The Pennsylvania State University
The Graduate School
Eberly College of Science

COMPARISON OF DIFFERENT DENSITY ESTIMATORS FOR INFINITE DIMENSIONAL EXPONENTIAL FAMILIES

A Thesis in
Statistics
by
Aniruddha Rajendra Rao

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

August 2019

The thesis of Aniruddha Rajendra Rao was reviewed and approved* by the following:

Bharath Sriperumbudur
Assistant Professor of Statistics
Thesis Advisor

Benjamin Shaby
Assistant Professor of Statistics

Ephraim Hanks
Professor of Statistics
Chair of Graduate Studies

*Signatures are on file in the Graduate School.

Abstract

In this thesis, we consider the problem of estimating an unknown density, p_o belonging to an infinite dimensional exponential family $\mathcal{P}$ parametrized by functions in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$. $\mathcal{P}$ is quite rich in the sense that a broad class of densities on $\mathbb{R}^d$ can be approximated arbitrarily well in Kullback-Leibler (KL) divergence by elements in it. The main focus of the thesis is to propose and compare the performance of various estimators of p_o. General methods like maximum likelihood estimation (MLE) or pseudo MLE do not result in practically useful estimators due to their inability to efficiently handle the log-partition function. In this work, we consider three different estimators, (i) Kernel Density Estimator (KDE), which is a classical non-parametric density estimator, (ii) Score Matching Estimator (SME), based on minimizing the Fisher divergence, $J(p_o \parallel p)$ between p_o and $p \in \mathcal{P}$, which involves solving a simple finite-dimensional linear system and (iii) Approximate Matching estimator (AME), which is a variation of SME but computationally more efficient. We show through numerical simulations that KDE performs better in the univariate case, while the other two methods have superior performance in high dimensional scenarios.

Table of Contents

List of Figures

List of Tables

List of Symbols

1 k denotes a positive definite kernel.

2 $J(p \parallel q)$ is the Fisher Divergence between densities p and q.

3 $||a|| = \sqrt{\sum_{i=1}^{d} a_i^2}$ and $\langle a, b \rangle = \sum_{i=1}^{d} a_i b_i$.

4 $C(\mathcal{X})$ is the space of all continuous function on $\mathcal{X}$.

5 $C_b(\mathcal{X})$ is the space of all bounded continuous function on $\mathcal{X}$.

6 $C_o(\mathcal{X})$ is the space of all continuous function on $\mathcal{X}$ that vanishes at infinity.

7 $M_b(\mathcal{X})$ is the set of all finite Borel measures on $\mathcal{X}$.

8 $L^r(\mathcal{X}, \mu)$ is the Banach space of r-power ($r \geq 1$) where μ is integrable function on $\mathcal{X}$.

9 Hellinger distance, $h(p, q) = ||\sqrt{p} - \sqrt{q}||_{L^2(\mathbb{R}^d)}$.

10 $||p - q||_{L^1(\mathbb{R}^d)}$ is the total variation (TV) distance between p and q.

11 Kullback-Leibler divergence, $KL(p \parallel q) = \int_{\mathbb{R}^d} p(x) \log \frac{p(x)}{q(x)} dx$.

Acknowledgments

This thesis would not have been completed without the help of my advisor, Dr. Sriperumbudur. I am heartily grateful for his patience, encouragement, guidance and supervision during the past several years. I would also want to thank my committee members Dr. Shaby and Dr. Hanks for their help in developing this document. Their comments brought great insight into helping improve this project and future projects to come.

Also, I would like to thank all the professors, staff, and colleagues in the Department of Statistics at Penn State for always being so supportive. Special mention to Dr. Slavković and Dr. Altman for helping and believing in me. A huge shout out to Dr. Buchanan for being an inspiration to us all. A big thank you to all my dear friends for being a part of this journey and beyond.

Finally, I would like to thank my parents Mamatha and Rajendra Rao, and my sister Anagha Rao for their unconditional love, support and care.

Chapter 1

Introduction

1.1 Exponential Family

The exponential family is a flexible family of distributions, with many common distributions such as the Bernoulli, Gaussian, Multinomial, Dirichlet, Gamma, Poisson, Beta belonging to it.

