_3$; $\mu_{3,s_1} + \mu_{3,s_2} = b_3$; $\mu_{4,s_1} - \mu_{4,s_2} = a_4$ and $\mu_{4,s_1} - \mu_{4,s_2} = b_4$.

Case-II(a) Let the sources are symmetrical also. Accordingly, $k_{3,s_1} = k_{3,s_2} = a_3 = b_3 = 0$. The equation (4.27) reduces to,

$$\begin{aligned}
& k_{4,s_1}c_{4,y_1} + \frac{1}{16}k^3_{4,s_1} = k_{4,s_2}c_{4,y_2} + \frac{1}{16}k^3_{4,s_2} && (4.34)\\
\Rightarrow & \left(rc_{4,y_1} - c_{4,y_2}\right) + \frac{1}{16}\left(r{k_{4,s_1}}^2 - k^2_{4,s_2}\right) he = 0 \\
& \because \text{taking } r = \tfrac{k_{4,s_1}}{k_{4,s_2}} \\
\Rightarrow & c_{4,y_1} - c_{4,y_2} = -\frac{1}{16}ab + (1-r)c_{4,y_1} - (1-r)\frac{k^2_{4,s_1}}{16} && (4.35)
\end{aligned}$$

The actual sources, though assumed *i.i.d.*, empirically will not measure identical cumulant measures. For them, $r \to 1 \Rightarrow a \to 0$. Using this values,

$$\begin{aligned}
& c_{4,y_1} - c_{4,y_2} \approx \epsilon, \text{ where, } \epsilon \to 0 \text{ as } r \to 1. && (4.36)\\
\Rightarrow & \sin\theta\cos\theta\cos 2\theta(-\frac{b_4}{6} + 1) = \epsilon && (4.37)
\end{aligned}$$

The solution implies that the stationary points are $\theta = 0{+}\theta(\epsilon)$, $\theta = pi/2{+}\theta(\epsilon)$ and $k_{4,s_1}{+}k_{4,s_2} = 6$, where $\theta(\epsilon)$ is a very small angle, as a function of ϵ.

Case-II(b) Let the sources are asymmetric, with zero kurtosis and un-identical independent distributions. Accordingly, $k_{4,s_1} = k_{4,s_2} = 0$ The equation (4.27) reduces to,

$$\begin{aligned} k_{3,s_1}c_{3,y_1} &= k_{3,s_2}c_{3,y_2} \\ \Rightarrow a(c_{3,y_1} - c_{3,y_2}) &= 0 \end{aligned}$$

This same as the results for **Case-I(b)**. That is, $\theta = 0$, $\theta = \frac{\pi}{2}$ and $k_{3,s_1} = k_{3,s_2}$ are the stationary points. So, there do not exist a spurious local minima of an entropy measure for asymmetric, zero-kurtosis, un-identical independent sources.

4.4.2 Extrema Analysis of kurtosis measure for near Gaussian and independent sources

Taking $\Phi^{k4}(\mathbf{r_i}) = kurtosis(y_i)$, the equation (4.10) can be written as:

$$\begin{aligned} &\nabla k4(\mathbf{r_1})\mathbf{r_2}^T = \nabla k4(\mathbf{r_2})\mathbf{r_1}^T \\ \Rightarrow & r_{21}\frac{\partial k4(\mathbf{r_1})}{\partial r_{11}} + r_{22}\frac{\partial k4(\mathbf{r_1})}{\partial r_{12}} = r_{11}\frac{\partial k4(\mathbf{r_2})}{\partial r_{21}} + r_{12}\frac{\partial k4(\mathbf{r_2})}{\partial r_{22}} \\ \text{Also, } & \frac{\partial k4(\mathbf{r_i})}{\partial r_{ij}} = 4r_{ij}^3\mu_{4,s_j} + 12r_{ij}r_{ik}^2,\ k \neq j \\ \Rightarrow & 4r_{11}^3r_{21}\mu_{4,s_1} + 12r_{11}r_{12}^2r_{21} + 4r_{12}^3r_{22}\mu_{4,s_2} + 12r_{11}r_{12}^2r_{21} \\ & = 4r_{11}r_{21}^3\mu_{4,s_1} + 12r_{11}r_{21}r_{22}^2 + 4r_{12}r_{22}^3\mu_{4,s_2} + 12r_{12}r_{21}^2r_{22} \end{aligned} \tag{4.38}$$

case-I:- Near Gaussian, *i.i.d.* sources

Assuming *i.i.d.* sources, $\mu_{4,s_1} = \mu_{4,s_2} = \mu_4$. Re-writing equation (4.38), we get:

$$\begin{aligned} \Rightarrow & \{4\cos\theta^3(-\sin\theta) - 4(-\sin\theta)^3\cos\theta\}\mu_{4,s_1} \\ & + \{4\sin\theta^3\cos\theta - 4\sin\theta\cos\theta^3\}\mu_{4,s_2} \\ & + 12\cos\theta(-\sin\theta)(\sin\theta^2 \\ & - \cos\theta^2) + 12\sin\theta\cos\theta(\cos\theta^2 - \sin\theta^2) = 0 \\ \Rightarrow & 8\sin\theta\cos\theta\cos2\theta(\mu_4 - 3) = 0 \end{aligned} \tag{4.39}$$

This proves that $\theta = 0, \frac{pi}{2}, \frac{pi}{4}$ and $\mu_4 = 3$ are the stationary points of the kurtosis measure. So, there do not exist spurious local maxima of kurtosis measure for near Gaussian, *i.i.d.* sources.

case-II:- Near Gaussian, un-identical but independent sources

Assuming un-identical but independent sources, $\mu_{4,s_1} \neq \mu_{4,s_2}$ and $|\mu_{4,s_1} - \mu_{4,s_2}| = a$. Let $\mu_{4,s_1} > \mu_{4,s_2} \Rightarrow \mu_{4,s_1} = \mu_{4,s_2} + a$. Re-writing equation 4.38,

$$\begin{aligned} &4\sin\theta\cos\theta(-\cos 2\theta)(\mu_{4,s_2} + a) + 4\sin\theta\cos\theta(-\cos 2\theta)\mu_{4,s_2} \\ &\quad + 24\sin\theta\cos\theta\cos 2\theta = 0 \\ &\quad 4\sin\theta\cos\theta\cos 2\theta(2(\mu_{4,s_2} - 3) + a) = 0 \end{aligned} \tag{4.40}$$

So, $\theta = 0, \frac{pi}{2}, \frac{pi}{4}$ and $(\mu_4 - 3) = -\frac{a}{2} \Rightarrow \mu_{4,s_1} + \mu_{4,s_2} = 6$ are the stationary points of the kurtosis measure.

Now, let $\mu_{4,s_1} < \mu_{4,s_2} \Rightarrow \mu_{4,s_1} = \mu_{4,s_2} - a$

$$\begin{aligned} &4\sin\theta\cos\theta(-\cos 2\theta)(\mu_{4,s_2} - a) + 4\sin\theta\cos\theta(-\cos 2\theta)\mu_{4,s_2} \\ &\quad + 24\sin\theta\cos\theta\cos 2\theta = 0 \\ &\quad 4\sin\theta\cos\theta\cos 2\theta(2(\mu_{4,s_2} - 3) - a) = 0 \end{aligned} \tag{4.41}$$

So, $\theta = 0, \frac{pi}{2}, \frac{pi}{4}$ and $(\mu_4 - 3) = \frac{a}{e}2 \Rightarrow \mu_{4,s_1} + \mu_{4,s_2} = 6$ are the stationary points of the kurtosis measure. Overall, there do not exists spurious local maxima of kurtosis measure for all - *i.i.d* or un-identical but independent; uni-modal or multi-modal; near Gaussian sources.

4.5 Conclusions on the Extrema Analysis of the Conventional Contrasts

The empirical observations in section 4.3 and the theoretical analysis in the current section 4.4 brings the following conclusions.

- The $\Phi^{hyi}(\mathbf{y})$ measure shows spurious local minima for densities f, g and j. The measure, show spurious minima that is global for density i. The theoretical analysis in the articles support the existence of spurious local minima for some of the multimodal densities but spurious global minima is not justified. Similar empirical results are obtained for the $\Phi^{radH}(\mathbf{y})$. The same extrema analysis is valid for the $\Phi^{radH}(\mathbf{y})$ measure after applying smoothening to the separated sources.

- The $\Phi^{k4}(\mathbf{y})$ measure does not show existence of spurious maxima for any densities. Instead, it shows shift of the global maxima for densities i and t.

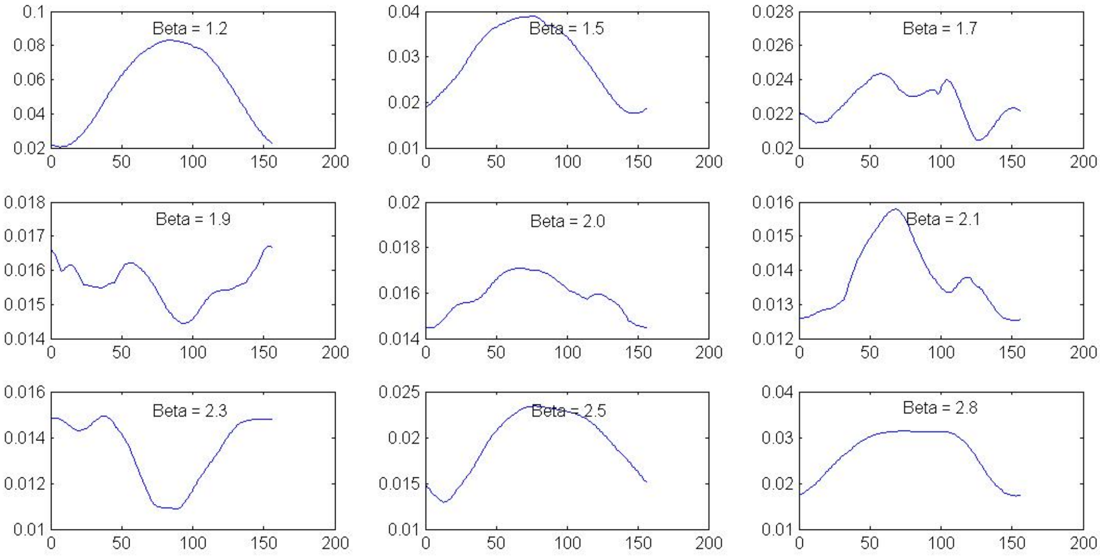

Figure 4.4: The plots of independence measures $\Phi^{hyi}(\mathbf{y})$ versus theta for Generalized Gaussian Distributions with varying shape parameters

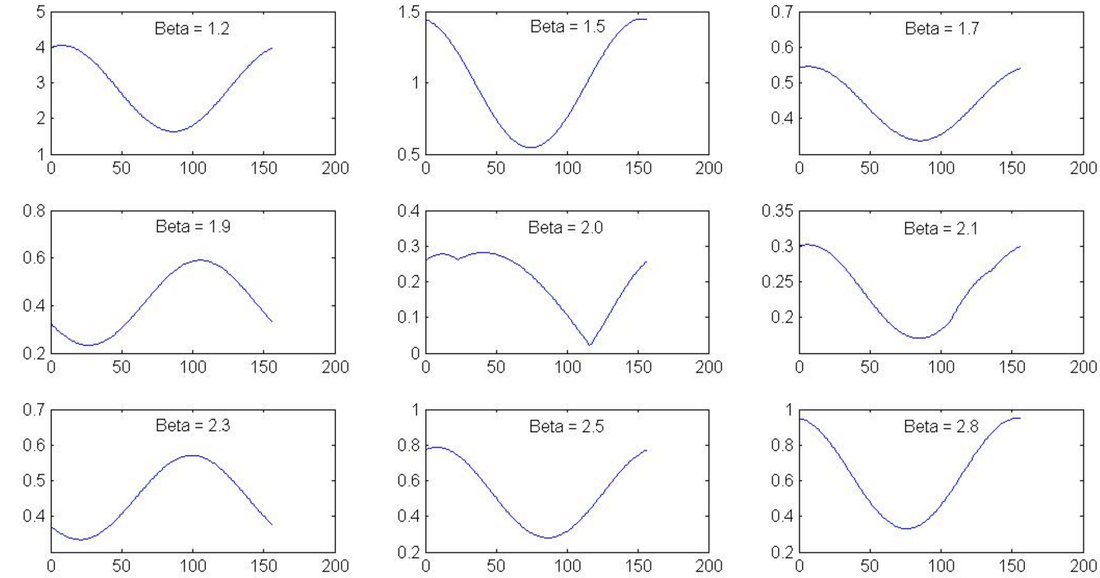

Figure 4.5: The plots of independence measures $\Phi^{k4}(\mathbf{y})$ versus theta for Generalized Gaussian Distributions with varying shape parameters

- The previous analysis in **Case I(a)** for the entropy based independence measure and the analysis in **Case I** for the kurtosis based independence measure give a hint that for the densities with near zero kurtosis the gradient is also near zero. This may have caused spurious minima, local or global, due to randomization and limited samples. To verify the correctness of this logical possibility, a separate experiment was done. The ICA solutions were obtained for the pairs of sources with GGD distributions for varying shape parameter. The results are plotted for $\Phi^{hyi}(\mathbf{y})$ measure in figure 4.4 and the similar are plotted for $\Phi^{k4}(\mathbf{y})$ measure in figure 4.5. It is found that either the existence of spurious minima (local or global) or the shift of the global occurs, specifically for the GGD distributions with the shape parameter $\beta \in [1.7, 2.5]$ that is when they are well near Gaussian with $k_4 \in [-0.5, 0.5]$. The theoretical analysis in the current section is for unimodal sources only and can not applied as it is to the multi-modal densities. But, possibly the behavior for the multi-modal density i is also due to the same reason as $k_4 = -0.5$ for this density. Separate analysis need be done for this.

- The behavior for density t is justified by **Case I(b)** for the entropy based independence measure and the analysis in **Case I** for the kurtosis based independence measure.

4.6 Empirical Extrema Analysis of Derived BSS Contrasts

The experiment is designed to test the existence of spurious local minima in the optimization landscape of the derived contrast for BSS in Chpater 2 of two *i.i.d.* sources with varying distributions. The contrast estimation was supported through both Silverman's ROT and ExROT method. The number of samples (N) were kept 300.

The contrasts tested include the derived LSFD (LSFD with ROT for bandwidth parameter selection), LSFD (with ExROT for bandwidth parameter selection), LSFD2 (LSFD2 with ROT for bandwidth parameter selection), LSFD2 (with ExROT for bandwidth parameter selection), LSGFD (LSGFD with ROT for bandwidth parameter selection), LSGFD (with ExROT for bandwidth parameter selection), LSGFD2 (LSGFD2 with ROT for bandwidth parameter selection), LSGFD2 (with ExROT for bandwidth parameter selection) and existing least squares based independence measures LSMI (with Cross-Validation (CV) for bandwidth parameter selection) (120), LSMI2 (with CV for bandwidth parameter selection) (106) for comparision. There are used same previous 21 types of distributions and used first 20 (type a to t) for this experiment. All 21 distributions are shown in the Figure 4.1.

The results of the Experiment for local minima analysis are shown in terms of the plots of negative of the contrast value versus the rotation angle theta. The minima of the plots corresponds to the actual sources. Ideally, it should be at $\theta = 0$ or $\pi/2$. The plots show the θ values in radian multiplied by 100. The comparative study shows that the local minima plots of all the derived estimators (fig. 4.6 to fig. 4.13) are far better than LSMI in fig. 4.14 and LSMI2 in fig.

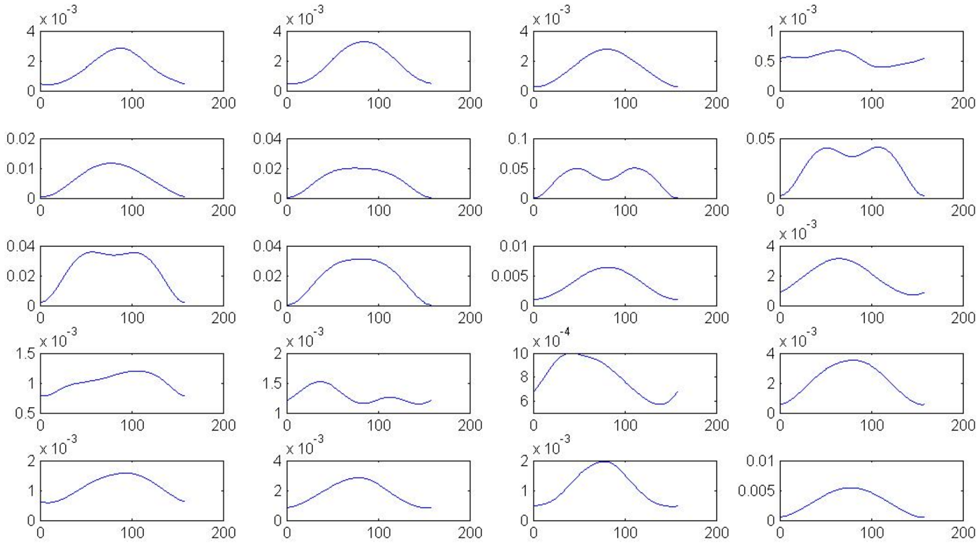

Figure 4.6: Plots of LSFD Contrast estimated with bandwidth parameter through ROT versus theta value for the first 20 distributions a-s stacked rowwise

4.15 estimators those using the same type of least squares based estimation method. They are also better to kernel based naive estimator QMI_{ED} estimator (the plots of QMI_{ED} are not shown here). They are comparable to those of conventional contrasts. For high kurtotic density type (d), all the derived contrasts give multiple local minima. The LSGFD2 estimator with ExROT as bandwidth parameter selection in fig. 4.13 has the best performance for multimodal distributions compare to all other contrasts, though it has local minima for distributions Type d (student with 5 degrees of freedom as high kurtotic) and Type o (symmetric mixture of four Gaussians unimodal). One more fact observable is, the local minima plots of gradient based contrasts (LSGFD and LSGFD2) are having small gradient compare to the Φ^{hyi} contrast. The plots also gives the performance comparison of ExROT and ROT on BSS contrasts. The ExROT has performed definitely better than the ROT. But, the overall performance of the derived contrasts is not expected to get largely improved by the ExROT.

The optimization landscape analysis of the derived and conventional contrasts is done. Now, let us see how much separation quality they achieve when used as a BSS contrast; whether the optimization algorithm can overcome the ideal landscape violations or not? For that, we need a BSS or ICA algorithm that can have global optimization technique and can be used to compare performances of various contrasts. The following section proposes such an ICA algorithm.

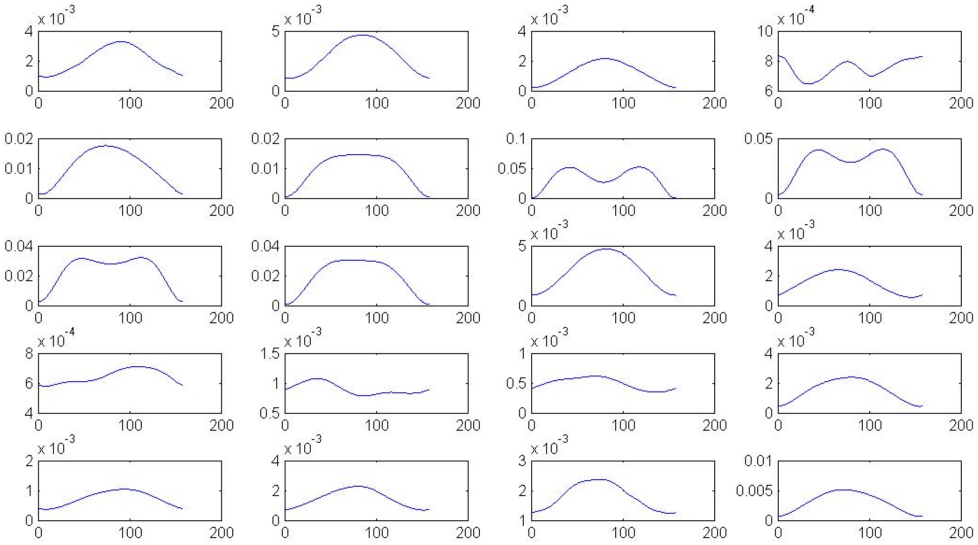

Figure 4.7: Plots of LSFD Contrast estimated with bandwidth parameter through ExROT versus theta value for the first 20 distributions a-s stacked rowwise

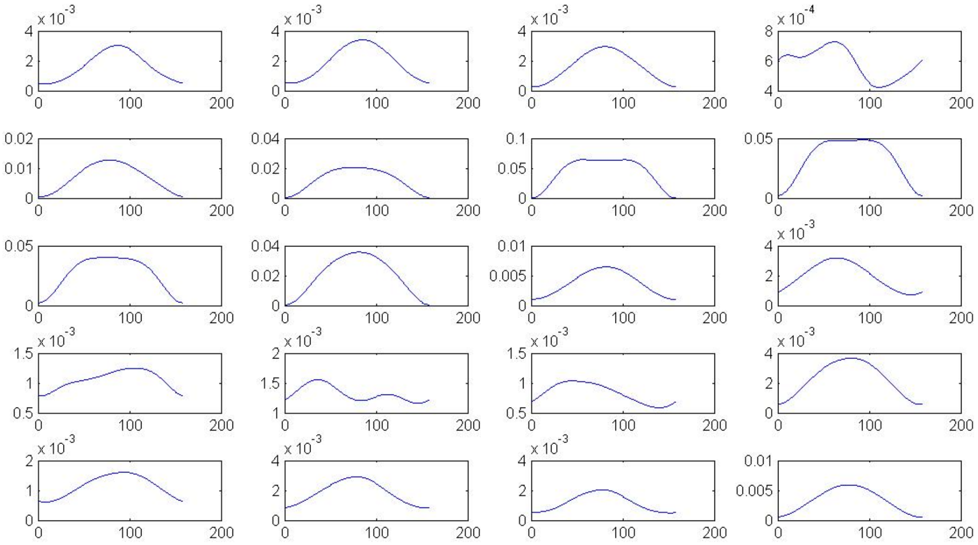

Figure 4.8: Plots of LSFD2 Contrast estimated with bandwidth parameter through ROT versus theta value for the first 20 distributions a-s stacked rowwise

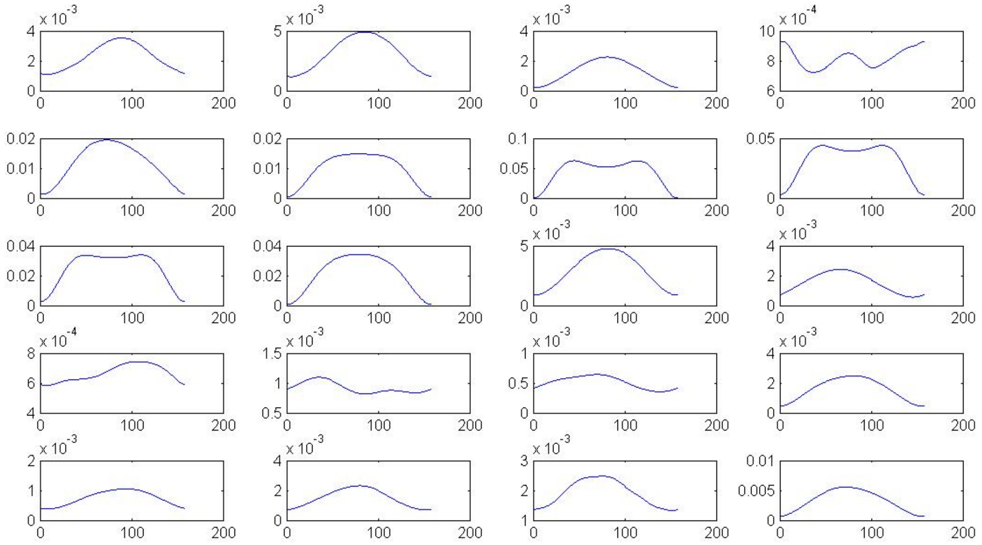

Figure 4.9: Plots of LSFD2 Contrast estimated with bandwidth parameter through ExROT versus theta value for the first 20 distributions a-s stacked rowwise

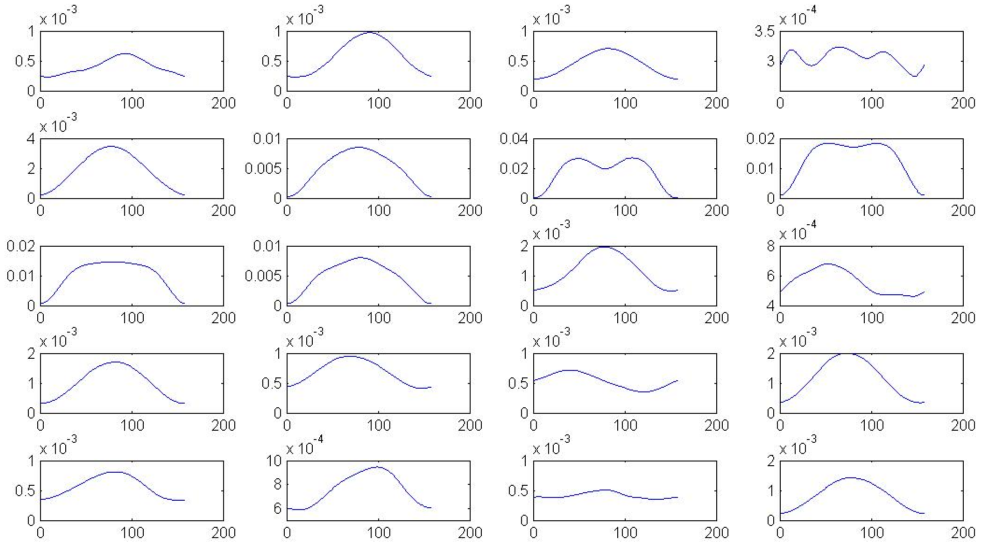

Figure 4.10: Plots of LSGFD Contrast estimated with bandwidth parameter through ROT versus theta value for the first 20 distributions a-s stacked rowwise

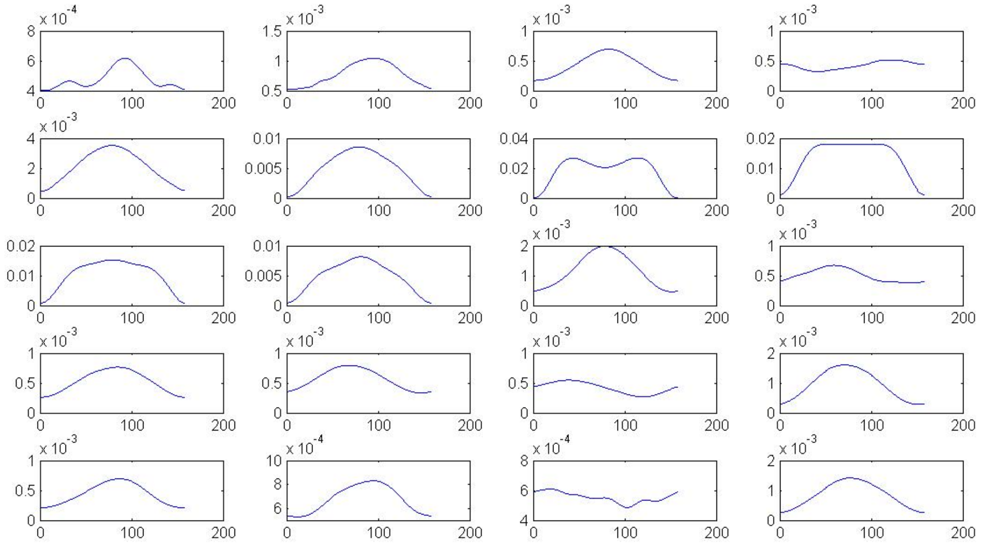

Figure 4.11: Plots of LSGFD Contrast estimated with bandwidth parameter through ExROT versus theta value for the first 20 distributions a-s stacked rowwise

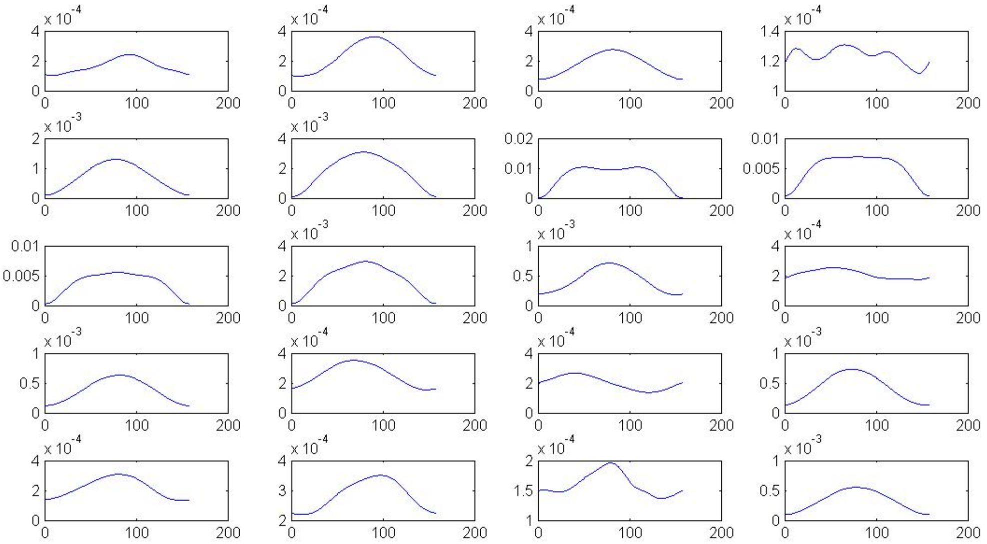

Figure 4.12: Plots of LSGFD2 Contrast estimated with bandwidth parameter through ROT versus theta value for the first 20 distributions a-s stacked rowwise

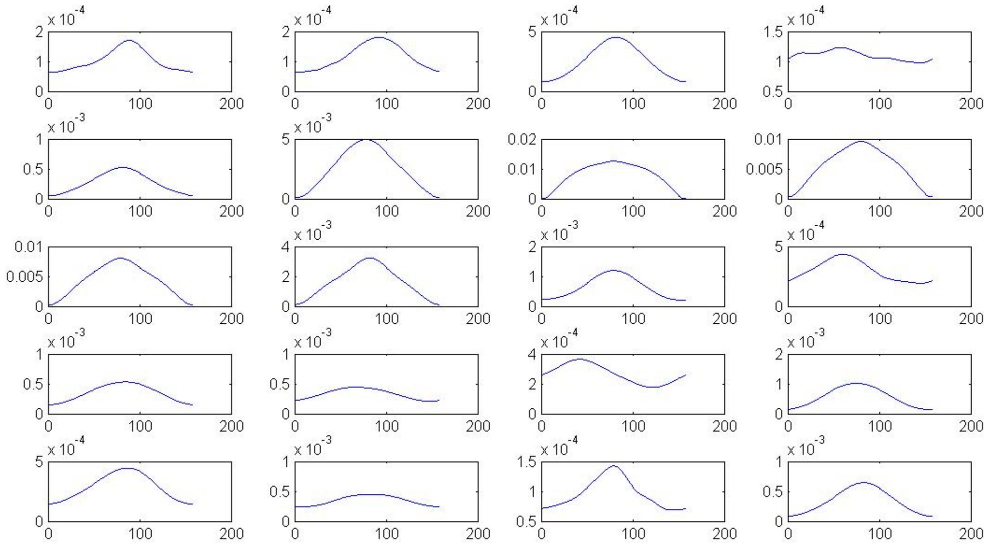

Figure 4.13: Plots of LSGFD2 Contrast estimated with bandwidth parameter through ExROT versus theta value for the first 20 distributions a-s stacked rowwise

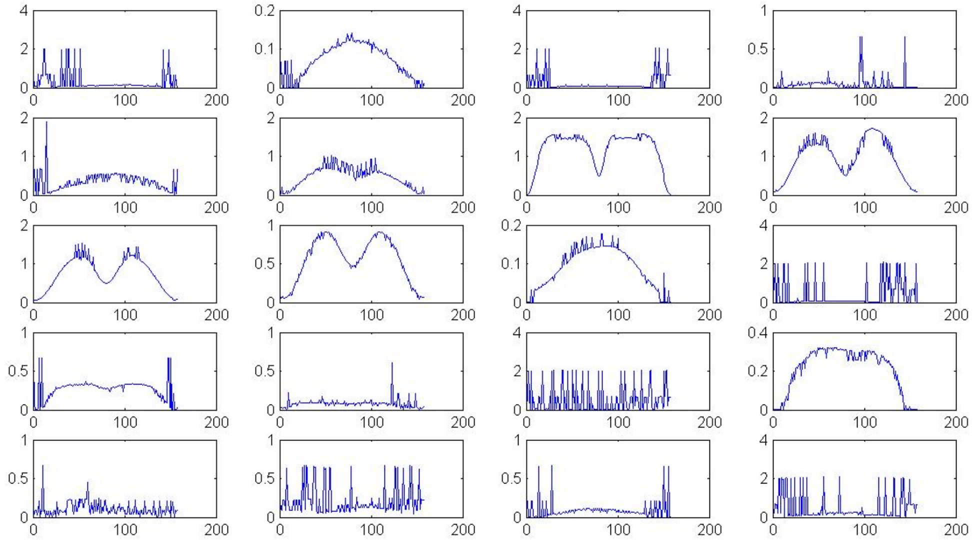

Figure 4.14: Plots of LSMI Contrast estimated with bandwidth parameter through CV versus theta value for the first 20 distributions a-s stacked rowwise

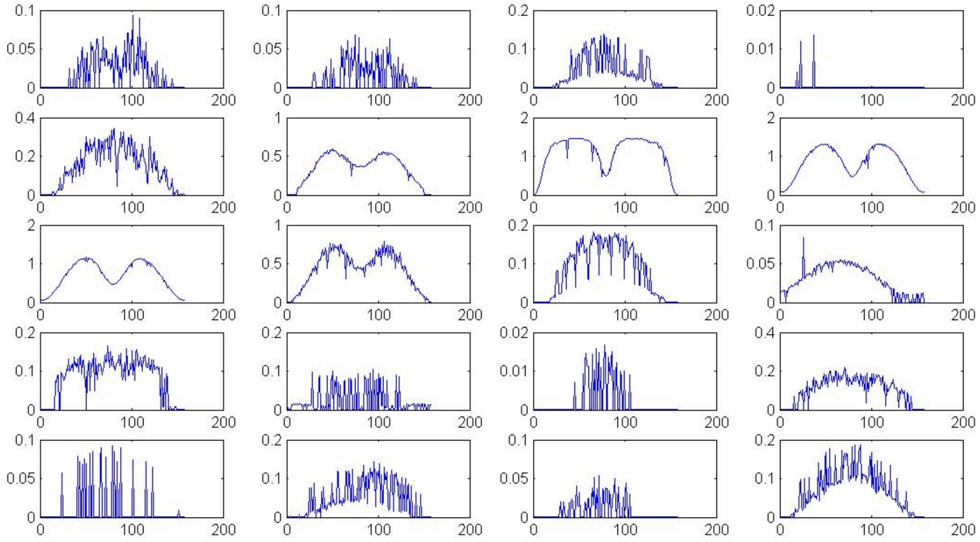

Figure 4.15: Plots of LSMI2 Contrast estimated with bandwidth parameter through CV versus theta value for the first 20 distributions a-s stacked rowwise

4.7 The SRICA Proposal

Here, the orthogonal approach for ICA, explained in Chapter 1.3.1 is used. The goal is to obtain ICs (y_is) from the correlated mixtures (x_is) of them. The method can be summarized in two steps.

1. Find zero mean whiten components using PCA (Principle Component Analysis) through EVD (Eigen Value Decomposition) or SVD (Singular Value Decomposition), as a compulsory step. Let a zero mean observed mixture data matrix **x**, be linearly transformed through a whitening matrix $\mathbf{V}$, to give a zero mean, univariant, whiten data matrix $\mathbf{z}$. $\mathbf{z} = \mathbf{Vx} = \mathbf{VAs}$

2. Search for the optimal $n-$dimensional angle of rotation, through a global search technique, to transform the whiten components into independent components. The optimality can be defined through either maximization of independence or minimization of dependency. Let $\mathbf{R}$ be the rotation orthogonal matrix. Then,

$$\mathbf{y} = \mathbf{Rz} = \mathbf{RVAs} = \mathbf{WAs} \tag{4.42}$$

where, $\mathbf{W} = \mathbf{RV}$ is the estimated unmixing matrix.

As the algorithm is searching for a perfect angle of rotation, let it be called the *Search for Rotation based ICA (SRICA)* algorithm. The current chapter uses GA as the global search method.

4.7.1 How to rotate in a higher dimensional space?

A two dimensional rotation can be achieved through Given's rotation. To convert a 3-dimensional rotation matrix Q into an identity matrix I, we need three, 2-dimensional rotations.

$$R_{yz}R_{xz}R_{xy}Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$\Rightarrow R = R_{yz}R_{xz}R_{xy} = Q'$ or $Q = R_{xy}^{-1}R_{xz}^{-1}R_{yz}^{-1} = R'$ An $n \times n$ rotation matrix will have $M = (n-1) + (n-2) + \ldots + 1 = n(n-1)/2$ entries below the diagonal to zero. So, it can be concluded that rotation in n dimension can be performed through $n(n-1)/2$ two dimensional rotations sequentially performed.

4.7.2 The Genetic algorithm

GA tries to imitate the evolution process, where natural selection is the guiding principle. Nature selects those, which are fit to survive. The fitness is measured based on the optimization function called the fitness criteria. GA represents the solution as a binary string or a float variable. GA, randomly selects a set of possible solutions, called initial population, from the given solution space (total population). Assigns fitness to all. Then assuming the current population as the parent, generates a new child population using selection, crossover and mutation operators. The operators are some definite rules, using random number generation and fitness. The child becomes parents for the next generations. Thus, iteratively. it achieves the optimal solution. The GA, with elitist model, keeps track of the best obtained so far. The stopping criteria can be based on number of iterations or optimality.

The genetic operators defined are:

- *Selection:* Selects the parents, which should contribute to the next generation
- *Crossover:* Forms the pairs and decides whether and how to breed
- *Mutation:* Individual parents can be mutated, changed with very small probabilities

4.7.3 Summarizing the suggested solution

The above discussed solution for ICA has three components.

1. Data whitening technique
2. GA as an optimization technique

3. Fitness function for GA as an optimality criteria

Overall, the method differs from the existing methods, not just in using GA as an optimization technique. Compare to most other ICA techniques, which search for $\mathbf{W}$ or w_is in the gradient directions and then separately following the orthogonalization steps; SRICA is searching for $\mathbf{W}$, among the rotational matrices only. Conventionally, most of the ICA techniques aim orthogonalization of the w_is separately and not just search for $\mathbf{W}$ from a rotational transformation matrices, as in the suggested solution. Here, the GA simply, makes it possible to search for the optimal from the infinite set of rotational transformation matrices. So, it is worth naming the technique as Search for Rotation based ICA (SRICA) algorithm. By varying the optimality criteria or the global search algorithm (other than GA) used, a different variant to the SRICA algorithm can be derived. GA has been used for a post-nonlinear BSS (53, 104), but never before for a linear BSS or ICA problem.

4.7.4 SRICA characteristics and Algorithm Complexity

- For data whitening, the first component of SRICA, numerically stable versions are available.
- GA, the second component, is a technique, applicable to optimize an objective function which is constrained or unconstrained or non differentiable or discontinuous or probabilistic or complex. So, SRICA can be used with any such type of independence measure. The GA requires neither about the fitness landscape nor about the solution any previous information. So, SRICA could be completely blind. But, if available either a partial or full prior information regarding the possible distribution of few or all sources or the mixing matrix, then that could be easily incorporated in the fitness criteria. Accordingly, SRICA could be used for ICA, BSS or semi-BSS problems.
- GA has a very good characteristic to converge to a global optima and thus SRICA finds if not optimal, near optimal solution.
- All above advantages come at the cost of more computation.
- The SRICA can be thought having three variable parameters: 1) a global search based optimization technique 2) ICA contrast as an optimization criteria and 3) estimation technique of the optimization criteria. There can be derived variants of the SRICA by varying either of these variable parameters. Overall, SRICA facilitates to tailor made or customize the algorithm, as on the performance or application requirement by selecting specific optimization technique, optimization criteria and the estimation method.

For an n-dimensional ICA problem, the GA has to search for optimality of $d = n(n-1)/2$ variables. If these m variables are independently affecting the fitness function f i.e. if the fitness function can be represented as $f(\theta_1, \theta_2, ..., \theta_d) = \sum_{i=1}^{d} f(\theta_i)$ then $f(\cdot)$ is called a decomposable (separable) function. In case of a decomposable fitness function, the optimal value of parameters can be found by just running the GA for each of the variable one by one, irrespective of their order. So, the algorithm complexity for continuous or float type of m variable, elitist GA with decomposable fitness function, would have been just $O(kd)$ or maximally $O(kd \ln d)$ (107), where assumed number of function evaluation for a single variable is k and number of computations for a function evaluation is 1. In case of a non-decomposable (nonseparable) fitness function optimization, GA algorithm complexity may go up to kd^d or $k \exp(d \ln d)$ theoretically and more significantly, the convergence is not assured (107). This is true for any other Evolutionary Algorithm (EA). In the EA literature, optimization of nonseparable functions is an open unsolved problem that face the problem of 'curse of dimensionality'. In this chapter, GA is used as a representative of the EA community. As proved in the Chapter 6, the simultaneous BSS or ICA contrasts are nonseparable optimization functions. Also the chapters then report the efforts either to avoid misconvergence problem or to reduce the algorithm complexity in case of large scale nonseparable function optimization.

One possible way to avoid application of GA in large scale for BSS is derived by (33) using pairwise independence principle. Here, to search for optimal d number of 2-d angle of rotations, $n - 1$ number of sweeps are performed, whereas in each sweep the all 2-d rotations are individually found. The multiple sweeps are expected to remove the shift of optimal due to mutual dependencies. The search scheme is used exhaustive search and is identified as 'eqpart' scheme for GA.

4.8 Simulations

There have been designed three experiments with the aim to check the validity of the SRICA algorithm. The first experiment is to verify performance of SRICA and derived contrast against varying source distributions compare to the existing other contrasts and ICA algorithms. For that SRICA is used with some of the conventional contrasts and some of the derived contrasts. For all the simulations in this chapter, the existing 5 ICA algorithms for comparision are: FastICA (deflation mode and $tanh$ nonlinearity) by (64), EFICA by (72), NPICA by (18) and RADICAL (without augmentation) by (75) with their standard parameters. The fifth algorithm for comparision is done using Exhaustive search method of (75) on Φ^{hyi} contrast. For all the experiments with 2 sources following GA parameters were tuned to balance exploration and exploitation abilities. The selection of the individuals was done through normalized geometric distribution based on ranking through fitness, with the probability to select the best being 0.08. The float GA was using - arithmetic crossover (linear combination of the parents) and heuristic crossover (linear extrapolation, with the child generated near the more fit parent) - both with equal probability and total probability of crossover being .90. The mutation probability was kept to be 0.95 for the initial 10% of the total generations. Then, for the remaining 90% generations, it was reduced to 0.05. The initial high mutation probability takes care about the exploration and then the lower mutation probabilities allow for exploitation. In the experiments the population size was taken to be 10 and the maximum number of generations were kept to be 15. The termination criteria for all the experiments was based on maximum generations.

The parameter for performance comparision is explained below. As mentioned, **y** is a permuted and scaled to univariant version of **s**. If **u** is zero mean, univariant source matrix, then $\mathbf{u} = \mathbf{D^{-1}}(\mathrm{s} - \bar{\mathrm{s}})$ where, **D** is a diagonal matrix, with inverse of the standard deviations of all s_i as the diagonal. So, from equation 1.13

$$\mathbf{y} = \mathbf{PWADu} = \mathbf{Gu}$$

where P is a permutation matrix and $\mathbf{G} = \mathbf{PWAD}$ is the gain matrix. Ideally the so called performance or gain matrix **G** should be an identity matrix. Based on this criteria, the *Amari Performance Index (API)* measures the deviation from diagonalization, of the gain matrix.

$$API(G) = \sum_{i=1}^{n}\left(\sum_{j=1}^{n}\frac{|g_{ij}|}{max_k\,|g_{ik}|} - 1\right) + \sum_{j=1}^{n}\left(\sum_{i=1}^{n}\frac{|g_{ij}|}{max_k\,|g_{kj}|} - 1\right) \tag{4.43}$$

So, better the performance, the measure should be more nearer to zero. The following experiments are done.

Experiment: To verify the performance of SRICA and derived contrasts against varying distributions

There have been generated mixtures of two '*i.i.d.*' sources, with varying distributions and sample size (N) of 300. The results were obtained using 10 simulation trials, for 20 types of distributions. The first 18 types (a to r) of distributions are suggested by (10) and three more skewed types of distributions were added to test the performance of the ICA algorithms against skewed sources. The s type is a GGD with skewness $s = -0.25$ (left skewed) and kurtosis $k = 3.75$ and the t type is a GGD with skewness $s = 0.75$ (right skewed) and kurtosis $k = 0$. Both the distributions are generated using Power Method with parameters $b = 0.75031534111078$, $c = -0.02734119591845$, $d = 0.07699282409939$ for s type and $b = 1.11251460048528$, $c = 0.17363001955694$ and $d = -.05033444870926$ for t type. The u type is a Gaussian distribution that is added for experiment randomly selecting the density. All 21 distributions are shown in the Figure 4.16. To study the performances of different independence measure on the source separation SRICA is used with

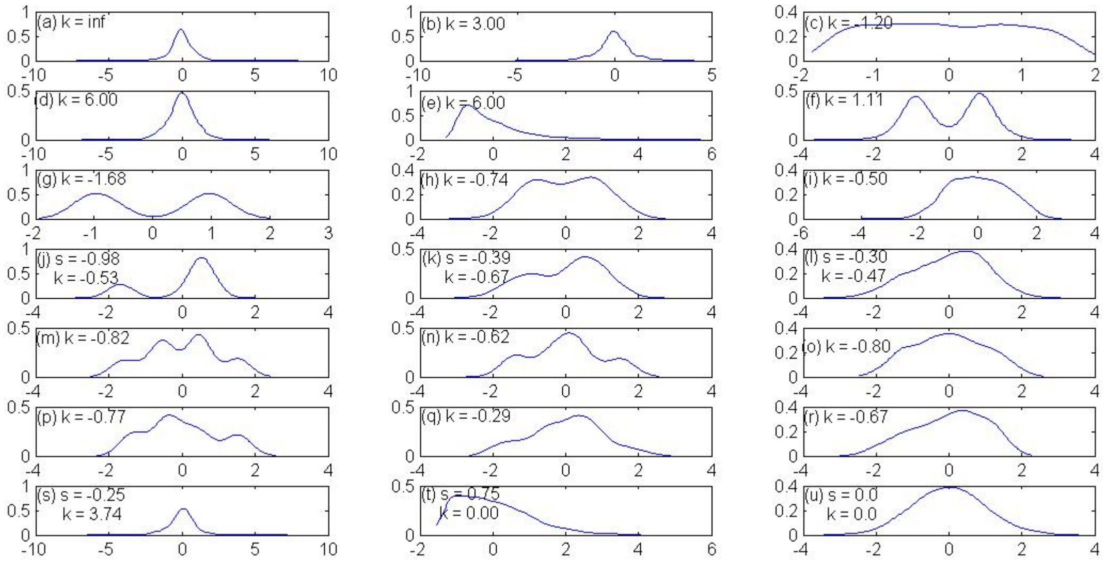

Figure 4.16: Probability density functions of sources with their kurtosis: (a) Student with 3 degrees of freedom; (b) double exponential; (c) uniform; (d) Student with 5 degrees of freedom; (e) exponential; (f) mixture of two double exponentials; (g)-(h)-(i) symmetric mixtures of two Gaussians: multimodal, transitional and unimodal; (j)-(k)-(l) nonsymmetric mixtures of two Gaussians, multimodal, transitional and unimodal; (m)-(n)-(o) symmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (p)-(q)-(r) nonsymmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (s) left skewed Generalized Gaussian Distribution(GGD); (t) right skewed GGD; (u) Gaussian distribution

the five conventional contrasts Φ^{jskld}, Φ^{kld}, Φ^{hyi}, Φ^{k4} as defined in 4.2 and the fifth Φ^{radH}, as per entropy definition in (75). The SRICA is also used with newly derived $\Phi^{QMI_{ED},ROT}$ (96) using Quadratic Mutual Information based on Euclidean distance (QMI_{ED}) with bandwidth parameter through ROT, $\Phi^{QMI_{ED},ExROT}$ as QMI_{ED} with bandwidth parameter through ExROT, Φ^{LSMI} and Φ^{LSMI2}. The last two techniques use bandwidth parameter selection through cross-validation. For the PDF estimation, the kernel method with Gaussian kernel had been used. The SRICA is also used with derived contrasts with 7 versions $\Phi^{LSFD,ROT}$, $\Phi^{LSFD,ExROT}$, $\Phi^{LSFD,ExROT}$, $\Phi^{LSFD2,ROT}$, $\Phi^{LSGFD,ROT}$, $\Phi^{LSGFD,ExROT}$, $\Phi^{LSGFD2,ROT}$.