The exponential family of distributions can be expressed in a *natural form* as,

$$\wp = \{p_\theta(x) = q_o(x)\, e^{\theta^T T(x) - A(\theta)} : x \in \Omega, \theta \in \Theta \subset \mathbb{R}^m\} \tag{1.1}$$

where q_o is a probability density defined over $\Omega \subseteq \mathbb{R}^d$, $T : \Omega \to \mathbb{R}^m$ is the sufficient statistics, $A(\theta) = \log \int_\Omega e^{(\theta^T T(x))} q_o(x) dx$ is the cumulant generating function (or log-partition function), $\Theta \subset \mathbb{R}^m \cap \{\theta : A(\theta) < \infty\}$ is the natural parameter space and θ is the vector of natural parameter. This special form is chosen for mathematical convenience and generality, based on some useful algebraic properties.

In this thesis we consider an infinite dimensional generalization [1, 2] of the finite dimensional exponential family in the natural form,

$$\mathcal{P} = \{p_f(x) = e^{f(x) - A(f)}\, q_o(x) : x \in \Omega, f \in \mathcal{F}\} \tag{1.2}$$

where $\mathcal{F}$ is a function space defined as $\mathcal{F} = \{f \in \mathcal{H} : A(f) < \infty\}$,

$A(f) = \log \int_\Omega e^{f(x)} q_o(x)\, dx$, $\mathcal{H}$ is a reproducing kernel Hilbert space (RKHS) [3] with k as its reproducing kernel. We refer the readers to Sections 1.3 and 1.4 for more details on kernel functions and RKHS.

While there are various generalizations for different choices of $\mathcal{F}$, the connection of $\mathcal{P}$ to the natural exponential family in (1.1) is particularly enlightening when $\mathcal{H}$ is an RKHS. This is due to the reproducing property of the kernel, $f(x) = \langle f, k(x, \cdot)\rangle_{\mathcal{H}}$, through which $k(x, \cdot)$ takes the role of the sufficient statistic [4]. It can be shown that every $p \in \wp$ is generated by $\mathcal{P}$ induced by a finite dimensional RKHS $\mathcal{H}$, and therefore the family $\mathcal{P}$ with $\mathcal{H}$ being an infinite dimensional RKHS is a natural infinite dimensional generalization of $\wp$. Also, this generalization is specifically intriguing as in contrast to $\wp$, it can be shown that $\mathcal{P}$ is a rich class of densities (depending on the choice of k) that can approximate a broad class of probability densities arbitrarily well in KL divergence (refer to Theorem 10 in [4]). This generalization has created interest in statistical and machine learning applications in different areas like Bayesian non-parametric estimation [5], dimension reduction [2], Markov chain Monte-carlo [6], nonparametric hypothesis testing [7, 8] etc. Because of richness of $\mathcal{P}$, it is of interest to model densities by $\mathcal{P}$. Therefore the goal is to estimate unknown densities by elements in $\mathcal{P}$ when $\mathcal{H}$ is an infinite dimensional RKHS.

This formal setting is as follows : Given i.i.d. random samples $(X_a)_{a=1}^n$ drawn from an unknown density p_o, the goal is to estimate p_o through densities in $\mathcal{P}$. This is beneficial as $\mathcal{P}$ is a rich class of densities that can approximate a broad class of probability densities arbitrarily well, hence it may be widely used in place of non-parametric density estimation methods (e.g., kernel density estimation). We refer to the cases of $p_o \in \mathcal{P}$ as well-specified and $p_o \notin \mathcal{P}$ as misspecified. Through numerical simulations, we show in Chapter 3 that estimating p_o through $\mathcal{P}$ performs better than KDE in both well-specified and misspecified scenarios when the dimensionality is high.

1.2 Motivation

We know in the finite dimensional case, $\wp$ can be estimated using maximum likelihood (ML), which leads to solving elegant likelihood equations [9]. However, to solve infinite dimensional case, as in many non-parametric estimation methods, a simple extension of maximum likelihood estimation (MLE) suffers from the problem of ill-posedness. To overcome this bottleneck, Fukumizu [2] and Barron and Sheu [10] proposed methods but they suffered from many drawbacks that are both theoretical and computational in nature. Along with having problems with consistency and convergence rate, the methods are practically very difficult to construct even under strong assumptions.

In short, the maximum likelihood estimator (MLE) approach to learn $p_o \in \mathcal{P}$ results in estimators that are of limited practical interest. We can see this especially when one can treat the problem of estimating $p_o \in \mathcal{P}$ in a completely non-parametric fashion by using Kernel Density Estimation, which is well-studied and easy to implement. However, this method ignores the structure of $\mathcal{P}$ and is known to perform poorly even for moderate values of d. To address this Sriperumbudur et al. [4], proposed SME which we will discuss in Chapter 2. Based on SME, in this thesis we propose a computationally efficient version of SME which is also discussed in Chapter 2.Here, we will not concentrate on the ML approach but we will compare SME and AME with Kernel Density Estimation and show that it performs poorly as the dimension d increases.