The results are tabulated in Table 4.1 to Table 4.3. For a density type, the performance of all the ICA algorithms is available in a column. The boldface values indicate best and boldface with italics indicate $API > 0.1$ i.e. performance is not acceptable within permissible range. The second last column in Table 4.3 indicate mean of the results for all the 20 density types and then last column in the same table show the result for two sources randomly selected from 21 types.

The above experimental results for two *i.i.d.* sources against varying source PDF verifies

Table 4.1: Performances of SRICA with various independence measures; varying distributions form Type (a) to Type (h) in fig. 4.16 Comparison with other ICA algorithms using API as the performance measure. Number of sources = 2 with varying source distributions; number of samples = 300; GA parameters: float version, population size = 10, generations = 15. All other entries are median of 10 trials.PDF Types: (a) Student with 3 degrees of freedom; (b) double exponential; (c) uniform; (d) Student with 5 degrees of freedom; (e) exponential; (f) mixture of two double exponential; (g)-(h)-(i) symmetric mixtures of two Gaussians: multimodal, transitional and unimodal

pdf → SRICA(Φ)	a	b	c	d	e	f	g	h
Φ^{jskld}	**0.0282**	0.0469	0.0386	0.0942	0.0313	0.0263	0.0090	0.0184
Φ^{kld}	0.0337	0.0565	0.0382	***0.1299***	0.0330	0.0263	0.0090	**0.0136**
Φ^{hyi}	0.0395	0.0495	0.0317	***0.1083***	0.0274	0.0264	0.0141	0.0184
Φ^{k4}	0.0462	0.0441	0.0311	0.0718	0.0547	0.0342	0.0087	0.0138
Φ^{radH}	0.0864	0.0565	0.0420	***0.1791***	0.0223	0.0356	0.0129	0.0140
$\Phi^{QMI_{ED},ROT}$	0.0457	0.0440	***0.9468***	**0.0717**	0.0539	***0.9250***	***0.9847***	***0.9677***
$\Phi^{QMI_{ED},ExROT}$	0.0424	0.0447	***0.9418***	0.0804	0.0589	***0.9300***	***0.9847***	***0.9677***
Φ^{LSFDR}	0.0539	0.0411	0.0407	***0.1488***	0.0305	0.0264	0.0141	0.0184
Φ^{LSFD}	***0.1027***	0.0459	0.0426	***0.1639***	0.0262	0.0262	0.0086	0.0184
Φ^{LSFD2R}	0.0499	0.0418	0.0407	***0.1455***	0.0258	0.0262	0.0196	**0.0136**
Φ^{LSGFDR}	***0.2831***	0.0608	0.0382	***0.4147***	0.0315	0.0261	0.0087	**0.0136**
Φ^{LSGFD}	***0.7885***	***0.9204***	***0.6353***	***0.8056***	***0.8344***	***0.7074***	***0.9826***	***0.9676***
$\Phi^{LSGFD2R}$	***0.4897***	***0.3891***	0.0945	***0.4546***	***0.4987***	0.0793	***0.1035***	0.0352
Φ^{LSMI}	***0.1651***	***0.2017***	0.0673	***0.4410***	0.0488	0.0381	0.0145	0.0184
Φ^{LSMI2}	***0.7852***	***0.8518***	0.0452	***0.8293***	0.0542	0.0347	0.0105	0.0177
Algorithm	Performance of Existing ICA Algorithms							
FastICA	0.0481	0.0512	0.0430	0.0897	0.0646	0.0424	0.0134	0.0154
EFICA	0.0661	**0.0360**	**0.0205**	0.0781	0.0530	0.0666	0.0058	0.0265
NPICA	0.0346	0.0542	0.0248	***0.1083***	0.0239	***0.8902***	**0.0056**	***0.2232***
RADICAL	0.0611	0.0577	0.0487	***0.1121***	**0.0207**	**0.0271**	0.0132	**0.0136**
Ex.Search(hyi)	0.0837	0.0894	0.0411	***0.1536***	0.0321	0.0440	0.0181	0.0233

the derived contrasts for BSS. The following conclusions can be derived.

1. Baesd on mean of the performances Φ^{hyi}, using SRICA is the best ICA algorithm and for randomly selected densities RADICAL has performed best. The sum of marginal densities estimated through KDE is a contrast used by both SRICA and NPICA. But, SRICA(Φ^{hyi}) gives almost consistent performance. This indicates SRICA has been able to avoid local minima. So, as an optimization algorithm it proves the worth of search based global optimization techniques.

2. The proposed contrasts in the thesis have performed very well for multi-modal densities. That indicate their partial superiority over other algorithms. The gradient based contrasts need caution in use.

3. The $API > 0.1$ for some of the entries indicate that the corresponding algorithm has converged to the spurious optima - either local minima or shifted global. This has happened

Table 4.2: Performances of SRICA with various independence measures; varying distributions form Type (i) to Type (p) in fig. 4.16 Comparison with other ICA algorithms using API as the performance measure. Number of sources = 2 with varying source distributions; number of samples = 300; GA parameters: float version, population size = 10, generations = 15. All other entries are median of 10 trials.PDF Types: (g)-(h)-(i) symmetric mixtures of two Gaussians: multimodal, transitional and unimodal; (j)-(k)-(l) nonsymmetric mixtures of two Gaussians, multimodal, transitional and unimodal; (m)-(n)-(o) symmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (p)-(q)-(r) nonsymmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (s) left skewed Generalized Gaussian Distribution(GGD); (t) right skewed GGD; (u) Gaussian distribution

pdf $\rightarrow$ SRICA(Φ)	i	j	k	l	m	n	o	p
Φ^{jskld}	0.0198	0.0303	0.0490	**0.0590**	0.0229	0.0383	0.0766	0.0424
Φ^{kld}	0.0198	**0.0302**	0.0503	0.0591	0.0234	0.0382	0.0612	0.0420
Φ^{hyi}	**0.0197**	**0.0302**	0.0473	0.0574	0.0158	0.0362	**0.0552**	**0.0234**
Φ^{k4}	0.0227	***0.1805***	0.0377	***0.1316***	0.0360	0.0688	0.0663	0.0493
Φ^{radH}	0.0200	0.0345	0.0554	***0.1288***	0.0310	0.0342	***0.1765***	0.0363
$\Phi^{QMI_ED,ROT}$	***0.9585***	***0.7605***	***0.9338***	***0.7655***	***0.8945***	***0.8779***	***0.8560***	***0.9160***
$\Phi^{QMI_ED,ExROT}$	***0.9614***	***0.9286***	***0.9111***	***0.8920***	***0.8803***	***0.8808***	***0.8774***	***0.8874***
$\Phi^{LSFD,ROT}$	0.0198	0.0334	0.0569	0.0763	0.0416	0.0354	***0.1235***	0.0501
$\Phi^{LSFD,ExROT}$	0.0198	0.0348	0.0576	0.0774	***0.1031***	0.0482	0.0780	0.0546
$\Phi^{LSFD2,ROT}$	0.0198	0.0349	0.0567	0.0751	0.0348	0.0330	***0.1226***	0.0504
$\Phi^{LSGFD,ROT}$	0.0198	0.0354	0.0615	0.0939	**0.0136**	**0.0323**	***0.1110***	0.0315
$\Phi^{LSGFD,ExROT}$	***0.9615***	***0.9092***	***0.4273***	***0.5862***	***0.3138***	***0.1101***	***0.4379***	***0.5352***
$\Phi^{LSGFD2,ROT}$	0.0394	***0.4253***	***0.2840***	***0.6803***	***0.1028***	***0.1454***	***0.3398***	***0.3006***
Φ^{LSMI}	0.0268	0.0396	0.0720	***0.1187***	0.0538	0.0641	0.0594	0.0601
Φ^{LSMI2}	0.0241	0.0387	0.0449	0.0719	0.0409	0.0653	0.0850	0.0590
Algorithm	Performance of Existing ICA Algorithms							
FastICA	0.0281	***0.7992***	**0.0332**	***0.1376***	0.0607	***0.1282***	0.0627	***0.1169***
EFICA	0.0214	***0.7521***	0.0406	***0.1252***	0.0353	0.0626	0.0602	0.0381
NPICA	0.0217	***0.4680***	0.0449	0.0593	0.0165	***0.4590***	***0.1109***	0.0257
RADICAL	0.0230	0.0467	0.0579	***0.2029***	0.0208	***0.4543***	***0.7915***	0.0243
Ex.Search(hyi)	0.0297	0.0576	0.0843	***0.1358***	0.0317	0.0632	0.0929	0.0262

for all the algorithms, including NPICA. As stated as an open problem by the author in (18), such occurrences ($API > 0.1$) identify the distributions for those NPICA has misconverged. This also hampers NPICA's claim to seamlessly handle the large scale ICA problems.

4. The Φ^{QMI_ED}, Φ^{LSMI} and Φ^{LSMI2} indicate failure in most cases including those of - mean performance and performance on randomly selected densities.

Experiment: To verify the performance of SRICA in higher dimensions

The Experiment is done with gradually varying number of sources randomly selected from the given 21 sources. The sample size for all the experiments is kept constant as 600. The results are tabulated in the Table 4.4. The SRICA is used with LSFD, as a representative of the derived

Table 4.3: Performances of SRICA with various independence measures; against varying distributions form Type (q) to Type (u), mean performance of Types (a) to (u) and randomly selected two sources; in fig. 4.16 Comparison with other ICA algorithms using API as the performance measure. Number of sources = 2 with varying source distributions; number of samples = 300; GA parameters: float version, population size = 10, generations = 15. All other entries are median of 10 trials.PDF Types: (a) Student with 3 degrees of freedom; (b) double exponential; (c) uniform; (d) Student with 5 degrees of freedom; (e) exponential; (f) mixture of two double exponentials; (g)-(h)-(i) symmetric mixtures of two Gaussians: multimodal, transitional and unimodal; (j)-(k)-(l) nonsymmetric mixtures of two Gaussians, multimodal, transitional and unimodal; (m)-(n)-(o) symmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (p)-(q)-(r) nonsymmetric mixtures of four Gaussians: multimodal, transitional and unimodal; (s) left skewed Generalized Gaussian Distribution(GGD); (t) right skewed GGD; (u) Gaussian distribution

pdf → SRICA(Φ)	q	r	s	t	u	mean	rand
Φ^{jskld}	0.0496	**0.0525**	0.0736	0.0372	***0.5881***	0.0682	0.0679
Φ^{kld}	**0.0480**	0.0529	0.0683	0.0351	***0.5881***	0.0694	0.0672
Φ^{hyi}	0.0494	0.0526	**0.0525**	0.0333	***0.5628***	**0.0644**	0.0513
Φ^{k4}	0.0759	***0.1123***	0.0629	***0.3680***	***0.2375***	0.0835	0.0724
Φ^{radH}	***0.1120***	0.0889	0.0665	0.0167	***0.4518***	0.0810	0.0382
$\Phi^{QMI_ED,ROT}$	***0.8612***	***0.7974***	0.0628	***0.1885***	***0.6419***	***0.6454***	***0.6924***
$\Phi^{QMI_ED,ExROT}$	***0.5443***	***0.8885***	0.0614	***0.9235***	***0.4563***	***0.6735***	***0.8121***
$\Phi^{LSFD,ROT}$	0.0832	0.0723	0.0595	0.0473	***0.5634***	0.0779	0.0480
$\Phi^{LSFD,ExROT}$	0.0812	0.0718	***0.1142***	0.0470	***0.5780***	0.0857	0.0531
$\Phi^{LSFD2,ROT}$	0.0819	0.0847	0.0760	0.0474	***0.5705***	0.0786	0.0489
$\Phi^{LSGFD,ROT}$	0.0642	0.0734	***0.7228***	0.0421	***0.2146***	***0.1140***	0.0626
$\Phi^{LSGFD,ExROT}$	***0.1016***	***0.5070***	***0.8409***	***0.8868***	***0.2337***	***0.6425***	***0.5639***
$\Phi^{LSGFD2,ROT}$	***0.4824***	***0.4331***	***0.3971***	***0.3337***	***0.5216***	***0.3157***	***0.3002***
Φ^{LSMI}	***0.1587***	***0.1455***	***0.2022***	***0.1298***	***0.3324***	***0.1170***	0.0586
Φ^{LSMI2}	***0.1627***	0.0540	***0.8328***	0.0548	***0.1948***	***0.2075***	0.0692
Algorithm	Performance of Existing ICA Algorithms						
FastICA	***0.1551***	0.0981	0.0574	***0.2821***	***0.4565***	***0.1325***	0.0549
EFICA	***0.1874***	***0.1161***	0.0695	***0.4182***	***0.2496***	***0.1204***	0.0652
NPICA	0.0742	0.0500	0.0528	**0.0237**	***0.4159***	***0.1518***	0.0455
RADICAL	0.0807	***0.1258***	0.0618	0.0278	***0.6389***	***0.1386***	**0.0411**
Ex.Search(hyi)	0.0587	0.0741	0.0797	0.0420	***0.5329***	0.0854	0.0600

contrasts, and with Φ^{k4}, as a representative of the conventional contrasts. It is used with two different search schemes fro GA. The 'simultaneous' scheme, searches for the optimal in all search dimensions (d) simultaneously. The 'eqpart' scheme searches for the optimal, simultaneously only in one search dimension. But, to remove the affect due to inter-dependence it uses (d-1) sweeps. The 'eqpart' scheme requires $(d-1)*d*I_{min}$ amount of function evaluations; where, d is the number of search variables, I_{min} is the minimum number of evaluations required to search for optimal in one dimension simultaneously. As the ideal implementation requires large amount of computation, there have been used here $70*d$ function evaluations.

The results show that the SRICA with LSFD contrast performs comparable to other existing ICA algorithms. It almost performs second best to NPICA for 2, 4 and 8 sources. It could be noticed that with increasing number of sources, the performance of all the algorithms degrades. For, 16 sources all the algorithms; except EFICA; give $API > 0.1$. The SRICA with kurtosis contrast badly performs even for 8 sources. Overall, even SRICA, using GA as a global optimization technique; as well, the RADICAL with exhaustive search fails in higher dimensions.

Table 4.4: Performances of SRICA using varying independence measures in higher dimensions; Comparison with other ICA algorithms using API as the performance measure. Number of samples = 600; Each row entry is a median of the number of trials (niter) in column 2.

nsrc	niter	**srica(kurt)**	**srica(kurt)**	**srica(LSFD)**	**FastICA**	**EFICA**	**NPICA**	**radical**
		simultaneous	eqpart	eqpart				
2	50	0.0618	0.0616	0.0352	0.0615	0.0427	**0.0296**	0.0422
4	25	0.0966	0.0642	0.0426	0.0631	0.0500	**0.0380**	0.0484
8	10	***0.1880***	0.0843	0.0536	0.0760	0.0800	**0.0386**	0.0623
16	5	***0.2422***	***0.1555***	***0.1767***	***0.1023***	**0.0937**	***0.2179***	***0.2850***

Experiment: Performance of SRICA against real world sources

The experiment tests the performance of SRICA against real world sources. As a representative of derived contrasts, LSFD is considered with SRICA in 'eqpart' scheme. For comparison, the conventional contrast sum of marginal entropies (Φ^{hyi}) using KDE is also considered with SRICA in 'eqpart' scheme. The real world sources are taken from ICALAB Cichocki et al. (29).

The results show that SRICA with Φ^{hyi} fails for real world sources 8 and 10. But, SRICA with LSFD has comparable performance to other ICA algorithms, even best in one of the cases. The failure of FastICA for one of the test case is also noticeable.

Table 4.5: Performances of SRICA using varying independence measures against real world sources from ICALAB; Comparison with other ICA algorithms using API as the performance measure. GA number of function evaluations = number of search variables * 70

Source	nsrc	**srica($\sum hy_i$)**	**srica(LSFD)**	**FastICA**	**EFICA**	**NPICA**	**radical**
		eqpart	eqpart				
Speech4	4	0.0562	0.0167	0.0179	**0.0160**	0.0219	0.189
Speech8	8	***0.1858***	**0.0408**	***NaN***	0.0467	0.0592	0.0475
Speech10	10	***0.1800***	0.0687	0.0700	0.0734	**0.0499**	0.0639

The SRICA results in higher dimensions show misconvergence or near optimal solution. With the failure of other algorithms and even the Exhaustive Search based RADICAL, there emerges a need to go into more details of misconvergence and suitable choice of an LSGO algorithm. This inspires to look at the GA like, optimization techniques for large scale in two ways: (i) How the search for optimal progresses in GA? and (ii) What are the reasons leading to misconvergence in GA, specifically in large scale, and how the BSS contrasts behave as an optimization functions for an LSGO algorithm? The dual focus should be able to bring more theoretical knowledge and possible remedies to avoid misconvergence and to reduce computations for LSGO and specific LSnIBSS problems. The Chapter 5 focuses on the first aspect and then next Chapter 6 focuses on the second aspect.

4.8.1 Discussion On ICA as BSS

The experimental results show that there was no algorithm giving performance in a permissible range ($API < 0.1$), even in two source separation or higher dimensions. The failure may be due to the ideal optimization landscape violation by the sources for a given contrast or due to the failure of optimization algorithm. The performance analysis in two source separation and empirical local minima analysis has proved the presence of near-independence sources even in laboratory generated source cases. This atleast puts a caution on using ICA algorithm for BSS, even in linear case. Though use of ICA algorithm for BSS can not be denounced, for large scale and some specific type of distributions there question whether they should be considered equivalent. There were performed some experiment on adding priors like identical distributions or un-identical distributions, but further testing is needed before anything firmly could be stated. Also, a mathematical extrema analysis in higher dimensions is needed, for which the GGC in multivariate is developed, but due to time constraint could not be done.

4.9 conclusion

The SRICA algorithm has been best (based on mean performance) at being able to avoid local minima, even better than the exhaustive search based algorithm. It gave consistently good performance for sum of marginal entropy $\Phi(hyi)$ estimated through KDE based contrast. The algorithm has facilitated the study of the effects of various independence definitions and measures on ICA solution. The study has resulted declaring minimization of sum of marginal entropies, with kernel based PDF estimation method as the mean wise best independence measure, in terms of source matching, among the used other independence measure. There SRICA algorithm and the contrasts Φ_2^{FD} and Φ_2^{GFD} are verified. They have performance better for the multimodal densities but lack consistency. The failure of GA and SRICA leads to further two queries. The first is how GA succeeds in global search and how it could be improved. This query has been handled in next Chapter 5. The second query is LSGO algorithms perform on BSS problems and how that performance can be improved. This query has been handled in Chapter 6.

Chapter 5

Extended Forma Analysis and Mendelian Genetic Algorithm

There exists a long discussion over the issue of whether minimal alphabet or maximal alphabet gives maximum schemata. The chapter generalizes the concept of schemata to dependency relation based 'extended formae'. Further, it proves that theoretical maximum schemata could be achieved through an operator exploiting extended formae and using maximal alphabet. It shows that the previous conclusion of minimal alphabet giving maximum schemata is also true for some operators. It is believed that maximum schemata is advantageous to achieve maximum implicit parallelism. The chapter raises a discussion over the disadvantages of maximum schemata. As a conclusion, it suggests to use an intermediate-level alphabet for representation, balancing maximal alphabet to avail maximum schemata and minimal alphabet to overcome the disadvantages due to maximum schemata.

5.1 Introduction

Genetic Algorithm (GA) has been fairly successful as a huiristic search and optimization technique, in many practical engineering and design problems. Towards the success of GA, Holland made a powerful observation of progress in search through prorogation (inheritance) of similarities within chromosomal representation of the solution space. To signify the importance of similarities among the chromosomes, he gave mathematical notion to it deriving the concept of Schema and Schema Theorem.

The significance of the schema concept can be understood by the generalizations it has obtained. Towards the efforts to understand real coded GA (RCGA), the schema concept got generalized as Wright's schema by Wright (138), as interval-schema by Eshelman (47) and as virtual alphabet by Goldberg (52). Schema, as a way to define similarity among the set of chromosomes, got generalizations as forma induced by an equivalence relation by Radcliffe (99, 100), as a predicate by Vose (131) and also through work by Antonisse (4).

Though available many generalizations, the maximum schemata[*] has not achieved the theoretical maximum possible. Towards providing maximum schemata[†], this chapter contributes in the following ways:

1. It provides generalization of the schemata concept through further generalization of equivalence relation based formae definition to the dependency relation based extended formae definition. This makes it possible to achieve theoretical maximum possible schemata for both string and non-string representations, which would not have been possible through formae or other previous definitions of schemata.

2. It signifies that similarity has to be an operator perspective. To be able to exploit similarity information through extended forma definition -

   (a) it proposes the use of already existing ploidy representation and derives for it a novel ploidy schemata (p-schemata) definition. Thus, it adds the notion of past similarities to define schemata.

   (b) it derives a specific Mendelian Crossover Operator to exploit the similarity information through p-schemata.

3. It proves that not the minimal but the maximal alphabet gives maximum schemata, through extended formae definition. It also proves that extended formae could achieve theoretical maximum schemata.

4. It provides discussions on the relevance of previous concept of minimal alphabet giving maximum schemata (49, 51, 52, 59) and the need of maximum schemata for efficient GA. Based on the discussions and inspiration from nature it provides a conclusion.

The next section 5.3, defines the extended forma induced by an arbitrary dependency relation. It compares extended forma with the existing schema definitions, in terms of the definition and available maximum schemata. Section 5.4, signifies the role of operator by concluding that the schemata has to be an operators perspective. Towards, achieving an operator exploiting extended formae it proposes a ploidy representation and a Mendelian crossover operator in section 5.5. Then

[*] Schemata - is a plural of Schema

[†] To avoid confusion, this chapter uses ‘schemata’ nomenclature as a more generalized form for similarity subsets through any existing (original Holland’s schemata, o-schemata, formae, predicates etc.) or futuristic representation and definition; and ‘Holland’s schemata’ as that specifically defined by Holland.

after, section 5.6 derives the important theorems and interpretation to achieve maximum schemata. Section 5.7 provides discussion on the requirement of maximum schemata and then finally the section 5.10 provides conclusion.

5.2 Concept of Schema and the Motivation for Further Generalization

Let ρ be a mapping of the n-point discrete search space $\mathcal{S}$ onto the space of chromosomes $\mathcal{C}$.

$$\rho : \mathcal{S} \longrightarrow \mathcal{C}$$

Let the chromosomes be a k-ary string of length l; set $\mathcal{A}$ be the set of all k symbols of k-ary alphabet, $\mathcal{A} = \{0, 1, \ldots, k{-}1\}$, $\mathcal{P}(\mathcal{A})$ be the power set of $\mathcal{A}$ and $\Omega = \mathcal{P}(\mathcal{A}) - \{$set of all subsets of $\mathcal{P}(\mathcal{A})$ with cardinality 0 and 1$\}$.

There are different ways possible to define the similarity in $\mathcal{C}$.

Holland's Schemata

Holland (59, 60) defined schema, S as a similarity template describing a subset of k-ary strings with fixed values, say either 0 or 1 for $k = 2$, at certain positions and 'don't care' (#) at the other positions. For example, a Holland's schema $1\#00$ implies a subset $\{1000, 1100\}$.

Generalization of schemata as an equivalence relation based Formae

Radcliffe (99, 100) defined similarity through an arbitrary equivalence relation on $\mathcal{C}$ and identified it as the Forma. That generalized the concept of schemata to be applicable for string and non-string structures.

The similarity among a subset of chromosomes in $\mathcal{C}$ can be represented by a string of symbols ■ and □, indicating 'fixed' or 'matching' values at ■ symbol positions and 'don't care' at the □ symbol positions. There each symbol in the string was called a *component* and the string containing symbols may be named as a *relational string*. More precisely, a relation $\sim \in \Psi$ be defined for $\eta, \zeta \in \mathcal{C}$

$$\eta \sim \zeta \Leftrightarrow (\forall i \in \mathbb{Z}_l; (\sim_i = \blacksquare) : \eta_i = \zeta_i) \tag{5.1}$$

where, $\mathbb{Z}_l = \{1, 2, \ldots, l\}$; $\sim_i$ indicates ith component of the relational string. The properties of $=$, defines $\sim$ as an equivalence relation, satisfying the properties of symmetry, reflexivity and transitivity. An equivalence relation string may be written for example as ■□□■. A specific fixed value at ■ symbol, in a given equivalence relation string will induce equivalence classes, called formae. The formae, induced by a particular equivalence relation, will divide $\mathcal{C}$ into a disjoint partition.

Theoretical maximum possible schemata and motivation for further generalization of schemata

The maximum possible correlated subsets among the n-point search region $\mathcal{S}$ (NS_{max}) would give an upper bound on the theoretical maximum possible schemata in $\mathcal{C}$ (NC_{max}).

$$NS_{max} = \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{n} = 2^n - 1 \tag{5.2}$$

If ρ is a one-one mapping, $NC_{max} = NS_{max}$. Without loss of generality, this is the case assumed for rest of the chapter. In case of, an under-specified or over-specified representation $NC_{max} = \alpha NS_{max}$, where $\alpha = |\mathcal{C}|/|\mathcal{S}|$ and $|\cdot|$ indicates cardinality of a set.

It could be observed that the theoretical maximum possible $2^n - 1$ subsets of $\mathcal{C}$ (NC_{max}) are not disjoint. Many of theses subsets share some common elements. This implies that forma definition can not achieve all correlated subsets in $\mathcal{S}$ as schemata in $\mathcal{C}$ i.e. $NC_{max} \neq NS_{max}$. By discarding the transitivity condition, an equivalence relation gets generalized to dependency relation (finite tolerance relation)[‡]. This will allow dependency relation based extended formae to define similarity subsets also for those schemata in $\mathcal{C}$, which are not available through forma definition. As has been proved later in Theorem 5.4, section 5.6 that this will achieve for extended formae $NC_{max} = NS_{max}$.

5.3 Proposed Dependency Relations based Extended Formae

Towards the goal of schema definition, the similarity in $\mathcal{C}$ can be obtained defining an arbitrary dependency relation (finite toletrence relation). Let,

$$\Psi = \{\blacksquare, \square\}^l$$

is the space of all dependency relations on $\mathcal{C}$. The related chromosome strings are having 'fixed' or 'matching' values from $\mathcal{A}$, at the ■ symbol positions and are having either of the elements of the specific subset $\square_{i..j}^{(p)}$ at symbol □ positions, where $\square_{i..j}^{(p)} \in \Omega$. So, the □ could be interpreted as 'either or' symbol reminding selection of either of the element of Ω as the specific subset. The specific subset $\square_{i..j}^{(p)}$ could be interpreted as 'p-level either or'; where $p = |\square|$, $|\cdot|$ indicates

[‡] Tolerance relation is a relation which is symmetric and reflexive but not necessarily transitive. Dependency relation is a finite tolerance relation.

cardinality of a set, $1 < p \leq k$ aand the suffix $i..j$ is the list of all elements in that subset. For ease, $\square^{(k)}$ may be used and the suffix may be avoided to indicate 'k-level either or' or 'don't care'. For the same reason of ease, the cardinality representation as power may be avoided. Given, say $l = 4$, a relation $\sim \in \Psi$, $(\blacksquare, \square, \blacksquare, \square)$ could be represented as the relational string $\blacksquare\square\blacksquare\square$ and $\sim_i$ indicates ith component of the relational string.

More precisely, a relation $\sim \in \Psi$ be defined for $\eta, \zeta \in \mathcal{C}$ and $\mathbb{Z}_l = \{1, 2, \ldots, l\}$

$$\eta \sim \zeta \Leftrightarrow \Big(\forall i \in \mathbb{Z}_l; (\sim_i = \blacksquare) : \eta_i = \zeta_i, (\sim_i = \square) : \eta_i, \zeta_i \in \square^{(p)}_{i..j}\ \square^{(p)}_{i..j} \in \Omega\Big) \tag{5.3}$$

As all the members of Ω are not disjoint, based on the definition and properties of $=$ and $\in$ operators, the relation $\sim$ satisfies the properties of symmetry and reflexivity but not transitivity. Also, due to finite chromosome space assumed, it is a dependency relation or a finite tolerance relation.

Fixing the value of $\blacksquare$ and $\square$ at the given positions, induces the similarity subsets (schemata) for each dependency relation in Ψ. As dependency relations are generalization of equivalence relations and schemata induced by equivalence relations are called formae, the schemata induced by dependency relations could be named 'extended formae'.

The comparision of equation 5.3 with equation 5.1 and the used relational strings makes clear the difference between formae and extended formae. As the first step, let equivalence relation be defined as $\blacksquare\square^{(3)}\square^{(3)}\blacksquare$ for $k = 3, \mathcal{A} = \{0, 1, 2\}$ alphabet. Fixing the values for $\blacksquare$ symbols, will induce a disjoint subset of a partition. Let, $1\square^{(3)}\square^{(3)}2$ is one of the disjoint subset of a partition. Now, as the second step, let $\square$ be interpreted as 'either or' interpretation taking different possible values from Ω. Extending the example, a relational string $1\square_{12}\square_{01}2$ and $1\square_{01}\square_{12}2$ are under consideration. It could be observed that this is like dividing the previous disjoint subset of a partition further into correlated subsets. This further division is correlated as having a non-empty intersection. The common string between the considered two correlated subsets would be $\{1112\}$.

This makes clear the relation between the formae and extended formae, and also the nomenclature 'extended formae'.

For example: k=2 (a binary alphabet), then $\mathcal{P}(\mathcal{A}) = \{\{\}, \{0\}, \{1\}, \{0, 1\}\}$ and $\Omega = \{\{0, 1\}\}$. Accordingly, $\square = \{0, 1\}$ is the only possibility. An extended forma $1\square00 = \{1000, 1100\}$

Similarly, for k=4, a 4-ary alphabet with $\mathcal{A} = \{0, 1, 2, 3\}$, corresponding
$\Omega = \{\{0, 1\}, \{1, 2\}, \{2, 3\}, \{3, 0\}, \{0, 2\}, \{1, 3\}, \{0, 1, 2\}, \{1, 2, 3\}, \{2, 3, 0\}, \{3, 0, 1\},$
$\{0, 1, 2, 3\}\}$, Symbolically, $\Omega = \{\square_{01}, \square_{12}, \square_{23}, \square_{30}, \square_{02}, \square_{13}, \square_{012}, \square_{123}, \square_{230}, \square_{301},$
$\square_{0123}\}$. $\square$ could be any of these 11 sets. An extended formae $20\square_{12}1 = \{2011, 2021\}$, where $\square_{12} = \{1, 2\}$.

Antonisse work (4) and comparision

It must be acknowledged that the chapter (4) by Antonisse suggested almost similar generalization of Holland's schemata. The chapter interpreted the symbol $\#$ in the Holland's Schema as any subset from Ω. But, the dependency relation was not used there for schemata definition. The proposed arbitrary dependency relation based extended forma definition is applicable to both string and non-string structures. Also, that chapter (4) did not tried to answer from where or how these extra schemata will be available to GA. There it was almost neglected that the similarity has to be an operator perspective. Rather, in the concluding remark it hoped to achieve the extra schemata with crossover operator being intact. Compare to this, the present work, identifies the significance of an operator (section 5.4) and derives an operator (section 5.5) exploiting the extra schemata available through dependency relation.

5.4 Schemata and Operators

GA has to find the optimal and generally, is blind i.e. there is an absence of any information about the correlated regions in the fitness landscape. The goal of schema definition is to get subsets in the set of chromosomes, with some similarities (schemata) based on two assumptions:

1. The similarities in the set of chromosomes (representational landscape) gives possible information about the correlations in the actual fitness landscape. To validate this assumption for a wide range of fitness landscapes, the notion of schemata definition has to be based upon the most common search principles of searching through say, locality, periodicity etc. For a specific fitness landscape there could be thought upon specific notions of similarity.

2. There exists a GA operator, using this information about the similarities or dissimilarities in the set of chromosomes to direct the next generation search towards the corresponding most correlated or uncorrelated regions.

The success of GA or the used schemata definition and a corresponding representation depends upon the degree by which both the assumptions are satisfied.

Holland interpreted the used symbol $\#$ as a 'wild-card' or a 'don't care'. But, the then existing crossover operators, were interpreting the symbol as '2-level either or', as they select a specific value from either of the two parents. The assumed interpretation of 'don't care' and the actual interpretation of '2-level either or' by a crossover operator - are same only in case of binary. The difference between the interpretations is important as it will create difference in the number of schemata possible. If assumed the conventional single-point, multi-point or uniform crossover then the maximum possible schemata per position (NCP_{max}) for k-ary would be, $NCP_{max} = \binom{k}{1} + \binom{k}{2}$;

which is more than $k + 1$ schemata for all $k > 2$. An example of an operator, which interprets Holland's schemata as he defined i.e. 'don't care' or a 'wild card' interpretation - is the Random Respectful Recombination (R^3) Operator by Radcliffe (100). So, for this operator NCP_{max} with k-ary symbol would be, $NCP_{max} = k + 1$, as expected by Holland.

As mention earlier, Antoisse expressed schemata in a way similar to extended formae. But, then existing conventional crossover operator is an operation between two parents. So, these operators would not be able to interpret schema position with $\#$ as any subset with cardinality $|\#| > 2$ from a power set $\mathcal{P}(\mathcal{A})$. Say, for $k = 4$, a schema $20\#_{013}$ indicates the set of strings with value at not matching positions to be '3-level either or' from the subset $\{0, 1, 3\}$. But, the available single-point, multi-point or uniform crossover can not operate on this interpretation.

So, this section concludes that for a successful GA, similarity has to be an operator perspective or schemata should be the perception of an operator. Accordingly, there must be designed a GA operator exploiting similarity information through extended formae.

5.5 GA Operator - Exploiting Extended Formae

Achieving 'p-level either or' with $2 < p < k$ interpretation for $\Box_{i..j}^{(p)}$ necessitates two things.

1. *Representation Perspective*: There has to be available at least p allele values at a position.

2. *Operator Perspective*: There has to be an operator implementing 'p-level either or'.

The requirements lead to at least two solutions. One of the solutions is to use a multi-parent (p parents) recombination operator. The other solution is to use a ploidy (multiplicity of chromosomes) representation with atleat $p/2$ copies of chromosomes for both parents with an operator working on this representation. The operator with multiple parents and ploidy representation for each parent - both together; and many others may be thought. But, it is better to use a solution approved by nature.

5.5.1 Inspiration from Nature for Representation Perspective

It is known that in nature the most complex organisms are diploid or polyploid. Humans, most animals and many plants are diploid (pairs of chromosomes). Polyploidy is also existing in few plants and animals. The ploidy structure makes it possible to store and inherit the past genetic history in terms of the unexpressed. It is one of the sources of wide diversity in nature.

Also, the ploidy structures, specifically the diploidy are not new to the GA community and have already proved their significance for non-stationary or time-varying applications (51, 119).

Also, the multi-parent recombination operators (44) have been found useful in some applications, but are not that popular compare to ploidy structures.

Finally, there has been decided to follow nature and to use ploidy representation and corresponding operator than a multi-parent haploidy, for the further development of the topic.

5.5.2 Deriving p-schemata for Ploidy Representations

Conventionally, schemata definition is applied to get similarities among the parents or chromosomes from different individuals. In case of ploidy representations, there exists two way similarities. The first one is, within an individual among the multiple copies of chromosomes i.e. intra-parent similarity. The other one is, similarity between (among) the parental ploidy representations - inter-parent similarity. It could be observed that the intra-parent similarities and the inter-parent similarities - both could be obtained by extended forma definition.

For example, let $k = 4$, $\mathcal{A} = \{0, 1, 2, 3\}$ and ploidy structurewith four copies of chromosomes per parent is in use. Let, the parent P_1 be with chromosomes 0123, 1103, 2133, 2103 and the parent P_2 be with copies 3120, 2130, 0120, 0130. The intra-parent extended forma for parent P_1 could be written as $\square_{012}1\square_{023}3$ and that for parent P_2 could be written as $\square_{023}1\square_{23}0$. Then, inter-parent extended forma for parent P_1 and P_2 could be written as $\square_{0123}1\square_{023}\square_{0,3}$.

It is possible to give importance to the multiplicity of alleles and precedence of the chromosomes in terms of the generation in which they were expressed. If the operator needs to use these information, then instead of the used conventional set structure for $\square$; there has to be used ordered multiset [§] structure.

Overall, the application of the extended forma definition to ploidy representation for similarity subsets has been achieved. Let both the intra-parent and inter-parent schemata be given a common name 'p-schemata', where p reminds us the term ploidy. To be more precise, if required, the intra-parent schemata may specifically be named as 'p^1-schemata' and the inter-parent schemata may specifically be named as 'p^2-schemata'.

The introduction to p-schemata, introduces the notion of either past similarities or past inheritance or past successful changes in allele values or past, in general - as the search strategy to define schemata. GA using diploidy representations are existing. That means the past or the unexpressed genetic information was already in use for search in GA. But, the notion of past with the already existing notion of search through locality, periodicity etc. for schemata definition is new and may find useful.

[§] Multiset is a generalization of set to allow repeatation of elements.

5.5.3 Inspiration from Nature for Operator Perspective

Towards the goal of deriving an operator exploiting extended formae, the first necessity of availing more than p alleles at a gene position has been achieved through deriving ploidy representation and p-schemata for it. The next step is to derive an operator implementing 'p-level either or'.

Organisms with ploidy, follow largely two major inheritance mechanisms. First, the Mendelian inheritance and second, the non-Mendelian inheritance. Accordingly, there could be suggested crossover operators matching Mendelian Inheritance or Non-Mendelian Inheritance using p-schemata.

Mendelian Inheritance

Though basically defined for diplody, can be made applicable to all ploidy. There are two basic principles of Mendelian inheritance, described here in the way they are understood now.

1. *Law of Segregation* : It says that during gameteeogenesis¶ through meiosis, in both the parents, the sets of chromosomes is segregated and gametes have only half of the total chromosomes. During fertilization, any of the gamete is randomly selected and without any exchange of genetic material just passed on to the embryo. The embryo will have full sets of chromosomes, adding half from each gamete. Generalizing, the inheritance does not depend upon the allele at a gene position being expressed or unexpressed. The dominancy decides what is expressed and does not affect the probability of what is inherited.

2. *Law of Independent Assortment or Inheritance Law*: This law states that all genes are passed on independent of each other to the gametes during gameteeogenesis. This says that no two genes are having linkage to get inherited always together. Practically, crossover point could be anywhere in the string of genes.

Non-Mendelian Inheritance

Non-Mendelian inheritance is a collective nomenclature for inheritance mechanisms not following laws of Mendelian Inheritance. Say, importance of dominancy to the inheritance, assuming unequal probabilities for crossover points etc. could be considered Non-Mendelian.

5.5.4 Mendelian Crossover Operator Exploiting Extended Formae

Just to give example of an operator exploiting extended formae there is described a Mendelian Crossover Operator assuming diploidy and following Mendelian inheritance. The operator is three step:

¶The process of gamete formation, before fertilization

1. During gameteeogenesis through meiosis, there will be a crossover between the two copies of chromosomes in each parent. In humans, there are created 4 haploid gametes through separate crossover for each. For ease of operation and representation, let the simulated operator be using the expressed (i.e. dominant) allele combination and the unexpressed (i.e. dormnat) allele combination as two chromosomes in a parent. Also, two crossover operations would be sufficient - one, modifying the expressed and the other, modifying the unexpressed[‖].

2. During the fertilization step a child gets one copy of chromosome from a gamete from each parent randomly.

3. Then, the dominance mechanism decides the alleles which are expressed and unexpressed. In case of no knowledge about dominant alleles, there could be assumed random dominancy mechanism i.e. any allele at a gene position is equally likely to get expressed. This could be implemented using a one more crossover operator application on the available two haploids with the child.

The above operator requires three crossovers (two during gameteeogenesis and one for dependency). It could be assumed that the gene locations affected by the crossover during gameteeogenesis are the same as those affected by the crossover during dominancy decision. This assumption, can bring the same overall result through a single crossover between randomly selected one chromosome from each parent. This has been shown in table 5.1. Other than computational reductions, use of single crossover will also bring reduction in the population diversity. The table, specifically the last column entries for equivalent single crossover among the selected chromosomes from parents, reminds the outcomes of Mendel's experiments in terms of the probabilities of inheritance of the dominant and dormant alleles.

5.6 Whether Small Alphabet or Large Alphabet?

Till now, dependency relation based extended formae and a Mendelian operator exploiting that has been derived. So, it will be interesting to re-look at the issue of whether small alphabet or large alphabet to get maximum schemata, based on this new schemata definition.

Theorem 5.1. *An alphabet of cardinality k will induce $n_s = (2^k - 1)^{1/\log_2 k}$ schemata or similarity subsets per bit of information, based on extended formae definition.*

[‖] The conventional crossover will keep the most significant gene position unaffected. So, the modified chromosome is the one with the most significant gene position allele intact.

Table 5.1: Mendelian Crossover Operator

Parent p_1 Parent p_2	Expressed Chromosome (p1e): A1B1; Unexpressed Chromosome (p1u): a1b1 Expressed Chromosome (p2e): A2B2; Unexpressed Chromosome (p2u): a2b2
Gametogenesis	

Parent p_1
Parent p_2
	gamete 1 (g11) : A1b1; gamete 2 (g12) : a1B1;
	gamete 1 (g21) : A2b2; gamete 2 (g22) : a2B2;

Possible children after fertilization and dominancy mechanism				
Child	Gametes Selected for fertilization	Expressed Chromosome	Unexpressed Chromosome	Equivalent Parent Chromosomes for single crossover
c_1	g11-g21	A1b2	A2b1	p1e-p2u
c_2	g11-g22	A1B2	a2b1	p1e-p2e
c_3	g12-g21	a1b2	a2B1	p1u-p2u
c_4	g12-g22	a1B2	a2B1	p1u-p2e
c_5	g21-g11	A2b1	A1b2	p2e-p1u
c_6	g21-g22	A2B1	a1b2	p2e-p1e
c_7	g22-g21	a2b1	A1B2	p2u-p1u
c_8	g22-g22	a2B1	a1B2	p2u-p1e

Proof. Let s be the set of all k symbols and S_p be the set of all schemata per position, through extended formae definition. Then, the maximum schemata per position,

$$NCP_{max} = |S_p| = |\mathcal{P}(s) - \{\}| = \binom{k}{1} + \binom{k}{2} + \ldots + \binom{k}{k} = 2^k - 1$$

Each position represents $\log_2 k$ bits.
So,

$$n_s = |S_p|^{1/\log_2 k} = (2^k - 1)^{1/\log_2 k} \quad \square$$

□

Corollary 5.2. *The set of extended formae as schemata for k-ary alphabet contains all extended formae (schemata) induced by all m-ary alphabets, $m < k$.*

Proof. Let $\mathcal{A}_k$ be the set of all k symbols and S_k be the set of all schemata induced by k-ary alphabet. Let $\mathcal{P}(\mathcal{A}_k)$ be the power set of $\mathcal{A}_k$. Similarly, let $\mathcal{A}_m$ be the set of all m symbols and S_m be the set of all schemata induced by m-ary alphabet. Let $\mathcal{P}(\mathcal{A}_m)$ be the power set of $\mathcal{A}_m$.
Given $m < k$, $\mathcal{A}_m \subset \mathcal{A}_k \Rightarrow \mathcal{P}(\mathcal{A}_m) \subset \mathcal{P}(\mathcal{A}_k)$
As schema position takes any value from the set $\{\mathcal{P}(\cdot) - \{\}\}$, $S_m \subset S_k$. □ □

Theorem 5.3. *For a given information content, strings coded with larger alphabets give more extended formae as schemata per bit of information than that coded with smaller alphabets.*

Proof. From Theorem 5.1,

$$n_s = (2^k - 1)^{1/\log_2 k}$$

We are interested in the rate of change of n_s with respect to k. Then,

$$\log_2 n_s = \frac{\log_2(2^k - 1)}{\log_2 k}$$

$$\Rightarrow \frac{1}{n_s}\frac{dn_s}{dk} = \frac{1}{\log_2 k}\left[\frac{2^k}{2^k - 1} - \frac{1}{(\ln 2)^2}\frac{\log_2(2^k - 1)}{k}\right] = \frac{1}{\log_2 k}\left[p\left(k\right) - q\left(k\right)\right]$$

where, $p\left(k\right) = \frac{2^k}{2^k - 1}$ and $q\left(k\right) = \frac{1}{(\ln 2)^2}\frac{\log_2(2^k-1)}{k}$. As $p\left(k\right) > 1$ and $q\left(k\right) < 1$, the derivative is always positive, n_s is monotonically increasing function. This proves that for a given information content, larger alphabets give more extended formae as schemata than that coded with smaller alphabet. □ □

Theorem 5.4. *For an n-point search space $\mathcal{S}$, maximum schemata induction requires n-ary alphabet and schemata as extended formae.*

Proof. Without loss of generality, let us assume a one-one mapping $\rho : \mathcal{S} \longrightarrow \mathcal{C}$. Given n-point search grid, the length of binary coded chromosome string will be $\log_2 n$. Let also assume that a k-ary alphabet gives maximum schemata. As already discussed in section 5.2 and the calculation of maximum schemata per position (NCP_{max}) in section 5.4; the maximum possible schemata in $\mathcal{C}$ is

$$NC_{max} = NS_{max} = 2^n - 1$$

and neither the Holland's schemata nor the formae can achieve these maximum schemata.

Now, from Theorem 5.1, total extended formae as schemata per bit of information are $n_s = \left(2^k - 1\right)^{1/\log_2 k}$. Then, the total extended formae (schemata) for the n-point search grid using k-ary alphabet (N_k),

$$\begin{aligned} N_k &= \left(\left(2^k - 1\right)^{(1/\log_2 k)}\right)^{(\log_2 n)} \\ N_k &= NC_{max}, \text{iff } k = n \end{aligned}$$

□ □

5.7 Discussion and Interpretations

The old result had stated that minimal alphabet gives maximum schemata (49, 51, 59) and the current chapter proved that maximal alphabet could give maximum schemata, same as maximum correlated search points are available. This raises a question, whether the previous conclusion was correct or the current conclusion? The answer is, both the conclusions - the minimal alphabet giving maximum schemata and the maximal alphabet giving maximum schemata - are correct, but both with respect to separate group of operators. The R^3 like operators, interpreting $\Box$ as ‘don’t care’ will give maximum schemata for minimal alphabet. The uniform crossover operator interpreting $\Box$ as ‘2-level either or’ will give maximum schemata for maximal alphabet, but available schemata will not be same as the theoretical maximum possible. It is the combination of ploidy representation and the derived operator 5.5.4, interpreting $\Box$ as ‘p-level either or’ with $2 \leq p \leq k$ that will provide theoretical maximum possible schemata.

Also, it is important to ask whether successful GA requires maximum schemata? It is true that implicit parallelism increases with the increased schemata. This may suggests to maximize the schemata. The schema generalizations have shown that other than the traditional Holland’s schemata, these generalized schemata may also contribute to the implicit parallelism. Radcliffe (99) emphasized that only the schemata with correlated performances are significant and not the total number. May be, in a blind case it could be assumed that maximization of schemata gives maximization of correlated schemata i.e. with least fitness variance schemata.