1.3 Kernel Function

Before we move further, let us understand what kernel functions are. Generally speaking, any symmetric, positive definite function $k(x, y)$ can serve as a kernel function. Kernel functions are widely used in machine learning and data mining. One reason is due to the so-called property of "kernel trick" which allows you to compute the inner product in the higher-dimensional space without explicitly transforming

the vector into the higher-dimensional space they use no extra memory, and have a minimal effect on computation time. Hence, kernel functions can be regarded as a generalized inner-product. That is,

$k(x, y) = \langle \phi(x), \phi(y) \rangle$, for some $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is an inner-product space.

We call $\mathcal{X}$ as the input space, $\mathcal{H}$ the feature space and ϕ the feature map (associated with kernel functions k). In particular, if $\mathcal{X}$ is already an inner-product space, we may choose ϕ to be the identity map. The advantage of using a kernel function is that it allows constructing algorithms in the inner-product space $\mathcal{H}$. Kernel functions provide an elegant way of dealing with nonlinear algorithms by reducing them to linear ones in some feature space $\mathcal{H}$ nonlinearly related with the input space $\mathcal{X}$. For those algorithms relying on inner-products, instead of explicitly mapping the data with a feature map ϕ and taking the inner-product, one can take a kernel function and use it right away without knowing the exact form of ϕ, which can be difficult to obtain. This is the "Kernel Trick". This will be illustrated in the following example.

Check Figure 1.1, where we have a 2-D space which is clearly not linearly separable, so we transform it into 3-D space $(x_1^2, y_1^2, \sqrt{2}x_i x_j)$, where the data points are linearly separable.

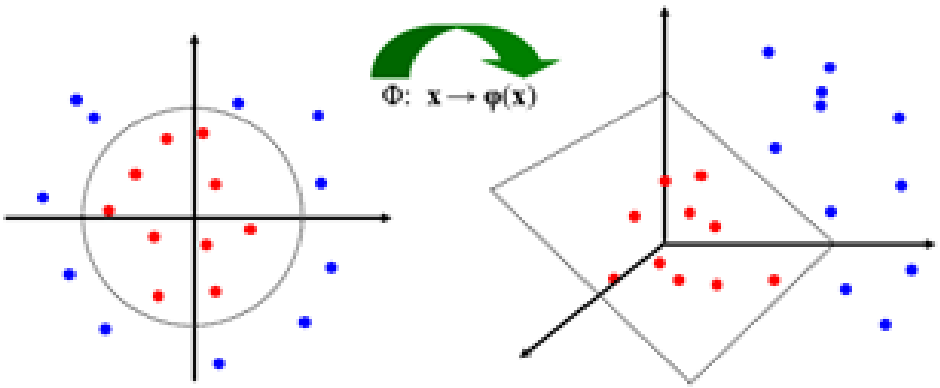

Figure 1.1. Example of a kernel function

One of the most common kernel functions is the Gaussian kernel function: $k(x_i, x_j) = e^{-||x_i - x_j||^2/(2\sigma^2)}$. It is very popular due to its universal approximation capability, desirable smoothness and numeric stability. The positive parameter σ^2 is called shape parameter or bandwidth in literature, and $\|\cdot\|$ is the Euclidean $L-$norm. The Gaussian kernel generates the function space spanned by radial basis functions (RBF). There are other kernel functions like polynomial kernel, triangular kernel, sigmoid kernel, Laplace kernel etc. In this thesis we have used Gaussian kernel for experiments.

1.4 RKHS

The feature space mentioned in Section 1.2 is also called reproducing kernel Hilbert spaces (RKHS). In this section, we will introduce the RKHS and explore the relationship between kernel function and RKHS.

A reproducing kernel Hilbert space (RKHS) is a Hilbert space of functions in which point evaluation is a continuous linear functional. This means that if two functions f and g in the RKHS are close in norm, i.e., $\|f - g\|$ is small, then f and g are also pointwise close, i.e., $|f(x) - g(x)|$ is small for all x. The converse need not be true ie any RKHS is associated with a kernel that reproduces every function in the space in the sense that for any x in the set on which the functions are defined, evaluation at x can be done by taking an inner product with a function determined by the kernel. This kind of a reproducing kernel exists if and only if every evaluation functional is continuous.