But, can a question be raised, whether maximal schemata have any disadvantage? The previous Theorem 5.4 gives us a hint. It says that maximum schemata for an n-point search space could be available through n-ary alphabet. That means, single position schema would be required. This is correct intuitively also, as more the chromosome string length - the ‘Hamming Cliff’ [**] causes more schemata reprsenting correlated nearby search points to be missed. But, at the same time single digit schema implies search through just locality principle; and periodicity principle is not in use for search. The total schemata is more but all based on the principle of similarity through locality. In general, it could be observed that larger the alphabet, larger the locality region and smaller the periodicity (number of repeatations).

For a succesful GA, based on the application and search stage, both the principles of searching through locality and searching through periodicity may be given varying significances. Say for example, as the search progresses - the locality search regions should become smaller to make the search finer (exploitative search) and at the same time the periodicity (number of repeatations) should be increased to look for the alternative search regions so that the search is not confined to a local region (exploratory search). Similarly, the initial search will be benefited by increased implicit parallelism and progress in search through locality may be sufficient enough.

This interprets that large alphabet may serve better in the initial stages. But, it may not be that good in the latter generations. So, for latter generations minimal alphabet providing search through more periodicity and smaller local width is a better options. Usually, it is known that

[**] Hamming Cliff is the effect of the used representation causing points nearer in the actual search space to be far in the representation space. Say, 7 and 8 are the given search points nearer to each other. But when represented using binary coding, they are mapped as 0111 and 1000 and are at the farthest hamming distance.

GA, if property initialized, will search out the regions with above average fitness in very few initial generations, irrespective of used initial representation. So, it may be sufficient to use a single representation throughout. The used representation should be a balance between - maximal alphabet giving maximum schemata and a minimal alphabet providing better search regions i.e. providing smaller locality width and more periodicity for further search.

Nature also supports this argument. The genetic information is coded in terms of four nucleobases. The four DNA-bases[††] are cytosine (C), guanine (G), adenine (A) and thymine (T) and four RNA-bases[‡‡] are A, G, C and Uracil (U). Thus, nature balances the small alphabet and large alphabet using 4-ary and diploidy representation.

5.8 The Mendelian Genetic Algorithm (MGA)

The MGA uses diploidy (pair of chromosomes per individual) representation and Mendelian inheritance through the Mendelian crossover operator derived in the previous subsection. The conventional selection and mutation techniques are extended for MGA also. Important is the initialization techniques. As with double the initial chromosomes, computational cost for initialization may increase. But, there are suggested initialization schemes with same computations as for haploid representation. There are also derived various child selection schemes. The schemes define how many children, what way the expressed and unexpressed chromosomes of a child are generated and have different results on diversity and convergence of the generated population. Based on whether at a given stage exploitation or exploration is required, a specific child selection scheme can be used. Overall, MGA can have same computational complexity as for conventional GA. But, the abundance of schema is inferred to avoid loss of genetic material after higher generations. The experimental analysis proved low rate of convergence during initial stages, and again misconvergence at higher generations, though at a better value than conventional GA. This demands intelligent use of parameter values, balancing the exploration and exploitation ability of MGA. The demand is fulfilled providing convergence analysis and deriving adaptation rules based on it.

5.9 Convergence Analysis of MGA

The goal is to derive relations between the progress in fitness value and the free parameters, specifically the degree of importance given to the past generation. More significance to the past information imply more diversity at the cost of fitness improvement. The analysis follows the similar for

[††] Nucleobses for Deoxyribo-nucleic acid

[‡‡] Nucleobses for Ribo-nucleic acid

GA in (97, 98) and assumes infinite population in continuous space. The analysis brings adaptive MGA (aMGA).

5.10 Conclusion

The chapter generalizes the concept of schemata to dependency relation based extended formae. The generalization makes it possible to achieve theoretical maximum possible schemata for both the conventional string and non-string representations. The chapter also derives an operator exploiting extended formae. For that it uses the ploidy representation and derives p-schemata definition for it. This brings the notion of ‘past’ information with existing locality, periodicity and other notions for schemata definition. It proves that whether minimal alphabet gives maximum schemata or maximal alphabet gives maximum schemata is decided by the operator used, as schemata has to be exploited by an operator. Also, successful GA do not just require maximum schemata but a balance between the available maximum schemata for intrinsic parallelism and required similarity regions in terms of locality and periodicity for further progress. The nature suggests to use 4-ary, diploidy representation.

Chapter 6

Optimization Issues in Large Scale and BSS

The Chapter starts with the results of BSS in higher dimensions through SRICA (using GA) and other BSS algorithms. Specifically, the failure of SRICA (using GA) bring two requirements: 1) How to avoid misconvergence of GA in higher dimensions 2) How the BSS contrasts behave as optimization functions. The Chapter examines both the view points and provides furher research directions.

6.1 Introduction

There exists many bio-inspired search and optimization algorithms imitating natural phenomena or a behaviour of some biological species. The Evolutionary Algorithms (EA) are the subset of this bio-inspired algorithms. Instead of a single search trajectory, they have multiple search trajectories formed due to repeated applications of operators on the randomly selected initial population (of possible solutions). There are three basic operators named selection, recombination (crossover) and mutation as already explained in the previous Chapter 4, Section 4.7.2. The EAs may differ in using some or withdrawing some of the operators from them. For example, the Evolution Strategies (ES) more depend upon the mutation operator and so the recombination operator is either completely missing or used with small probabilities. Compare to them, GA more depends on the recombination operator. In general, the better search algorithms should have two types of abilities: 1) *Exploration*: The algorithm should not get stuck to the better regions found by previous generations. Instead, it should be able to explore in the new regions. This ability of an algorithm corresponds to the global search ability. 2) *Exploitation*: Once found the possible better fitness regions (niche) from the search space, the algorithm should be able to exploit that acquired knowledge in terms of fine tuning the search. This ability correspondce to a local search ability. The performance of an algorithm depends upon the balanced utilization of the abilities of exploration and exploitation.

The GA is one of the EAs imitating natural evolution process, where 'Survivle of the fittest' and 'Natural selection' are the guiding principles. In a successive generations, how GA achieves better population is explained by the concept of Schema and schema Theorem (51, 60). There exists many generalizations of schema theorem that makes it applicable to almost all EAs with slight variations. The schema, as defined by Holland, is a similarity template describing a subset of binary strings with fixed value either 0 or 1 at certain positions and don't care at the other positions. The Schema Theorem, the fundamental theorem of Genetic Algorithm (GA) states that the short, low-order, above average fitness schemata (plural of schema), also called the building blocks (BBs), receive exponentially increasing trials in subsequent generations. Thus, the matching schema between the parents is seen as a genetic material being propogeted (inherited) to the next generation. In terms of a search strategy, schema gives direction for the next generation search in the intervals correlated through the principles of locality and periodicity (99). As each chromosome could be considered a sample from multiple schemata (intervals), a fitness calculation for a chromosome implicitely contibutes towards the knowledgw about the mean fitness of multiple schemata (intervals). So, an alphabet providing maximum schemata better fascilitates implicit parallelism. The understanding has brought some design priciples for GA (51) guiding on representation and operators issues. Ingeneral, all EAs face the 'curse of dimensionality' that demands exponential increase in the amount of computtaion with the increase in simaltaneus search dimensions. This implies with increase in computation the convrgence is assured. But, there are situations when increased computations also do not assure convergence.

6.2 GA Misconvergence and BSS Contrasts

The misconvergence in GA coud be grouped due to the following reasons.

- *Schema Deception*: This is due to representation, specifically in binary coded GA (BCGA). The GA search progresses in correlated strings. But, binary representation many times makes the near by regions coded in completely uncorrelated strings. This is identified as Hamming cliff, for example the points 7 and 8 coded in binary would be 0111 and 1000. Thus, though in the actual optimization landscap they are nearer, they are too far in the representational landscap. The problem and the possible solutions can be studied in more detail in articles (50, 57, 82) and many others. Overall, this is a representation issue and can be solved using float representtaion or varying representations in consecutive generations.

- *Domino Convergence and Genetic Drift*: This misconvergence phenomena has its root in the unequal time taken in the convergence of different bits in a long binary solution string. For example, it is possible that before the best fitness region (niche) is found the lower side bits, indicating higher precision, may have converged to a specific value in the whole population. So, the population do not have the genetic material at the lower sides to generate themost optimal solution. Thus, the loss of genetic material in the population in higher generations causes misconvergence. More details of this problemcan be found in (127) and the possible solutions in (78).

- *Non-separable functions or Genetic Linkage problem* The optimal of the search variables may be linked with each other. That means the best of one variable, varies with the value of the other. The only possible solution is to search both of them together. The problem is explained in more detail in the next subsection here.

SRICA was either failed to converge to an optimal or required too large computations. The performance of an optimization method depends upon the balanced utilization of the abilities of exploration (ability to explore in the new regions or the global search) and exploitation (using the acquired knowledge for the betterment of search or the local search). With the aim to reduce the computations for high-dimensional ICA, keeping in mind the balance of exploration and exploitation abilities for better search there were done many efforts. The efforts were done chronologically towards using other Evolutionary algorithms (EA), using different representations in GA, setting parameter values in GA, adaptive GA. Most of the experiments and conclusions were already noted in the history of GA. Some of the new concepts were also tested and succeeded partially. Important among them are Co-operative Convolution GA (CCGA), delta search and the concept of gradual search. The gradual search implies varying the representations through the progress in search. Overall, a technique combining CCGA framework with gradual search concept and fixed adaption of parameters was the best proved. But, still the problem of misconvergence was not solved.

The understanding of the reasons of misconvergence required a lots of literature survey from GA and EA. There were identified three different classes of problems leading to misconvergence of GA. Schema deceptive problems are based on the representation of solutions in GA and is more GA specific. The misconvergence due to the phenomena of 'Domino convergence' and 'Genetic drift' is also to a specific class of problems in GA. The misconvergence due to non-separability is the common problem where all optimization techniques fail. The reasons are discussed in (107). The same article concludes that the algorithm complexity of GA for an n-dimensional separable function is $O(n \ln n)$ but the same for a non-separable function optimization is n^n or $exp(n \ln n)$. The definitions of separable functions and non-separable functions are as under (107).

Definition 6.1 (Separable (decomposable) function)**.** A function f(x) is separable (decomposable) iff

$$\arg \min_{(x_1,,x_n)} f(x_1,,x_n) = (\arg \min_{(x_1)} f(x_1,,x_n), \arg \min_{(x_2)} f(x_1,,x_n), ..., \arg \min_{(x_n)} f(x_1,,x_n))$$

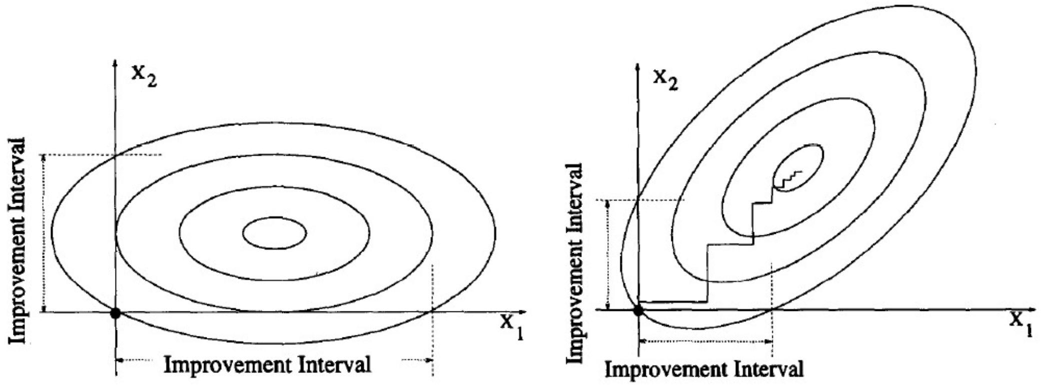

Fig. 1. Left: a quadratic function which is aligned with the coordinate system has a relatively large improvement interval. Right: a rotation of the same function leads to much smaller improvement intervals. The difference is increasing as the eccentricity is increasing.

In other words, a function of n variables is separable if it can be rewritten as a sum of n functions of just one variable. If a function f(x) is separable, its parameters x_is are called independent.

Functions which are not separable are called non-separable.

Definition 6.2 (nonSeparable (nondecomposable) function)**.** A nonseparable function f(x) is called m-nonseparable function if at most m of its parameters x_i are not independent. A non-separable function f(x) is called fully-nonseparable function if any two of its parameters x_i are not independent.

The following figures taken from (107) explain the reason what makes the non-separable function optimization a difficult problem. There was taken a random rotation of a separable function to simulate a non-separable function. The figure 6.2 explains the performance loss for unimodal functions, as a reduction of effective length, due to function non-aligned with the co-ordinates. Similarly, the figure 6.2 explains the loss of global optima for multimodal functions and requirement of simultaneous change in both the variables. The actual loss of performance in large scale global optimization is much higher because of the increased dimensions and varying essentricity throughout the search space. As a result, large scale optimization for non-separable and m-separable functions is still an unsolved problem. The solution to the misconvergence problems in GA requires to understand the theory on how GA works and progresses the search. The next section focuses on that. It will be interesting to look at ICA problem, in terms of an optimization problem. The ICA problem requires to find optimal n-dimensional rotation to have statistically independent sources. As discussed previously, rotation of separable function (independent variables) gives non-separable functions. So, ICA problem is like, from the given non-separable function, find the anti-rotation to get back the separable variables. Also, n-dimensional rotation in ICA problem, requires $m = n(n-1)/2$ angle of 2-D rotations to be found optimally. Accordingly, n-D ICA is a

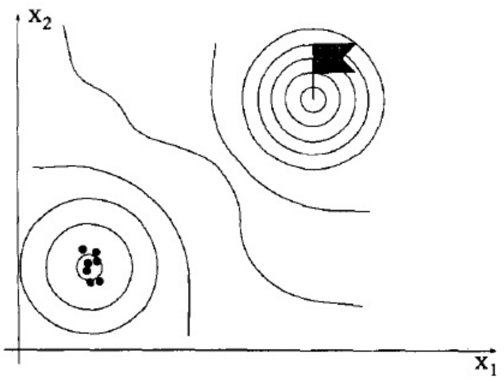

Fig. 2. A 2-dimensional function, in which the local optima are not distributed on a grid aligned with the coordinate system. The population has converged to the lower-left local optimum. In this situation, applying mutation to only one parameter is not sufficient to yield any progress.

specific type of m-D nonseparable function optimization problem.

6.3 GA Variants for BSS in higher dimensions

The initial experiments for BSS in higher dimensions using SRICA proved its failure to converge near optimal solution. Increasing computations helped only marginally. The misconvergence may be either due to improper use of the exploration and exploitation properties of GA or due to any of the previous reasons of misconvergence. Towards achieving this balance of exploitation and exploration, there were done many experiments. Chronologically they are: using other Evolutionary algorithms (EA), using different representations in GA, setting fixed parameter values in GA, making parameter values semi-adaptive (fixed in adaptive manner) in GA, implementing existing other search strategies like Co-operative Coevolution GA (CCGA) and then defining new search concepts like gradual search, delta search and others. They are reported in Section 6.3. Either the partial success or faiure in convergence demands the need to further explore the reasons of misconvergence. To reduce the coputations and avoid misconvergence, there were derived many variants of the basic or simple GA. The variants are based on varying the type of representation and varying the search 'scheme'. This GA variants are used with simaltaneous approach for BSS.

6.3.1 Representation Types

There were used three types of variable representations. With conventional binary and float repreesntation, there is also developed a gradual type of representtaion. The idea behind is, each representation has a specific accuracy in solution representation. More number of bits used for representation achieves better accuracy of representation but results into long strings of solutions specifically, for higher dimension simaltaneous search. This results into increased misconvergence probability due to genetic drift. Also, the problem due to 'Hamming Cliff' can be avoided by varying representation. Initial generations aims at exploration. So, small length binary code with low precision, reducing the total variable length are more suitable. In the letter generations, when exploitation or local search is needed the search can use more precise representations.

6.3.2 Various Search Schemes

There are defined 6 basic search schemes with 'simaltaneous' approach.

- *Simaltaneous*: The scheme searches for all the variables simaltaneously. To remind, n-dimensional BSS problem requires $m = n(n-1)/2$ search variables. The scheme may be combined with all the threee types.

- $Eqpart_k$: The scheme searches for k variables simaltaneously, where $1 \leq k \leq m$. The scheme basically corresponds to the Cooperative Coevolution (CC) or the 'divide and conquer' principle (95) that has been successfully aplied in many EAs, including GA (CCGA). The scheme may be combined with all the threee types. This scheme with $k = m$ is not same as the scheme Simaltaneous, as the sweep concept is not there in the Simaltaneous scheme.

- *Delta Search*: Usually, the search progresses in the whole solution space. But, when it is known that the optimal solution is in the vicinity of a certain part of the solution space, search should mainly focus on this highly probable area. This is achieved by generating a population through Gaussian distribution with small variace or Laplacian distribution with mean at the center of the most probable area. Thus, producing more individuals in the most probable region to find optimal and less samples in the remaining regions, the exploitation ability is increased. In case of lack of this knowledge, the scheme can be combined with other schemes and used whenever exploitation is needed.

- *Randpart*: The acheme is equivalent to the 'random grouping' principles in (143). The idea is to divide the search variables into random number of equal or unequal length groups in each sweep. In each sweep, the variables in a single group are simaltaneously searched to avoid linkage problem. So, with large number of sweeps, the scheme is supposed to avoid misconvergence in non-separable and semiseparable problems. Compare to the original scheme with swarm optimization in article (), here there are incorporated two changes with GA. One is, instead of suggetsed large groups in mutiple of 50 search variables, here there are used only small groups of say, 3 to 10 variables. The scheme may be combined with all the threee types.

- *Finegrad*: The scheme targets to use the power of gradual representation type in efiicient way. The total nmber of generations are divided into three stages: rough search, fine search and refined search. The rough search starts with small size binary codes and graually increasing the codesize. Then follows the few generation of fine search stage with search in float representation. Finally, the refine search stage assumes that the optimal fitness niche is already available in the population and now just further fine tuning matters. So, it uses float representtaion and generates new population members only around the selected best individuals. The part of the best solution from the end population of fine search stage are copied as an initial population of refine search stage. The remaining population is generated through Gaussian distribution with small variace or Laplacian distribution with centers as the best individuals. That means, by default the refine stage uses delta search. There may be internally more than one substages for each of these stages. The operator meters like selection, crossover and mutation probabilities are set to meet with the target of that stage. While switching from one stage or substage to the other, if the end population is as it is copiesd as the initial population then the lower bits are all same or zeros. That means, the genetic material corresponding to that bit positions is fixed and variation is lacking. To avoid this there three ways possible. First, the newer bit positions may be generated through random end bit populations. For example, while switching from 5 bit precision level to the 8 bit precision level, the lower endbits are added through randomly generated 3 bit population. This will also avoid misconvergence due to genetic drift at lower positions. One more fact not to be missed is, already searched best solution should be kept intact in the new population to assure ellitism. This is crucial as the new number system may not contain exactly the value in the previous number system. For example, the range [0 1] coded through 5 bit binary has found best at $11001 = 1/32 * 25 = 0.7813$. In an 8 bit binary coded representation, this value $25/32$ should be achievable. This best individual should be converted into $11001000 = 1/256 * 200 = 0.7813$. The second possible way is to throw away part of the end population and generating them newly with new precision at that stage. Thus, there is available sufficient genetic material at lower bit positions. The third that combines both the above solutions is also possible. Also, for the final stage the crossover probability is reduced.

- *Randfinegrad*: The scheme combines random grouping scheme with finegrad scheme. The random groups are search through finegrad scheme.

6.3.3 Various search strategies

: There are defined atleast 9 different search stratagies combining the various types and schemes. The Finegrad and 'randfinegrad' schemes by definition use gradual search type. Other schemes can be combined with any of the search type. While stage switch over in the simaltaneous scheme combined with gradual type, the best and randomly selected or most fit few of them are copied to the initial population of the next stage and the remaining population is newly generated. By default, randomly selected 25% of the total population including the best fit individuals are selected for the new stage. The $Eqpart_k$ scheme can be combined with the delta search, assuming in each new sweep only a correction is needed to the previous value and not the exploration. By default, the $Eqpart_k$ scheme uses delta search after each stage or substage.

The types, schemes and stratagies are summerized in the following Table 6.3.3.

Table 6.1: Description of the GA Search Schemes and Representation Types

Representation Type	Description
float	search variables are of type real
binary	search variables are of type binary
gradual	search variables are with gradually increasing accuracy, initially of type binary and finally real
delta	variable values as the deviation to the last best;
spiral	the search accuracy is increasing gradually in a cycle; repeatative such cycles; so the nomenclature spiral representation;
Search Scheme	**Description**
simultaneous	Simultaneously searching for all d variables
eqpart_k	Simultaneously searching for k variables sequentially selected, $1 \leq k \leq d$, thus $\lceil \frac{d}{k} \rceil$ group search per sweep; Basically CCGA with multiple sweeps - usually (d-1)
delta search	Search focused on the most probable optimality region; population generation through Gaussian or Laplacian distribution
finegrad	Gradual search with three stages: Coarse search, Fine search and Refine search
randgroup	Random grouping per sweep and small size groups
randfinegrad	Random grouping with finegard i.e. three stage search
spiral	uses spiral representation; going from high precision to low precision with elitism is possible through delta representation; each cycle aims exploration first and then exploitation; thus repeated cycles of exploration - exploitation
fixgrad	fixed way of number of search variables in each sweep; grouping randomly selected variables with randfinegrad, spiral

6.4 Application to LSGO standard test bench and LSnIBSS

The techniques are tested on the standard test bench for LSGO defined by (139) for the IEEE CEC2010 LSGO competition and also on BSS application. The partial results are shown in the following Table 6.2. The last row indicates the application of all the schemes on BSS of Speech4 data from ICALAB using Φ^{hyi} contrast.

Table 6.2: Performances of defined GA strategies against standard test bench functions; f_1: Shifted Elliptic function (Separable), f_{19}: Shifted Schwefels Problem 1.2 (Nonseparable), f_{20}: Shifted Rosenbrocks Function (Nonseparable)

fno	nsrc	Simal	Simal	Simal	randgroup	randgroup	randgroup	fixgrad
		float	binary	gradual	float	binary	gradual	finegrad
1	4	5.8003e06	1.2677e05	5.1319e04	3.7310e05	8.6028e03	3.1225e04	**3.8513e00**
	10	5.4364e07	1.4475e06	3.0447e05	3.5562e04	7.9341e02	2.8767e03	**1.3464e-01**
	20	2.3469e08	2.3401e06	2.1003e06	2.7746e01	1.8785e01	1.4290e02	**2.3793e-06**
19	4	6.2264e02	6.9459e01	**1.1141e02**	4.6643e02	2.9023e02	2.9171e02	2.8977e02
	10	6.2500e03	4.4429e02	6.2238e02	1.0866e03	7.0740e02	4.4494e02	**3.9499e02**
	20	3.4370e04	2.8443e03	3.1035e03	1.0565e03	1.5045e03	8.3672e02	**1.4052e03**
20	4	5.6068e07	5.5445e05	1.5886e04	3.2438e04	2.6523e03	3.4713e03	**1.6656e03**
	10	7.0200e08	6.7230e05	5.9044e05	2.2670e04	3.0648e03	2.6518e03	**1.2688e03**
	20	4.7301e09	1.1630e07	6.1228e06	2.8484e03	1.4763e03	1.5304e03	**9.3689e02**
Speec4	4	0.0558	0.0653	0.0755	0.0255	0.0450	0.0779	**0.0198**

The results verify the application of gradual search concept for nonseparable function LSGO. The real life problems are semiseparale type, where which variables are separable and which group of them are nonseparable is not known. The fixgrad scheme gives hope to be a better solution for semiseparable function LSGO by giving better results for both separable and nonseparable problems.

6.5 Conclusion

The gradual search concept has been proved useful for nonseparable function LSGO. The real life problems are semiseparale type, where which variables are separable and which group of them are nonseparable is not known. The fixgrad scheme gives hope to be a better solution for semiseparable function LSGO by giving better results for both separable and nonseparable problems. The same scheme has also assured better results i large scale for BSS problems with reduced computational cost.

Chapter 7

Conclusion and future work

The thesis actually has focused on three different issues: i. Linear BSS ii. Near independence BSS and iii. Large Scale BSS. The issues are combined as a large scale near-independence BSS problem. The solution has focused independently on deriving new contrasts for BSS based on the ITL theory and the large scale global optimization problem through EA. It has the following contributions:

- Towards independence measure for BSS
  - The difference between joint probability density function (PDF) and product of the marginal PDFs is defined as a Function Difference (FD) of a random vector. Based on the first and second order optima analysis, minimization of $L^p - Norm$ of FD is derived as a criteria for Blind Source Separation (BSS).
  - Instead of a, conventional, two stage estimation approach for FD (separate estimation of joint PDF and marginal PDFs and then the difference), the direct linear least squares FD (LSFD) estimation is achieved in two ways: (a) kernel basis placed at the selected paired sample points (b) kernel basis placed at the selected paired or un-paired sample points.
  - The performances of kernel methods depend upon the selected smoothing (bandwidth) parameter. There is derived *Extended Rule-of-Thumb* method for bandwidth selection in uni-variate Kernel Density Estimation (KDE). The derivation uses PDF approximation through Gram-Charlier series expansion.
  - A specific derivation for uni-variate generalized Gram-Charlier series is extended to multivariate generalized Gram-Charlier series. Based on this, the *Extended Rule-of-Thumb* method for bandwidth selection is extended to multivariate KDE.

  - Application of minimization of $L^2 - Norm$ of FD through LSFD estimator with bandwidth selection using the *Extended Rule-of-Thumb* method for BSS of linear mixtures is achieved.

- Towards analysis of non-independent BSS
  - A theoretical local minima analysis of some of the existing cumulants based approximations of indepependence measures for BSS is done. This is supported through empirical comparative study of the effects of the used independence measures and non-independent sources on the BSS solution. There has been observed either existence of local optima or shift of the global optima or both; through both the information theoretic and the kurtosis based optimization functions; for BSS of specific type of unimodal and multimodal PDF sources. The empirical study is done in higher dimensions also. The study is done through Search for Rotation based Independent Component Analysis (SRICA) algorithm, which uses GA as a search based global optimization method.

- Towards large scale global optimization (LSGO) and avoiding misconvergence in GA
  - Towards the algebra of GA, there exists the concept of equivalence relation based forma as a generalization to the notion of schema. This has been further generalized to dependency relation based extended forma. There has also been derived some operators exploiting extended formae (plural of forma) based similarities.
  - The suggested representation and operators are empirically used to derive Mendelian Genetic Algorithm (MGA).
  - Towards the partial success to reduce the computation for a non-separable global function optimization; there has been tried and tested different search strategies with GA. The search strategies, for example, are - varying representations (gradual search), spiral search, delta search, refine search, population reinitialization etc.

The part of the contribution can be viewed as significant extension to the theory of ITL and Machine Learning. The ITL, till now, is using only the direct kernel estimation methods to derive important statistics. The least square method for FD estimation gives a new way to use RKHS theory to derive important statistics. The Machine Learning and ITL based applications usually require to select the best solution from a given solution set. Using same bandwidth for all the solutions is equivalent to assuming all being with same PDF characteristics. The ExROT method can provide varying bandwidth parameter for varying solution from the given solution set.

The part of the contribution can also be viewed as significant towards the field of Applied Statistics. The theory behind derivation of the ExROT method uses PDF approximation through Gram-Charlier Series. The PDF approximation through infinite series is an independent area well travelled in Statistics. There exists many other such series defined using various reference PDFs and for various applications. The research can be used to derive new general and application specific rules for bandwidth parameter selection. Similarly, the multivariate extension of Gram-Charlier series can also be used for other applications in Statistics.

There seems many directions the future work can be extended. The following extensions are the ideas which are already verified through preliminary testing but have not been included due to the time constraint and limiting the scope of the thesis. They are as under:

1. The independence measure and the optimization methods applied here for linear BSS are also applicable to other mixing systems as Convolutive mixtures, post-nonlinear mixtures and convolutive post-nonlinear mixtures.

2. The Edgworth series is more precise and can definitely be used to get better estimations of bandwidth parameter. So, ExROT based on the Edgeworth series can be obtained.

3. The ICA concepts can be used to improve convergence in GA, MGA and aMGA by propagating search in independent directions. The concept already exist by propagating search in orthogonal directions through PCA.

4. The multivariate Gram-Charlier Series developed here can be used to perform theoretical local minima analysis in higher dimensions.

5. As FD, GFD and HFD as well their combinations can be used for BSS.

6. The gradient and projection based optimization techniques can be applied for the same independence measures.

7. The search strategies, for example, are - varying representations (gradual search), spiral search, delta search, refine search, population reinitialization etc. which are conceptualized during Thesis work, can be independently studied for their performances.

8. The overall solution, including the suggested independence measure and the optimization method can be applied to a real time large scale brain signal separation problem.

Bibliography

[1] Sophie Achard, Dinh Tuan Pham, and Christian Jutten. Quadratic dependence measure for nonlinear blind sources separation. In *4th International Conference on ICA and BSS*, pages 757–763, 2003.

[2] Shun-ichi Amari and Masayuki Kumon. Differential geometry of edgeworth expansions in curved exponential family. *Annals Of The Institute Of Statistical Mathematics*, 35(1):1–24, 1983.

[3] Shun-ichi Amari, Andrzej Cichocki, Howard Hua Yang, et al. A new learning algorithm for blind signal separation. *Advances in neural information processing systems*, pages 757–763, 1996.

[4] Jim Antonisse. A new interpretation of schema notation that overturns the binary coding constraint. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 86–91. Morgan Kaufmann (San Mateo), 1989.

[5] Leo A. Aroian. The type b gram-charlier series. *Ann. Math. Statist.*, 8(4):183–192, 12 1937.

[6] Leo A. Aroian. The probability function of the product of two normally distributed variables. *The Annals of Mathematical Statistics*, 18(2):pp. 265–271, 1947.

[7] Massoud Babaie-Zadeh. *On Blind Source Separation in Convolutive and Nonlinear Mixtures*. PhD thesis, Institut National Polytechnique de Grenoble (INPG), Grenoble, France and Sharif University of Technology, Tehran, IRAN, 20 septembre 2002.

[8] Massoud Babaie-Zadeh and Christian Jutten. A general approach for mutual information minimization and its application to blind source separation. *Signal Processing*, 85(5):975 – 995, 2005. ISSN 0165-1684.

[9] Massoud Babaie-Zadeh, Christian Jutten, and Kambiz Nayebi. Differential of the mutual information. *IEEE Signal Processing Letters*, 11(1):48–51, 2004.

[10] F.R. Bach and M.I. Jordan. Kernel independent component analysis. *The Journal of Machine Learning Research*, 3:1–48, 2003.

[11] A.J. Bell and T.J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[12] A. Belouchrani and J.-F. Cardoso. Maximum likelihood source separation by the expectation-maximization technique: deterministic and stochastic implementation. In *Proc. Int. Symp. on Nonlinear Theory and its Applications (NOLTA'95)*, pages 49–53, Las Vegas, Nevada, 1995.

[13] Mario Berberan-Santos. Expressing a probability density function in terms of another pdf: A generalized gram-charlier expansion. *Journal of Mathematical Chemistry*, 42(3):585–594, October 2007.

[14] S Berkowitz and FJ Garner. The calculation of multidimensional hermite polynomials and gram-charlier coefficients. *mathematics of computation*, pages 537–545, 1970.

[15] Dharmani Bhaveshkumar. The d-variate generalized gram-charlier series using k-order d-variate vector cumulants and kronecker product. *arXiv:1503.03212 [math.ST](Statistics Theory) (sent to Journal of Machine Learning Research (JMLR), MIT Press on 17th July, 2015)*, Mar:17, March 2015.

[16] C. M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, 1995.

[17] R. Boscolo and V. P. Roychowdhury. On the uniqueness of the minimum of the information-theoretic cost function for the separation of mixtures of nearly gaussian signals. In *ica03*, pages 137–141, Nara, Japan, apr 2003.

[18] R. Boscolo, H. Pan, and V.P. Roychowdhury. Independent component analysis based on nonparametric density estimation. *Neural Networks, IEEE Transactions on*, 15(1):55–65, Jan. 2004. ISSN 1045-9227. doi: 10.1109/TNN.2003.820667.

[19] Martijn Bousse, Otto Debals, and Lieven De Lathauwer. A tensor-based method for large-scale blind source separation using segmentation. *IEEE Journal of Selected Topics in Signal Processing*, 10(10), 2015.

[20] Newton L Bowers. Expansion of probability density functions as a sum of gamma densities with applications in risk theory. *Trans. Soc. Actuaries*, 18(52):125, 1966.

[21] J.-F. Cardoso. Source separation using higher order moments. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'89)*, pages 2109–2112, Glasgow, UK, 1989.

[22] J.-F. Cardoso. Eigen-structure of the fourth-order cumulant tensor with application to the blind source separation problem. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'90)*, pages 2655–2658, Albuquerque, New Mexico, 1990.

[23] J.-F. Cardoso. Iterative techniques for blind source separation using only fourth-order cumulants. In *Proc. EUSIPCO*, pages 739–742, Brussels, Belgium, 1992.

[24] J.-F. Cardoso. Infomax and maximum likelihood for source separation. *IEEE Letters on Signal Processing*, 4:112–114, 1997.

[25] J.-F. Cardoso. Blind signal separation: statistical principles. *Proceedings of the IEEE*, 9 (10):2009–2025, 1998.

[26] J.-F. Cardoso. High-order contrasts for independent component analysis. *Neural Computation*, 11(1):157–192, 1999.

[27] J.-F. Cardoso and P. Comon. Independent component analysis, a survey of some algebraic methods. In *Proc. ISCAS'96*, volume 2, pages 93–96, 1996.

[28] J.-F. Cardoso and B. Hvam Laheld. Equivariant adaptive source separation. *IEEE Trans. on Signal Processing*, 44(12):3017–3030, 1996.

[29] A. Cichocki, Amari S, Siwek K, T. Tanaka, Anh Huy Phan, and R. Zdunek. Icalab matlab toolbox ver. 3 for signal processing.

[30] A. Cichocki, D. Mandic, A-H. Phan, C. Caiafa, G. Zhou, Q. Zhao, and L. De Lathauwer. Tensor decompositions for signal processing applications from two-way to multiway component analysis. *IEEE Signal Processing Magazine*, 32(2):145–163, 2015.

[31] Leon Cohen. Generalization of the gram-charlier/edgeworth series andapplication to time-frequency analysis. *Multidimensional Syst. Signal Process.*, 9(4):363–372, October 1998.

[32] Leon Cohen. On the generalization of the edgeworth/gram-charlier series. *Journal of Mathematical Chemistry*, 49(3):625–628, March 2011.

[33] P. Comon. Independent component analysis—a new concept? *Signal Processing*, 36:287–314, 1994.

[34] P. Comon and B. Mourrain. Decomposition of quantics in sums of powers of linear forms. *Signal Processing*, 53(2):93–107, 1996.

[35] Pierre Comon and Christian Jutten. *Handbook of Blind Source Separation: Independent Component Analysis and Applications*. Academic Press, 1st edition, 2010. ISBN 0123747260, 9780123747266.

[36] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[37] Sergio A. Cruces-alvarez, Associate Member, and Andrzej Cichocki. From blind signal extraction to blind instantaneous signal separation: criteria, algorithm, and stability. *IEEE Trans. On Neural Networks*, 15:859–873, 2004.

[38] G. Darmois. Analyse générale des liaisons stochastiques. Étude particulière de l'analyse factorielle linéaire. *Rev. Inst. Internat. Statist.*, 21:2–8, 1953.

[39] A. W. Davis. Statistical distributions in univariate and multivariate edgeworth populations. *Biometrika*, 63(3):661–670, Dec., 1976.

[40] Esther B Del Brio, Trino-Manuel Ñíguez, and Javier Perote. Gram-charlier densities: A multivariate approach. *Quantitative Finance*, 9(7):855–868, 2009.

[41] J. Karhunen D.T. Pham, A. Ziehe and Ch. Jutten. Final technical report on linear ica. Technical report, HUT, INPG, FhG, McMaster University, 2003.

[42] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000. ISBN 0471056693.

[43] Tarn Duong. *Bandwidth selectors for multivariate kernel density estimation*. PhD thesis, School of Mathematics and Science, University of Western Australia, Australia, 2004.

[44] A. Eiben, P. Raué, and Z. Ruttkay. Genetic algorithms with multi-parent recombination. *Parallel Problem Solving from NaturePPSN III*, pages 78–87, 1994.

[45] V. A. Epanechnikov. Non parametric estimation of a multivariate probability density. *Theory of Probability and Its Applications, No. 14.*, pages 153–158, 1969.

[46] J. Eriksson and V. Koivunen. Identifiability, separability, and uniqueness of linear ica models. *Signal Processing Letters, IEEE*, 11(7):601–604, 2004.

[47] L.J. Eshelman. Real-coded genetic algorithms and interval-schemata. *Foundations of genetic algorithms*, 2:187–202, 1993.

[48] Mark Girolami and Colin Fyfe. Negentropy and kurtosis as projection pursuit indices provide generalised ica algorithms. In *A. C, Back A (eds.), NIPS-96 Blind Signal Separation Workshop*, volume 8, 1996.

[49] David E. Goldberg. Zen and the art of genetic algorithms. In *Proceedings of the third international conference on Genetic algorithms*, pages 80–85, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc. ISBN 1-55860-006-3.

[50] David E Goldberg. Construction of high-order deceptive functions using low-order walsh coefficients. *Annals of Mathematics and Artificial Intelligence*, 5(1):35–47, 1992.

[51] D.E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-wesley, 1989.

[52] D.E. Goldberg. Real-coded genetic algorithms, virtual alphabets, and blocking. *Complex Systems*, 5:139–167, 1991.

[53] JM Górriz, C.G. Puntonet, F. Rojas, R. Martin, S. Hornillo, and EW Lang. Optimizing blind source separation with guided genetic algorithms. *Neurocomputing*, 69(13-15):1442–1457, 2006. ISSN 0925-2312.

[54] Anders Hald. The early history of the cumulants and the gram-charlier series. *International Statistical Review*, 68(2):137–153, 2000. ISSN 1751-5823.

[55] Anders Hald and JF Steffensen. *On the history of series expansions of frequency functions and sampling distributions, 1873-1944*. Det Kongelige Danske Videnskabernes Selskab, 2002.

[56] Daniel J Henderson and Christopher F Parmeter. Normal reference bandwidths for the general order, multivariate kernel density derivative estimator. *Statistics & Probability Letters*, 82(12):2198–2205, 2012.

[57] F. Herrera, M. Lozano, and J.L. Verdegay. Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. *Artificial Intelligence Review*, 12(4):265–319, 1998.

[58] Kenneth E. II Hild, Deniz Erdogmus, and Jos Prncipe. Blind source separation using renyis mutual information. *IEEE Signal Processing Letters*, 8(6), 2001.

[59] J.H. Holland. *Adaptation in natural and artificial systems*. University of Michigan press, 1975.

[60] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, April 1992. ISBN 0262581116.

[61] Bjorn Holmquist. The d-variate vector hermite polynomial of order k. *Linear Algebra and its Applications*, 237-238(0):155–190, 1996. ISSN 0024-3795. Linear Algebra and Statistics: In Celebration of C. R. Rao's 75th Birthday (September 10, 1995).

[62] A. Hyvärinen. One-unit contrast functions for independent component analysis: A statistical analysis. In *Neural Networks for Signal Processing VII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, pages 388–397, Amelia Island, Florida, 1997.

[63] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3):626–634, 1999.

[64] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.

[65] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley, New York, 2001. 481+xxii pages.

[66] M. P. Wand J. S. Marron. Exact mean integrated squared error. *The Annals of Statistics, Vol. 20, No. 2.*, pages 712–736, 1992.

[67] Paul Janssen, James Stephen Marron, Noel Veraverbeke, and Warren Sarle. Scale measures for bandwidth selection. *Journaltitle of Nonparametric Statistics*, 5(4):359–380, 1995.

[68] Eric Jondeau and Michael Rockinger. Gram–charlier densities. *Journal of Economic Dynamics and Control*, 25(10):1457–1483, 2001.

[69] M. C. Jones, J. S. Marron, and S. J. Sheather. A brief survey of bandwidth selection for density estimation. *Journal of the American Statistical Association, 91*, pages 401–407, 1996.

[70] J. Karhunen, F. Meinecke, H. Valpola, and A. Ziehe. Final technical report on bss models and methods for non-independent bss. Technical report, HUT, FhG, 2003.

[71] Z. Koldovsky and P. Tichavsky. Efficient variant of algorithm fastica for independent component analysis attaining the cramer-rao lower bound. *Statistical Signal Processing, 2005 IEEE/SP 13th Workshop on*, pages 1090–1095, July 2005. doi: 10.1109/SSP.2005.1628758.

[72] Z. Koldovskỳ, P. Tichavskỳ, and E. Oja. Efficient variant of algorithm fastica for independent component analysis attaining the cramer-rao lower bound. *IEEE Trans. Neural Netw*, 17(5):1265–1277, 2006.

[73] Z. Koldovsky, J. Malek, P. Tichavsky, Y. Deville, and S. Hosseini. Extension of EFICA algorithm for blind separation of piecewise stationary non Gaussian sources. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 1913–1916. IEEE, 2008.

[74] Tonu Kollo and Dietrich von Rosen. A unified approach to the approximation of multivariate densities. *Scandinavian Journal of Statistics*, 25(1):pp. 93–109, 1998.

[75] E.G. Learned-Miller and W.F. John III. Ica using spacings estimates of entropy. *The Journal of Machine Learning Research*, 4:1271–1295, 2003.

[76] T.-W. Lee. *Independent Component Analysis - Theory and Applications*. Kluwer, 1998.

[77] T.-W. Lee, M. Girolami, and T. J. Sejnowski. Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, 11(2):417–441, 1999.

[78] F.G. Lobo, D.E. Goldberg, and M. Pelikan. Time complexity of genetic algorithms on exponentially scaled problems. *Urbana*, 51:61801, 2000.

[79] Jan R. Magnus. On the concept of matrix derivative. *Journal of multivariate analysis*, 101 (101):2200–2206, 2010.

[80] Jan R. Magnus and Heinz Neudecker. *Matrix differential calculus with applications in statistics and econometrics*. J. Wiley & Sons, Chichester, New York, Weinheim, 1999.

[81] P. McCullagh. *Tensor Methods in Statistics*. London; New York: Chapman & Hall, 1987.

[82] Melanie Mitchell, Stephanie Forrest, and John H Holland. The royal road for genetic algorithms: Fitness landscapes and ga performance. In *Proceedings of the first european conference on artificial life*, pages 245–254. Cambridge: The MIT Press, 1992.

[83] K.R. Müller, R. Vigario, F. Meinecke, and A. Ziehe. Blind source separation techniques for decomposing evoked brain signals. *International Journal of Bifurcation and Chaos*, 14(2): 773–791, 2004.

[84] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 3rd edition, 1991.

[85] Byeong Park and J. S. Marron. Comparison of data driven bandwidth selectors. *J. Statist. Assoc*, pages 66–72, 1990.

[86] Byeong Park and Berwin Turlach. Practical performance of several data driven bandwidth selectors. CORE Discussion Papers 1992005, University catholique de Louvain, Center for Operations Research and Econometrics (CORE), 1992.

[87] D.-T. Pham. Blind separation of instantaneous mixture sources via an independent component analysis. *IEEE Trans. on Signal Processing*, 44(11):2768–2779, 1996.

[88] Dinh-Tuan Pham. Fast algorithms for estimating mutual information, entropies and score functions. In *ICA 2003*, pages 17–22, 2003.

[89] Dinh-Tuan Pham. Fast algorithms for mutual information based independent component analysis. *IEEE Transactions on Signal Processing*, 52(10):2690–2700, 2004.

[90] Dinh-Tuan Pham and Frederic Vrins. Local minima of information-theoretic criteria in blind source separation. *IEEE Signal Process. Lett.*, 12(11):788–791, November 2005.

[91] Dinh-Tuan Pham, Frédéric Vrins, and Michel Verleysen. Spurious entropy minima for multimodal source separation. In *Eighth International Symposium on Signal Processing and its Applications, ISSPA'2005, August, 2005*, volume 1, pages 37–40, Sydney, Australie, 2005. IEEE.

[92] Dinh-Tuan Pham, Frederic Vrins, and Michel Verleysen. Spurious entropy minima for multimodal source separation. In *ISSPA*, pages 37–40, 2005.

[93] Dinh-Tuan Pham, Frédéric Vrins, and Michel Verleysen. On the risk of using rényi's entropy for blind source separation. *Signal Processing, IEEE Transactions on*, 56(10):4611–4620, 2008.

[94] D.T. Pham. Contrast functions for ica and sources separation technical report - bliss project. Technical report, HUT and FhG, 2001.

[95] Mitchell A Potter and Kenneth A De Jong. A cooperative coevolutionary approach to function optimization. In *Parallel problem solving from naturePPSN III*, pages 249–257. Springer, 1994.

[96] Jose C. Principe. *Information Theoretic Learning: Renyi's Entropy and Kernel Perspectives*. Springer Publishing Company, Incorporated, 1st edition, 2010. ISBN 1441915699, 9781441915696.

[97] Xiaofeng Qi and Francesco Palmieri. Theoretical analysis of evolutionary algorithms with an infinite population size in continuous space. part i: Basic properties of selection and mutation. *IEEE Transactions on Neural Networks*, 5:102–119, 1994.

[98] Xiaofeng Qi and Francesco Palmieri. Theoretical analysis of evolutionary algorithms with an infinite population size in continuous space. part ii: Analysis of the diversification role of crossover. *IEEE Transactions on Neural Networks*, 5:120–129, 1994.

[99] N.J. Radcliffe. Equivalence class analysis of genetic algorithms. *Complex Systems*, 5(2): 183–205, 1991.

[100] N.J. Radcliffe. Forma analysis and random respectful recombination. In *Proceedings of the fourth international conference on genetic algorithms*, pages 222–229. San Marco CA: Morgan Kaufmann, 1991.

[101] V. C. Raykar and R. Duraiswami. Very fast optimal bandwidth selection for univariate kernel density estimation. Technical Report CS-TR-4774, Department of computer science, University of Maryland, Collegepark, 2005.

[102] V. C. Raykar and R. Duraiswami. Fast optimal bandwidth selection for kernel density estimation. In J. Ghosh, D. Lambert, D. Skillicorn, and J. Srivastava, editors, *Proceedings of the sixth SIAM International Conference on Data Mining*, pages 524–528, 2006.

[103] Henry Lewis Rietz. *Mathematical Statistics*, volume 3 of *Carus Mathematical Monographs*. Mathematical Association of America, 1 edition, 1927.

[104] F. Rojas, I. Rojas, RM Clemente, and CG Puntonet. Nonlinear blind source separation using genetic algorithms. In *Proc. Int. Conf. on Independent Component Analysis and Signal Separation (ICA2001)*, pages 400–405. Citeseer, 2001.

[105] T. Subba Rao S. Rao Jammalamadaka and Gyorgy Terdik. Higher order cumulants of random vectors and applications to statistical inference and time series. *Sankhya: The Indian Journal of Statistics (2003-2007)*, 68(2):326–356, May, 2006.

[106] T. Sakai and M. Sugiyama. Computationally efficient estimation of squared-loss mutual information with multiplicative kernel models. *IEICE Transactions on Information and Systems*, E97-D(4):968–971, 2014.

[107] R. Salomon. Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. a survey of some theoretical and practical aspects of genetic algorithms. *BioSystems*, 39(3):263–278, 1996.

[108] J. Sarela and R. Vigario. The problem of overlearning in high-order ICA approaches: analysis and solutions. In J. Mira and A. Prieto, editors, *Proc. Int. Workshop on Artificial Neural Networks (IWANN-2001)*, pages 818–825, Granada, Spain, June 13-15 2001.

[109] J. Sarela and R. Vigario. Overlearning in marginal distribution-based ica: analysis and solutions. *The Journal of Machine Learning Research*, 4:1447–1469, 2003.

[110] P Sauer and G Heydt. A convenient multivariate gram-charlier type a series. *IEEE Transactions on Communications*, pages 247–248, 1979.

[111] D. C. Schleher. Generalized gram-charlier series with application to the sum of log-normal variates (corresp.). *IEEE Transactions on Information Theory*, 23(2):275–280, 1977.

[112] S. Seth, M. Rao, Il Park, and J. C. Principe. A unified framework for quadratic measures of independence. *Signal Processing, IEEE Transactions on*, 59(8):3624–3635, August 2011.

[113] S. Sheather. A data-based algorithm for choosing the window width when estimating the density at a point. *Computational Statistics and Data Analysis, 1,*, page 229238, 1983.

[114] S. Sheather. An improved data-based algorithm for choosing the window width when estimating the density at a point. *Computational Statistics and Data Analysis, 4,*, page 6165, 1986.

[115] S. Sheather. The performance of six popular bandwidth selection methods as same real data sets (with discussion). *Computational Statistics, 7,*, page 225250, 1986.

[116] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, 1986.

[117] V. P. Skitovich. Linear forms of independent random variables and the normal distribution law (in russian). *Seriya Matematiceskaya*, 18:185200, 1954.

[118] Ib M. Skovgaard. On multivariate edgeworth expansions. *International Statistical Review / Revue Internationale de Statistique*, 54(2):pp. 169–186, 1986.

[119] R. E. Smith and D. E. Goldberg. Diploidy and dominance in artificial genetic search. *Complex Systems*, 6(3):251–285, 1992.

[120] M. Sugiyama. Machine learning with squared-loss mutual information. *Entropy*, 15(1): 80–112, 2013.

[121] M. Sugiyama, T. Suzuki, and T. Kanamori. Density ratio matching under the Bregman divergence: A unified framework of density ratio estimation. *Annals of the Institute of Statistical Mathematics*, 64(5):1009–1044, 2012.

[122] M. Sugiyama, S. Liu, M. C. du Plessis, M. Yamanaka, M. Yamada, T. Suzuki, and T. Kanamori. Direct divergence approximation between probability distributions and its applications in machine learning. *Journal of Computing Science and Engineering*, 7(2): 99–111, 2013.

[123] M. Sugiyama, T. Suzuki, T. Kanamori, M. C. du Plessis, S. Liu, and I. Takeuchi. Density-difference estimation. *Neural Computation*, 25(10):2734–2775, 2013.

[124] Taiji Suzuki and Masashi Sugiyama. Least-squares independent component analysis. *Neural Computation*, 23(1):284–301, 2011.

[125] Gyorgy Terdik. Higher order statistics and multivariate vector hermite polynomials. *Teor. Imovir. Mat. Stat.*, 66:147–168, 2002.

[126] Fabian J. Theis, Andreas Jung, Carlos G. Puntonet, and Elmar W. Lang. Linear geometric ica: fundamentals and algorithms. *Neural Comput.*, 15(2):419–439, February 2003. ISSN 0899-7667.

[127] D. Thierens, D.E. Goldberg, and A.G. Pereira. Domino convergence, drift, and the temporal-salience structure of problems. In *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, pages 535–540. IEEE, 1998.

[128] Aman Ullah. Entropy, divergence and distance measures with econometric applications. *Journal of Statistical Planning and Inference*, 49(1):137–162, 1996.

[129] H. Valpola and P. Pajunen. Fast algorithms for Bayesian independent component analysis. In *Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)*, pages 233–237, Espoo, Finland, 2000.

[130] Larry A. Viehland. Velocity distribution functions and transport coefficients of atomic ions in atomic gases by a gramcharlier approach. *Chemical Physics*, 179(1):71 – 92, 1994.

[131] Michael D. Vose. Generalizing the notion of schema in genetic algorithms. *Artif. Intell.*, 50 (3):385–396, August 1991. ISSN 0004-3702. doi: 10.1016/0004-3702(91)90019-G.

[132] Frdric Vrins and Michel Verleysen. On the entropy minimization of a linear mixture of variables for source separation. *Signal Processing*, 85:1029–1044, 2005.

[133] Frédéric Vrins, John Aldo Lee, and Michel Verleysen. A minimum-range approach to blind extraction of bounded sources. *IEEE Transactions on Neural Networks*, 18(3):809–822, 2007.

[134] Frederic Vrins, Dinh-Tuan Pham, and Michel Verleysen. Mixing and non-mixing local minima of the entropy contrast for blind source separation. *IEEE Transactions on Information Theory*, 53(3):1030–1042, 2007.

[135] M. P. Wand and M. C. Jones. Comparison of smoothing parameterizations in bivariate kernel density estimation. *Journal of the American Statistical Association*, 88(422):520–528, June 1993. URL `http://oro.open.ac.uk/28293/`.

[136] P. Wand and C. Jones. *Kernel Smoothing*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1994.

[137] Christopher S Withers and Saralees Nadarajah. The dual multivariate charlier and edgeworth expansions. *Statistics & Probability Letters*, 87:76–85, 2014.

[138] A.H. Wright. Genetic algorithms for real parameter optimization. *Foundations of genetic algorithms*, 1:205–218, 1991.

[139] Li Xiaodong, Ke Tang, Mohammad N. Omidvar, Zhenyu Yang, and Kai Qin. Benchmark functions for the cec2013 special session and competition on large scale global optimization. *2015 IEEE Conferenc on Evolutionary Computations, Competetition on Large Scale Global Optimization*, 2013.

[140] Dongxin Xu. *Energy, entropy and information potential for neural computation*. PhD thesis, Citeseer, 1999.

[141] Dongxin Xu, Jose C Principe, John Fisher, and Hsiao-Chun Wu. A novel measure for independent component analysis (ica). In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 2, pages 1161–1164. IEEE, 1998.

[142] M. Yamada, T. Suzuki, T. Kanamori, H. Hachiya, and M. Sugiyama. Relative density-ratio estimation for robust distribution comparison. *Neural Computation*, 25(5):1324–1370, 2013.

[143] Zhenyu Yang, Ke Tang, and Xin Yao. Large scale evolutionary optimization using cooperative coevolution. *Information Sciences*, 178(15):2985–2999, 2008.

Appendix A

Measure, Metric, Norm and L^p-norm

Measures necessarily assign some nonnegative number to the members of a set in some systematic way. The distance measures or distance functions assign nonnegative value for two elements of a set. Let there be set S. Then, a distance function $d : S \times S \rightarrow \mathbb{R}$ may satisfy the following conditions for $x, y, z \in S$:

1. $d(x, y) \geq 0$ (non-negativity)
2. $d(x, y) = 0$ iff $x = y$ (identity of indiscemibles)
3. $d(x, y) = d(y, x)$ (symmetry)
4. $d(x, z) \leq d(x, y) + d(y, z)$ (subadditivity or triangle inequality)

The distance functions satisfying first two conditions are called divergence measures and those satisfying all four conditions are called *metric*. For example, if S contains n-dimensional vectors then $\forall \mathbf{x}, \mathbf{y} \in S, p \geq 1$, $d_p : S \times S \rightarrow \mathbb{R}$ defined as under is a *metric*.

$$d_p(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{\frac{1}{p}}$$

The concept to derive above *metric* is inspired by the distances in Euclidean geometry. The generalization of this distance measure on sets to that on vector spaces is obtained by defining a norm. Given a vector space V over a field F, a norm is a function $\rho : V \rightarrow \mathbb{R}$ with the above four properties of *metric* and added property of absolute Scale Invariance defined as under:

$$\rho(a\mathbf{x}) = |a|\rho(\mathbf{x}), \forall \mathbf{x} \in V, a \in F$$

For example, given an n-dimensional vector space $\mathbb{R}^n$ and $\mathbf{x} \in \mathbb{R}^n$; the L^p-norm of $\mathbf{x}$ for a real number $p \geq 1$, is defined as:

$$\|\mathbf{x}\|_p = (|x_1|^p + |x_2|^p + \ldots + |x_n|^p)^{\frac{1}{p}}$$

The same definition has been also extended for functions in L^p-spaces. A point in L^p-space is an L^p integrable function. A function $f : \mathbb{R}^n \to \mathbb{R}$ is L^p integrable, if p-th power of its absolute value is finite, or equivalently,

$$\|f(\mathbf{x})\|_p = \left(\int_{\mathbb{R}^n} |f(\mathbf{x})|^p d\mathbf{x} \right)^{\frac{1}{p}} < \infty$$

It is a complete normed space with all the L^p integrable functions, their linear combinations through real coefficients and including all limit points.

Appendix B

Information Potential (IP) and related Concepts

In a general sense, potential means an unrealized ability. The gravitational potential and the electric potential are the known examples from Physics. In both the examples, potential created by a particle (with mass or charge) is inversely proportional to the distance. In kernel density estimation, a kernel is placed at each sample location and usually kernel is a positive definite function decaying with distance. This fact brings analogy with the potential theory. Each sample is an information particle. The PDF is the information potential field in which the information particles interact with each other. In a scalar field, the total potential is the summation of potential due to individual particles. The information potential (IP) due to the system of samples or the field is given in a same way. For a random variable $\mathbf{x}$, the potential on a sample x_j due to other samples, assuming Gaussian kernel, is given by

$$V_2(x_j) \overset{def}{=} \frac{1}{N}\sum_{i=1}^{N} V_2(x_j, x_i) \text{ where, } V_2(x_j, x_i) = G_{\sigma\sqrt{2}}(x_j - x_i)$$

So, the IP of $\mathbf{x}$ is

$$V_2(\mathbf{x}) \overset{def}{=} \frac{1}{N}\sum_{j=1}^{N} V_2(x_j) = \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} V_2(x_j, x_i) = \int \{f\hat{(x)}^2\}$$

The quantity $V_2(x)$ or IP is same as the integration of the square of the PDF. Instead of usual sum in potential theory, the normalization is done to get integral over PDF to be 1. The subscript of V reminds us that this is the quadratic information potential (QIP) as square of the PDF is integrated. The definition is generalized for any α by defining V_α as the integral of α power of the density. Also, instead of a Gaussian kernel any other kernel can be selected. But, they may not have as smooth characteristic as for $\alpha = 2$ with Gaussian kernel. Using this result, ITL theory has defined several scalar descriptors of PDF, that just depend upon the available samples with whole PDF structure into consideration.

The Ψ_2^{LSFD} defined in the article, is already defined as QMI_{ED} by (96). The quantity QMI_{ED}, for a random vector $\mathbf{x} = (x_1, x_2)$, in terms of IP is derived as under:

$$\begin{aligned}
QMI_{ED}(x_1, x_2) &= D_{ED}(p_{x_1x_2}(x_1, x_2), p_{x_1}(x_1)p_{x_2}(x_2)) \\
&= \int_{x_2}\int_{x_1} (p_{x_1x_2}(x_1, x_2) - p_{x_1}(x_1)p_{x_2}(x_2))^2 dx_1 dx_2 \\
&= \int_{x_2}\int_{x_1} (p_{x_1x_2}(x_1, x_2))^2 dx_1 dx_2 + \int_{x_2}\int_{x_1} (-p_{x_1}(x_1)p_{x_2}(x_2))^2 dx_1 dx_2 \\
&- \int_{x_2}\int_{x_1} 2p_{x_1x_2}(x_1, x_2)p_{x_1}(x_1)p_{x_2}(x_2) dx_1 dx_2 \\
&= V_J + V_M - 2V_C
\end{aligned}$$

where, V_J is the IP of the joint PDF, V_M is the potential of the product of the marginal PDFs and V_C is the Cross Information Potential (CIP) similar to the concepts of cross entropy or cross correlation.

The potentials can be estimated through kernel methods.

$$\begin{aligned}
\hat{V_J} &= \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} \hat{V_2}(\mathbf{x}(i), \mathbf{x}(j)) \\
&= \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} G_{\sigma\sqrt{2}}(\mathbf{x}(i), \mathbf{x}(j)) \\
&= \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} G_{\sigma\sqrt{2}}(x_1(i) - x_1(j))G_{\sigma\sqrt{2}}(x_2(i) - x_2(j)) \\
&= \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} \hat{V_2}(x_1(i), x_1(j))\hat{V_2}(x_2(i), x_2(j))
\end{aligned}$$

$$\begin{aligned}
\hat{V_M} &= \left(\frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} G_{\sigma\sqrt{2}}(x_1(i) - x_1(j))\right)\left(\frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} G_{\sigma\sqrt{2}}(x_2(i) - x_2(j))\right) \\
&= \hat{V_2}(x_1)\hat{V_2}(x_2)
\end{aligned}$$

$$
\begin{aligned}
\hat{V_C} &= \int_{x_2}\int_{x_1} p_{x_1x_2}(x_1,x_2)p_{x_1}(x_1)p_{x_2}(x_2)dx_1dx_2 \\
&= \int\int \left[\frac{1}{N}\sum_{k=1}^{N} G_\sigma(x_1 - x_1(k))G_\sigma(x_2 - x_2(k))\right]\left[\frac{1}{N}\sum_{j=1}^{N} G_\sigma(x_1 - x_1(i))\right] \\
&\left[\frac{1}{N}\sum_{j=1}^{N} G_\sigma(x_2 - x_2(j))\right] dx_1dx_2 \\
&= \frac{1}{N}\sum_{i=1}^{N}\frac{1}{N}\sum_{j=1}^{N}\frac{1}{N}\sum_{k=1}^{N}\int G_\sigma(x_1 - x_1(i))G_\sigma(x_1 - x_1(k))dx_1 \\
&\int G_\sigma(x_2 - x_2(j))G_\sigma(x_2 - x_2(k))dx_2 \\
&= \frac{1}{N}\sum_{k=1}^{N}\left[\frac{1}{N}\sum_{i=1}^{N} G_{\sigma\sqrt{2}}(x_1(k) - x_1(i))\right]\left[\frac{1}{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(x_2(k) - x_2(j))\right] \\
&= \frac{1}{N}\sum_{k=1}^{N}\hat{V_2}(x_1(k))\hat{V_2}(x_2(k))
\end{aligned}
$$

B.1 Information Forces (IF)

It is obvious to think of information forces, once defined the IP. Potential and the force are related concepts. One of the interpretation of potential is the amount of work done required to bring a unit charge or unit mass from infinity to the point in the force field. The particle contains amount of potential energy that has been applied to work against the force. The force on sample x_j is the derivative of the IP at a sample with respect to the position of sample x_j, that is:

$$
\begin{aligned}
F_2\hat{(x_j)} &\overset{def}{=} \frac{\partial}{\partial x_j}\hat{V_2}(x_j) \\
&= \frac{1}{N}\sum_{i=1}^{N} G'_{\sigma\sqrt{2}}(x_j - x_i) = \frac{1}{N}\sum_{i=1}^{N}\hat{F_2}(x_j - x_i) \\
&= \frac{1}{2N\sigma^2}(x_i - x_j)G_{\sigma\sqrt{2}}(x_j - x_i)
\end{aligned}
$$

Appendix C

Computational Details in Chapter 3

C.1 The Multivariate Representations of GCA Series and GGC Series

As mention in Section 3.1, this section of the appendix describes existing representations of GCA series and GGC series. The goal is to place together various historical representations for the ease of comparison, on the level of difficulty or simplicity in representation, to the readers. Therefore, no attempt is made to explain their derivation or each terms in representation. For further details the actual references need be referred.

The GCA series representation using multi-element matrix notations for cumulants and moments by Sauer and Heydt (110) is as under:

$$f_{\mathbf{x}}(\mathbf{x}) = \sum_{s_1=0}^{\infty}\sum_{s_2=0}^{\infty}\cdots\sum_{s_d=0}^{\infty}\left[\mathbf{C}_{s_1 s_2\cdots s_d}\cdot(-1)^{\sum_{i=1}^{d}s_i}\prod_{p=1}^{d}H_{s_p}(x_p)G(x_p)\right] \tag{C.1}$$

$$\text{with, } \mathbf{C}_{s_1 s_2\cdots s_d} = \frac{E\left\{\prod_{i=1}^{d}H_{s_i}(X_i)\right\}}{(-1)^{\sum_{i=1}^{d}s_i}\prod_{j=1}^{d}s_j!} \tag{C.2}$$

where, $\mathbf{C}_{s_1 s_2\cdots s_d}$ is the constant depending upon cross-moments and $H_i(x)$ is the one-dimensional Hermite polynomial of i^{th} order.

The GCA Series representation using recursive formula for Hermite polynomials by Berkowitz and Garner (14) is as under:

$$f_{\mathbf{x}}(\mathbf{x}) = G(\mathbf{x}) \sum_{m=0}^{\infty} A_m H_m(\mathbf{z}) \tag{C.3}$$

$$\text{with, } A_m = \prod_{i=1}^{d} (m_i!)^{-1} \int_{\mathbb{R}} J_m(z) f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \tag{C.4}$$

where, $G(\mathbf{x})$ denote the multivariate Gaussian; $\mathbf{z} = (\mathbf{x} - \boldsymbol{\mu})$ is the normalized variable; $\boldsymbol{\mu}$ is the mean vector; $\{H_m(\mathbf{x})\}$ and $\{J_m(\mathbf{x})\}$ are the complete bi-orthogonal system of Hermite polynomials.

Using recursive relations of $\{J_m(\mathbf{x})\}$, A_m is given recursively as under:

$$A_m = \frac{1}{N} \left[\prod_{i=1}^{d} (m_i!)^{-1} \sum_{i=1}^{N} z_k^{(i)} J_{m-e_k}(y^i) - \sum_{i=1}^{N} r_{ki} m_k^{-1} A_{m-e_k-e_f} \right], k = 1, \ldots, d \tag{C.5}$$

where, N is the number of available samples and $\mathbf{e}_k$ is a vector with a "1" as the k^{th} component and "0" elsewhere. This defines the coefficients of expansions also recursively.

The GGC series representation using tensor notations for cumulants and Hermite polynomials by McCullagh (81, Chapter 5) is as under:

$$f_{\mathbf{x}}(x; \kappa) = f_0(\mathbf{x}) \left[1 + \eta^i h_i(\mathbf{x}) + \eta^{ij} h_{ij}(\mathbf{x})/2! + \eta^{ijk} h_{ijk}(\mathbf{x})/3! + \ldots \right] \tag{C.6}$$

where, $h_i(\mathbf{x}) = h_i(\mathbf{x}; \boldsymbol{\lambda}) = f_i(\mathbf{x})/f_0(\mathbf{x}), \quad h_{ij}(\mathbf{x}) = h_{ij}(\mathbf{x}; \boldsymbol{\lambda}) = f_{ij}(\mathbf{x})/f_0(\mathbf{x}), \ \ldots$ and $f_i(\mathbf{x}) = \partial f_0(\mathbf{x})/\partial x^i, f_{ij}(\mathbf{x}) = \partial^2 f_0(\mathbf{x})/\partial x^i \partial x^j, \ldots$; so on. Also, given $\kappa^i, \kappa^{i,j}, \kappa^{i,j,k}, \ldots$ are the cumulant tensors of random vector $\mathbf{x}$ and $\lambda^i, \lambda^{i,j}, \lambda^{i,j,k}, \ldots$ are the cumulant tensors of the reference *pdf* $f_0(\mathbf{x})$; we get:

$$\eta^i = \kappa^i - \lambda^i, \eta^{i,j} = \kappa^{i,j} - \lambda^{i,j}, \eta^{i,j,k} = \kappa^{i,j,k} - \lambda^{i,j,k}, \ \ldots$$

The formal 'moments' $\eta^i, \eta^{ij}, \eta^{ijk}, \ldots$ are defined based on the formal 'cumulants' (or the cumulant differences) $\eta^i, \eta^{i,j}, \eta^{i,j,k}$ and so on.

Taking $f_0(\mathbf{x}) = G(\mathbf{x})$ i.e. multivariate Gaussian density as the reference *pdf* and taking $\eta^i = 0, \eta^{i,j} = 0$ in above Equation (C.6); the GCA series based on cumulant tensors is written as under:

$$\begin{aligned} f_{\mathbf{x}}(x; \kappa) = G(\mathbf{x}) \big[& 1 + \kappa^{i,j,k} h_{ijk}(\mathbf{x})/3! + \kappa^{i,j,k,l} h_{ijkl}(\mathbf{x})/4! + \kappa^{i,j,k,l,m} h_{ijklm}(\mathbf{x})/5! \\ & + \left(\kappa^{i,j,k,l,m,n} + 10\kappa^{i,j,k} \kappa^{l,m,n} \right) h_{ijklmn}(\mathbf{x})/6! \ldots \big] \end{aligned} \tag{C.7}$$

As could be observed, the GGC series and GCA series using tensor notations adds quite an ease to representation. But, with increase in number of terms, the difficulty in representation increases.

The GCA series using vector moments and vector Hermite polynomials by Holmquist (61) is as under:

$$f_{\mathbf{x}}(\mathbf{x}) = G(\mathbf{x}-\boldsymbol{\mu};\mathbf{C}_{\mathbf{x}})\sum_{k=0}^{\infty}\frac{1}{k!}G_k^T(\mathbf{x}-\boldsymbol{\mu};\mathbf{C}_{\mathbf{x}})E\left\{\mathbf{H}_k(\mathbf{x}-\boldsymbol{\mu};\mathbf{C}_{\mathbf{x}})\right\} \tag{C.8}$$

where, $G_k(\mathbf{x}-\boldsymbol{\mu})$ is the k^{th} order vector derivative of $G(\mathbf{x}-\boldsymbol{\mu})$ and $E\left\{\mathbf{H}_k(\mathbf{x}-\boldsymbol{\mu})\right\}$ is the expectation of k^{th} order vector Hermite polynomial that is the function of vector moments.

$$E_f\left\{\mathbf{H}_k(\mathbf{X}-\boldsymbol{\mu};\mathbf{C}_{\mathbf{x}}^{-1})^{\otimes j}\right\} = k!\mathbf{S}_{d1_k}\sum_{j=0}^{[k/2]}\frac{\mathbf{m}_{k-2j}\otimes(-Vec\,\mathbf{C}_{\mathbf{x}})^{\otimes j}}{(k-2j)!j!2^j} \tag{C.9}$$

where, $E_f\left\{(\mathbf{X}-\boldsymbol{\mu})^{\otimes j}\right\} \equiv \mathbf{m}_j(\boldsymbol{\mu})$.

The GCA series representation using Bell polynomials is obtained by Withers and Nadarajah (137). Here, the Bell polynomials are represented through cumulant tensors. With $r = (r_1, r_2, \ldots, r_d) \in \mathbb{I}^d$, $\mathbf{x}^r = x_1^{r_1}\ldots x_d^{r_d}$, $r! = r_1!r_2!\ldots r_d!$ and $|r| = r_1 + r_2 + \ldots + r_d$; the GCA series is as under:

$$f_{\mathbf{x}}(\mathbf{x})/G(\mathbf{x}) - 1 = \sum_{|r|\geq 3}^{\infty}\mathbf{B}_r H_r(x,\mathbf{C}_{\mathbf{x}})/r! \tag{C.10}$$

$$\text{where, } \mathbf{B}_r = \sum_{j=0}^{|r|}\mathbf{B}_{r,j} \tag{C.11}$$

$$\mathbf{B}_{r,j} = \frac{1}{k_1(r-j)}\sum_{l=1}^{r-j}\binom{r}{j}\left(j+1-\frac{r+1}{l+1}\right)k_{l+1}\mathbf{B}_{r-l,j} \tag{C.12}$$

$$k_r = \kappa(X_{r1}, X_{r2}, \ldots, X_{rd}) \tag{C.13}$$

where, k_r is the r^{th} order cumulant and $\kappa(\mathbf{x})$ is the moment.

Finally, the existing representations described in this Section C.1 can be compared with those derived in the current article. , can be compared with the GCA series derived in Equation (3.59) and the GGC series derived in Equation (3.66) combined with Equation (3.68). More specifically, the GCA series in Equation (C.1) using multi-element array representation and the GGC series in Equation (C.6) using tensor representation can be compared with the GCA series derived in Equation (3.59) and the GGC series derived in Equation (3.66), combined with Equation (3.68), in vector notations. The comparison demonstrates the ease obtained in representation. The same advantages are also obtained for other intermediate results in the article.

C.2 Some important properties of the K-derivative operator

Some important properties of the K-derivative operator are listed below.

Property 3 (Scaling Property)**.** *Let* $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_d)'$*,* $\boldsymbol{\lambda} \in \mathbb{R}^d$*,* $\mathbf{f}(\boldsymbol{\lambda}) \in \mathbb{R}^m$ *and* $\mathbf{f}_1(\boldsymbol{\lambda}) = \mathbf{A}\mathbf{f}(\boldsymbol{\lambda})$*, where* $\mathbf{A}$ *is an* $n \times m$ *matrix. Then*

$$\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}(\mathbf{f}_1) = (\mathbf{A} \otimes \mathbf{I}_d)\, \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}(\mathbf{f}) \tag{C.14}$$

where, $\mathbf{I}_d$ *is a d-dimensional unit matrix.*

Property 4 (Chain Rule)**.** *Let* $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_d)'$*,* $\boldsymbol{\lambda} \in \mathbb{R}^d$*,* $\mathbf{f}(\boldsymbol{\lambda}) \in \mathbb{R}^{m_1}$ *and* $\mathbf{g}(\boldsymbol{\lambda}) \in \mathbb{R}^{m_2}$*. Then*

$$\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}(\mathbf{f} \otimes \mathbf{g}) = \mathbf{K}_{3\leftrightarrow 2}^{-1}(m_1, m_2, d)((\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\mathbf{f}) \otimes \mathbf{g}) + \mathbf{f} \otimes (\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\mathbf{g}) \tag{C.15}$$

where, $\mathbf{K}_{3\leftrightarrow 2}(m_1, m_2, d)$ *is a commutation matrix of size* $m_1 m_2 d \times m_1 m_2 d$ *that changes the order of the the Kronecker product components. For example,*

$$\mathbf{K}_{3\leftrightarrow 2}(m_1, m_2, d)(\mathbf{a_1} \otimes \mathbf{a_2} \otimes \mathbf{a_3}) = \mathbf{a_1} \otimes \mathbf{a_3} \otimes \mathbf{a_2}$$

The above K-derivative properties can be used to derive further the following properties:

1. $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\boldsymbol{\lambda} = Vec\,\mathbf{I}_d$

2. $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\boldsymbol{\lambda}^{\otimes k} = k\left(\boldsymbol{\lambda}^{\otimes k-1} \otimes Vec\,\mathbf{I}_d\right)$

3. Let $g(\boldsymbol{\lambda}) = \mathbf{a}' f(\boldsymbol{\lambda})$
   $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes} g(\boldsymbol{\lambda}) = (\mathbf{a} \otimes \mathbf{I}_d)'\, \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes} f(\boldsymbol{\lambda})$

4. $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\mathbf{a}'\boldsymbol{\lambda}^{\otimes k} = k\,(\mathbf{a} \otimes \mathbf{I}_d)'\left(\boldsymbol{\lambda}^{\otimes k-1} \otimes Vec\,\mathbf{I}_d\right) = k\;\mathbf{a}'\left(\boldsymbol{\lambda}^{\otimes k-1} \otimes \mathbf{I}_d\right)$

5. $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes}\mathbf{a}'^{\otimes k}\boldsymbol{\lambda}^{\otimes k} = k\left(\mathbf{a}^{\otimes k} \otimes \mathbf{I}_d\right)'\left(\boldsymbol{\lambda}^{\otimes k-1} \otimes Vec\,\mathbf{I}_d\right) = k\,(\mathbf{a}'\boldsymbol{\lambda})^{\otimes k-1} \otimes \mathbf{a}$

6. $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes r}\boldsymbol{\lambda}^{\otimes k} = k(k-1)\cdots(k-r+1)\left(\boldsymbol{\lambda}^{\otimes k-r} \otimes (Vec\,\mathbf{I}_d)^{\otimes r}\right)$
   ($\because$ Repeated application of the Chain Rule)

7. $\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes r}\mathbf{a}'^{\otimes k}\boldsymbol{\lambda}^{\otimes k} = k(k-1)\cdots(k-r+1)\left(\mathbf{a}^{\otimes k} \otimes \mathbf{I}_d^{\otimes r}\right)'\left(\boldsymbol{\lambda}^{\otimes k-r} \otimes (Vec\,\mathbf{I}_d)^{\otimes r}\right)$
   $= k(k-1)\cdots(k-r+1)\,(\mathbf{a}'\boldsymbol{\lambda})^{\otimes k-r} \otimes \mathbf{a}^{\otimes r}$
   ($\because$ Repeated application of the Chain Rule and the property:)

C.3 The Taylor series expansion of some required functions near zero

The Taylor series expansion of the required functions near $\boldsymbol{\lambda} = \mathbf{0}$, based on the Equation (3.21), are given as under:

$$\mathrm{e}^{\mathbf{a}'\mathbf{x}} = \sum_{k=0}^{\infty} \frac{(\mathbf{a}'\mathbf{x})^{\otimes k}}{k!} = 1 + \mathbf{a}'\mathbf{x} + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 2}}{2!} + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 3}}{3!} + \dots \tag{C.16}$$

$$\sin(\mathbf{a}'\mathbf{x}) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!}(\mathbf{a}'\mathbf{x})^{\otimes 2k+1} = \mathbf{a}'\mathbf{x} - \frac{(\mathbf{a}'\mathbf{x})^{\otimes 3}}{3!} + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 5}}{5!} - \dots \tag{C.17}$$

$$\cos(\mathbf{a}'\mathbf{x}) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k)!}(\mathbf{a}'\mathbf{x})^{\otimes 2k} = 1 - \frac{(\mathbf{a}'\mathbf{x})^{\otimes 2}}{2!} + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 4}}{4!} - \dots \tag{C.18}$$

$$\begin{aligned}\mathrm{e}^{\mathbf{a}'\mathbf{x}}\cos(\mathbf{b}'\mathbf{y}) &= 1 + \mathbf{a}'\mathbf{x} + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 2}}{2!} - \frac{(\mathbf{b}'\mathbf{y})^{\otimes 2}}{2!} + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 3}}{3!} - \frac{(\mathbf{a}'\mathbf{x}) \otimes (\mathbf{b}'\mathbf{y})^{\otimes 2}}{2!} \\ &\quad + \frac{(\mathbf{a}'\mathbf{x})^{\otimes 4}}{4!} + \frac{(\mathbf{b}'\mathbf{y})^{\otimes 4}}{4!} - \frac{(\mathbf{a}'\mathbf{x})^{\otimes 2} \otimes (\mathbf{b}'\mathbf{y})^{\otimes 2}}{2!2!} + \dots\end{aligned} \tag{C.19}$$

$$\begin{aligned}\mathrm{e}^{\mathbf{a}'\mathbf{x}}\sin(\mathbf{b}'\mathbf{y}) &= \mathbf{b}'\mathbf{y} + (\mathbf{a}'\mathbf{x}) \otimes (\mathbf{b}'\mathbf{y}) - \frac{(\mathbf{a}'\mathbf{x})^{\otimes 2} \otimes (\mathbf{b}'\mathbf{y})}{2!} - \frac{(\mathbf{b}'\mathbf{y})^{\otimes 3}}{3!} \\ &\quad - \frac{(\mathbf{a}'\mathbf{x}) \otimes (\mathbf{b}'\mathbf{y})^{\otimes 3}}{3!} + \frac{(\mathbf{b}'\mathbf{y})^{\otimes 5}}{5!} - \frac{(\mathbf{a}'\mathbf{x})^{\otimes 2} \otimes (\mathbf{b}'\mathbf{y})^{\otimes 3}}{2!3!} + \dots\end{aligned} \tag{C.20}$$

C.4 Some Proofs

C.4.1 K-derivative of $\delta(\mathbf{x})$

Based on the Differentiation Property of Fourier Transform, the following is obtained:

$$\begin{aligned}\mathsf{F}^{-1}\left(\mathsf{F}\left(\mathbf{D}^{\otimes k} f(\mathbf{x})\right)\right) &= \mathsf{F}^{-1}\left((i\boldsymbol{\lambda})^{\otimes k}\mathsf{F}(f(\mathbf{x}))\right) \\ \Rightarrow \mathbf{D}^{\otimes k}\delta(\mathbf{x}) &= \mathsf{F}^{-1}(i\boldsymbol{\lambda})^{\otimes k}\end{aligned}$$

C.4.2 Relationship between $\mathrm{m}(k,d)$ and $\mathrm{c}(k,d)$ for $k=2$ and $k=3$

For $k = 2$; using Equation (3.29), Equation (3.33) and the differentiation rules for Kronecker product in Appendix C.2; the following is derived:

$$
\begin{aligned}
&\mathbf{m}(2,d) = \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes 2}\mathbf{M}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}} \qquad \text{(C.21)}\\
&\Rightarrow \sum_{p=2}^{\infty}\mathbf{m}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-2)}\otimes\mathbf{I}_d\otimes\mathbf{I}_d\right)}{(p-2)!}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}} = \mathbf{K}^{-1}_{\mathfrak{p}3\leftrightarrow 2}(d_1,d_2,d)\left\{\left(\sum_{p=2}^{\infty}\mathbf{c}(p,d)'\right.\right.\\
&\qquad\left.\left.\frac{\left(\boldsymbol{\lambda}^{\otimes(p-2)}\otimes\mathbf{I}_d\otimes\mathbf{I}_d\right)}{(p-2)!}\right)\otimes\exp\left(\sum_{q=1}^{\infty}\mathbf{c}(q,d)'\frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\right\}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}}\\
&\qquad+\left\{\left(\sum_{p=1}^{\infty}\mathbf{c}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)}\otimes\mathbf{I}_d\right)}{(p-1)!}\right)^{\otimes 2}\otimes\exp\left(\sum_{q=1}^{\infty}\mathbf{c}(q,d)'\frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\right\}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}}\\
&\Rightarrow\mathbf{m}(2,d) = \mathbf{c}(2,d)+\mathbf{c}(1,d)^{\otimes 2} \qquad \text{(C.22)}
\end{aligned}
$$

where, $\mathfrak{p}3 \leftrightarrow 2 \in \mathfrak{P}$ is the required permutation and $\mathbf{K}_{\mathfrak{p}3\leftrightarrow 2}(d_1,d_2,d)$ is the corresponding commutation matrix.

For $k = 3$; using Equation (3.29), Equation (3.33) and the differentiation rules for Kronecker product in Appendix C.2; the following is derived:

$$\mathbf{m}(3,d) = \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes 3}\mathbf{M}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=\mathbf{0}} \tag{C.23}$$

$$\Rightarrow \sum_{p=3}^{\infty}\mathbf{m}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-3)}\otimes\mathbf{I}_{d^3}\right)}{(p-3)!}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}} = \mathbf{K}^{-1}_{\mathfrak{p}3\leftrightarrow 2}(d_1,d_2,d)\mathbf{K}^{-1}_{\mathfrak{p}3\leftrightarrow 2}(d_1,d_2,d)$$

$$\left\{\left(\sum_{p=3}^{\infty}\mathbf{c}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-3)}\otimes\mathbf{I}_{d^3}\right)}{(p-3)!}\right)\otimes\exp\left(\sum_{q=1}^{\infty}\mathbf{c}(q,d)'\frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\right\}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}}$$

$$+\,\mathbf{K}^{-1}_{\mathfrak{p}3\leftrightarrow 2}(d_1,d_2,d)\left\{\left(\sum_{p=2}^{\infty}\mathbf{c}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-2)}\otimes\mathbf{I}_{d^2}\right)}{(p-2)!}\right)\otimes\left(\sum_{p=1}^{\infty}\mathbf{c}(p,d)'\right.\right.$$

$$\left.\left.\frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)}\otimes\mathbf{I}_{d}\right)}{(p-1)!}\right)\otimes\exp\left(\sum_{q=1}^{\infty}\mathbf{c}(q,d)'\frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\right\}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}}+\mathbf{K}^{-1}_{\mathfrak{p}3\leftrightarrow 2}(d_1,d_2,d)$$

$$\left\{2\left(\sum_{p=2}^{\infty}\mathbf{c}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-2)}\otimes\mathbf{I}_{d^2}\right)}{(p-2)!}\right)^{\otimes 1}\otimes\left(\sum_{p=1}^{\infty}\mathbf{c}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)}\otimes\mathbf{I}_{d}\right)}{(p-1)!}\right)^{\otimes 1}\right.$$

$$\left.\otimes\exp\left(\sum_{q=1}^{\infty}\mathbf{c}(q,d)'\frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\right\}\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}}+\left(\sum_{p=1}^{\infty}\mathbf{c}(p,d)'\frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)}\otimes\mathbf{I}_{d}\right)}{(p-1)!}\right)^{\otimes 2}\otimes\left(\sum_{p=1}^{\infty}\mathbf{c}(p,d)'\right.$$

$$\left.\frac{\left(\boldsymbol{\lambda}^{\otimes(p-1)}\otimes\mathbf{I}_{d}\right)}{(p-1)!}\right)^{\otimes 1}\exp\left(\sum_{q=1}^{\infty}\mathbf{c}(q,d)'\frac{\boldsymbol{\lambda}^{\otimes q}}{q!}\right)\Bigg|_{\boldsymbol{\lambda}=\mathbf{0}}$$

$$\Rightarrow\mathbf{m}(3,d) = \mathbf{c}(3,d) + 3\mathbf{c}(2,d)\otimes\mathbf{c}(1,d) + \mathbf{c}(1,d)^{\otimes 3} \tag{C.24}$$

C.4.3 Multivariate Gaussian representation in terms of the cumulants (The proof related to the Section 3.8

Applying $\mathbf{D}_{\mathbf{x}}^{\otimes}$ on Equation (3.42),

$$\mathbf{D}_{\mathbf{x}}^{\otimes} f(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} (\boldsymbol{\lambda} \otimes \mathbf{I}_d) \otimes \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2}\right) \sin\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda}\right) d\boldsymbol{\lambda} \tag{C.25}$$

$$\text{Let, } U(\boldsymbol{\lambda}) = \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2}\right) \tag{C.26}$$

$$\Rightarrow \mathbf{D}_{\boldsymbol{\lambda}}^{\otimes} U(\boldsymbol{\lambda}) = -\mathbf{c}(2,d)' (\boldsymbol{\lambda} \otimes \mathbf{I}_d) \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2}\right) \tag{C.27}$$

$$\text{Also, let } V(\boldsymbol{\lambda}) = (\mathbf{x} - \mathbf{c}(1,d)) \cos\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda}\right) \tag{C.28}$$

$$\Rightarrow \int V d\boldsymbol{\lambda} = \sin\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda}\right) \tag{C.29}$$

Using above Equation (C.26), Equation (C.28) in Equation (C.25) and taking $inv\ \mathbf{c}(2,d) = Vec\left(\mathbf{C}_{\mathbf{x}}^{-1}\right)$; we get:

$$\mathbf{D}_{\mathbf{x}}^{\otimes} f(\mathbf{x}) = -\frac{inv\ \mathbf{c}(2,d)'}{(2\pi)^d} \int_{\mathbb{R}^d} \left(\mathbf{D}_{\boldsymbol{\lambda}}^{\otimes} U(\boldsymbol{\lambda})\right) \left(\int V d\boldsymbol{\lambda}\right) d\boldsymbol{\lambda} \tag{C.30}$$

Applying integration by parts to above Equation (C.30), we get:

$$\begin{aligned} \mathbf{D}_{\mathbf{x}}^{\otimes} f(\mathbf{x}) = -\frac{inv\ \mathbf{c}(2,d)'}{(2\pi)^d} \Bigg\{ & \exp\left(-\frac{1}{2}\mathbf{c}(2,d)' \boldsymbol{\lambda}^{\otimes 2}\right) \sin\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda}\right)\Bigg|_{\mathbb{R}^d} \\ & - \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2}\right) (\mathbf{x} - \mathbf{c}(1,d)) \cos\left((\mathbf{x} - \mathbf{c}(1,d))' \boldsymbol{\lambda}\right) d\boldsymbol{\lambda} \Bigg\} \end{aligned} \tag{C.31}$$

$$= inv\ \mathbf{c}(2,d)' (\mathbf{x} - \mathbf{c}(1,d)) f(\mathbf{x}) \tag{C.32}$$

The solution of above differential equation leads to the following:

$$f(\mathbf{x}) = c \exp\left(-\left(inv\ \mathbf{c}(2,d)\right)' (\mathbf{x} - \mathbf{c}(1,d))\right) \tag{C.33}$$

$$\begin{aligned} \text{where, } c = f(\mathbf{0}) &= \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2}\right) d\boldsymbol{\lambda} \\ &= (2\pi)^{-d/2} \left(|\mathbf{C}_{\mathbf{x}}|\right)^{-1/2} \end{aligned} \tag{C.34}$$

$$\Rightarrow f(\mathbf{x}) = (2\pi)^{-d/2} \left(|\mathbf{C}_{\mathbf{x}}|\right)^{-1/2} \exp\left(-\left(inv\ \mathbf{c}(2,d)\right)' (\mathbf{x} - \mathbf{c}(1,d))\right) \tag{C.35}$$

$$\Rightarrow \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \exp\left(-\frac{\mathbf{c}(2,d)'}{2} \boldsymbol{\lambda}^{\otimes 2}\right) \cos\left(\mathbf{x}'\boldsymbol{\lambda} - \mathbf{c}(1,d)'\boldsymbol{\lambda}\right) d\boldsymbol{\lambda} = G(\mathbf{x}) \tag{C.36}$$

.1 AMISE for bandwidth parameter selection in KDE

Given N realizations of an unknown PDF $f(x)$, the kernel density estimate $f(\hat{x})$ is given by

$$f(\hat{x}) = \frac{1}{Nh}\sum_{i=1}^{N} K\left(\frac{x-x_i}{h}\right) = \frac{1}{N}\sum_{i=1}^{N} K_h\left(x-x_i\right) \tag{37}$$

where, $K(u)$ is the kernel function and h is the bandwidth parameter. Usually, $K(u)$ is a symmetric, positive definite and bounded function; mostly a PDF; satisfying the following properties:

$$K(u) \geq 0, \int_{\mathbb{R}^d} K(u)du = 1, \int_{\mathbb{R}^d} uK(u)Du = 0, \int_{\mathbb{R}^d} u^2K(u)du = \mu_2(K) < \infty$$

The accuracy of a PDF estimation can be quantified by the available distance measures between PDFs; like; L_1 norm based mean integrated absolute measure, L_2 norm based mean integrated square error (MISE), Kullback-Libeler divergence and others. The optimal smoothing parameter (the bandwidth) h is obtained by minimizing the selected distance measure. The bandwidth selection rule based on the most widely used criteria MISE or IMSE (Integrated Mean Square Error) is derived as under (116, 136).

$$\begin{aligned}
\text{ISE}(f(x), f(\hat{x})) &= L_2(f(x), f(\hat{x})) := \int_{\mathbb{R}^d} (f(\hat{x}) - f(x))^2 dx \\
\text{MISE}(f(x), f(\hat{x})) &= E\{ISE(f(x), f(\hat{x}))\} = E\left\{\int_{\mathbb{R}^d} (f(\hat{x}) - f(x))^2 dx\right\} \\
&= \int_{\mathbb{R}^d} E\{(f(\hat{x}) - f(x))^2\}dx = \int_{\mathbb{R}^d} \text{MSE}(f(x), f(\hat{x})) = \text{IMSE}(f(x), f(\hat{x})) \\
&= \int_{\mathbb{R}^d} (E\{f(\hat{x})\} - f(x))^2 + E\{(f(\hat{x}) - E\{f(\hat{x})\})^2\}dx \\
&= \int_{\mathbb{R}^d} \text{Bias}^2(f(\hat{x}))dx + \int_{\mathbb{R}^d} \text{Var}(f(\hat{x}))dx
\end{aligned} \tag{38}$$

$$\begin{aligned}
\text{Now, } E\{f(\hat{x})\} &= \frac{1}{Nh}\sum_{i=1}^{N} E\left\{K\left(\frac{x-x_i}{h}\right)\right\} \\
&= \frac{1}{h}\int_{\mathbb{R}^d} K\left(\frac{x-s}{h}\right) f(s)ds \\
&= \int_{\mathbb{R}^d} K(z)f(x-hz)dz \text{ (} \because \text{ substituting } z = \tfrac{x-s}{h} \text{)}
\end{aligned}$$

Expanding $f(x-hz)$ using Taylor Series with the assumption of $f(x)$ being a smooth PDF i.e. all derivatives exist

$$\begin{aligned}
E\{f\hat{(x)}\} &= \int_{\mathbb{R}^d} K(z)\left(f(x) - hzf'(x) + \frac{h^2z^2}{2}f''(x) + O(h^2)\right)dz \\
&= f(x) + \frac{h^2}{2}\mu_2(K)f''(x) + O(h^2) \\
\Rightarrow \text{Bias}(f\hat{(x)}) &\approx \frac{h^2}{2}\mu_2(K)f''(x) \qquad (39)
\end{aligned}$$

where, $\mu_2(K) = \int z^2K^2(z)dz$, the second order central moment of the kernel.

$$\begin{aligned}
\text{Similarly, Var}(f\hat{(x)}) &= \text{Var}\left(\frac{1}{Nh}\sum_{i=1}^{N} K\left(\frac{x-x_i}{h}\right)\right) \\
&= \frac{1}{N^2h^2}\sum_{i=1}^{N}\text{Var}\left(K\left(\frac{x-x_i}{h}\right)\right) \\
&= \frac{1}{Nh^2}\left[E\left\{K^2\left(\frac{x-x_i}{h}\right)\right\} - E^2\left\{K\left(\frac{x-x_i}{h}\right)\right\}\right] \\
&= \frac{1}{N}\int_{\mathbb{R}^d}\frac{1}{h^2}\left(K^2\left(\frac{x-s}{h}\right)\right)f(s)ds - \frac{1}{N}\left(\frac{1}{h}\int_{\mathbb{R}^d}K\left(\frac{x-s}{h}\right)f(s)ds\right)^2 \\
&= \frac{1}{Nh}\int_{\mathbb{R}^d}K^2(z)f(x-hz)dz - \frac{1}{N}\left(f(x) + \text{Bias}(f\hat{(x)})\right)^2 \qquad (40)
\end{aligned}$$

where, s is the mean of x and $z = \frac{x-s}{h}$. Now, expanding $f(x-hz)$ using Taylor Series with assumption of $f(x)$ being a smooth PDF i.e. all derivatives exist

$$\begin{aligned}
\text{Var}(f\hat{(x)}) &= \frac{1}{Nh}\int_{\mathbb{R}^d}K^2(z)\left(f(x) - hzf'(x) + \frac{h^2z^2}{2}f''(x) + O(h^2)\right)dz \\
&\quad -\frac{1}{N}\left(f(x) + \frac{h^2z^2}{2}f''(x) + O(h^2)\right)^2 \\
\Rightarrow \text{Var}(f\hat{(x)}) &\approx \frac{1}{Nh}f(x)\int K^2(z)dz \ (\because \text{assuming large } N\text{, small } h\text{ }) \qquad (41)
\end{aligned}$$

Combining equations (38), (39) and (41)

$$\text{MISE}(f\hat{(x)}) = \frac{h^4}{4}(\mu_2(K))^2R(f'') + \frac{1}{Nh}R(K) + O(h^4) + O\left(\frac{h}{N}\right)$$

where, $R(f'') = \int (f''(x))^2dx$ and $R(K) = \int K^2(z)dz$. In general, $R(g) = \int g^2(z)dz$ is identified as the roughness of function $g(x)$. An asymptotic large sample approximation AMISE is obtained, assuming $\lim_{N\to\infty} h = 0$ and $\lim_{N\to\infty} Nh = \infty$ i.e. h reduces to 0 at a rate slower than

$1/N$.

$$\text{AMISE}(f(\hat{x})) = \frac{h^4}{4}(\mu_2(K))^2 R(f'') + \frac{1}{Nh}R(K) \tag{42}$$

The Equation (42) interprets that a small h increases estimation variance, whereas, a larger h increases estimation bias. An optimal h minimizes the total AMISE$(f(\hat{x}))$. So,

$$\begin{aligned} \frac{d}{dh}AMISE(f(\hat{x})) &= h^3(\mu_2(K))^2 R(f'') - \frac{1}{Nh^2}R(K) = 0 \\ \Rightarrow h_{AMISE} &= \left(\frac{R(k)}{\mu_2(K)^2 R(f'')N}\right)^{\frac{1}{5}} \end{aligned} \tag{43}$$

Thus, the optimal bandwidth parameter depends upon some of the kernel parameters, number of samples and the second derivative of the actual PDF being estimated.