A Hilbert space which possesses a reproducing kernel is called a reproducing kernel Hilbert space. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive definite kernel if it is symmetric and if for any finite subset $\{x_i\}$, $i = 1, ..., n$ of $\mathcal{X}$ and any sequence of scalar coefficients $\{\alpha_i\}$, $i = 1, ..., n$ the following inequality holds: $\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, y_j) \geqslant 0$.

Theorem 1.1 (Moore-Aronszajn): Suppose k is a symmetric, positive definite kernel on a set $\mathcal{X}$. Then there is a unique Hilbert space of functions $\mathcal{H}$ on $\mathcal{X}$ for

which k is a reproducing kernel and vice versa.

Hence, if a function $k(z, z')$ of two arguments is symmetric and positive definite then it is a valid unique reproducing kernel.

1.5 Score Matching

Score matching method [11,12] helps us overcome the disadvantages of KDE and MLE. The SME was originally developed for densities on Euclidean space. The extension to Riemannian manifolds was sketched in [13, 14], but without a detailed analysis. Score matching is a parameter learning methodology that cleverly obviates the need to compute the log-partition function. Instead of minimizing the KL divergence the score matching minimize the Fisher divergence between two continuously differentiable densities, p and q on an open set $\Omega \subseteq \mathbb{R}^d$, given as

$$J(p \parallel q) = \frac{1}{2} \int_{\Omega} p(x) \parallel \bigtriangledown \log\ p(x) - \bigtriangledown \log\ q(x) \parallel_2^2 dx \tag{1.3}$$

where $\bigtriangledown \log\ p(x) = (\partial_1 \log\ p(x), \partial_2 \log\ p(x), ..., \partial_d \log\ p(x))$ with $\partial_i \log\ p(x) := \frac{\partial}{\partial x_i} \log\ p(x)$.

KL divergence and Fisher divergence are closely related through de Bruijn's identity. Convergence in Fisher divergence is a stronger form of convergence than that in KL, total variation and Hellinger distances [15, 16]. Other similarities to the KL divergence include that the Fisher divergence is non-negative and is zero if and only if $p = q$ (a.e.), yet it is not symmetric and does not form a distance metric. It is not hard to see that in score matching, there is no need to use the log-partition function, in other words, it can work directly with un-normalized models. A minor limitation is that it requires the underlying distributions to have sufficiently smooth probability densities. MLE would typically require a functional form of the

log-partition function which is approximated through numerical integration at every step of an iterative optimization algorithm, thus leading to major computational savings in score matching where numerical integration is needed only once.

The term "score" has several distinct connotations in estimation. (a) Conventionally, the "score" refers to a derivative of the log likelihood with respect to parameters; it has close connections to maximum likelihood estimation. (b) However, in the context of the score matching estimator, the "score" refers to the derivative of the log likelihood with respect to the state variable x. (c) In addition, the term "scoring rule" [17] refers to a more general function of x and a distribution. Each scoring rule determines a divergence, and the minimization of the divergence leads to an estimator. A scoring rule is different from the scores in (a) and (b), though suitable choices for scoring rules lead to both maximum likelihood and SME.

Chapter 2

Methods

In this chapter we will go through Kernel Density Estimation, Score Matching Estimation and Approximate Matching Estimation in detail.

2.1 Kernel Density Estimation

Kernel density estimation (KDE) is a non-parametric approach to estimate the probability density function of a random variable. Kernel density estimation is a fundamental data smoothing problem where inferences about the population are made, based on the data. Kernel density estimators estimate $p(x)$ by constructing a neighbourhood around the point of interest x. Observations within this neighbourhood are then assigned a mass based on their distance from x via a kernel function, resulting in a smooth estimate. KDE can be useful if you want to visualize just the "shape" of some data, as a kind of continuous replacement for the discrete histogram. In other words, it can be viewed as a generalization of histogram density estimation with improved statistical properties and not having it's drawbacks like non-smooth estimates.

2.1.1 Univariate Case

Let $X_1, X_2, ..., X_n$ be a univariate independent and identically distributed (iid) sample drawn from some distribution with an unknown density p_o, where the goal is to estimate p_o The kernel density estimator is given by,

$$\hat{p_n}(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\Big(\frac{x - x_i}{h}\Big), \qquad x \in \mathbb{R} \tag{2.1}$$

where K is the kernel function (non negative function) and $h > 0$ is a smoothing parameter called the bandwidth. A kernel function with subscript h is called the scaled kernel function and is defined as $K_h(x) = \frac{1}{h}K\big(\frac{x}{h}\big)$. Hence we would want the value of h as small as the data allows; however, there is always a trade-off between the bias of the estimator and its variance.

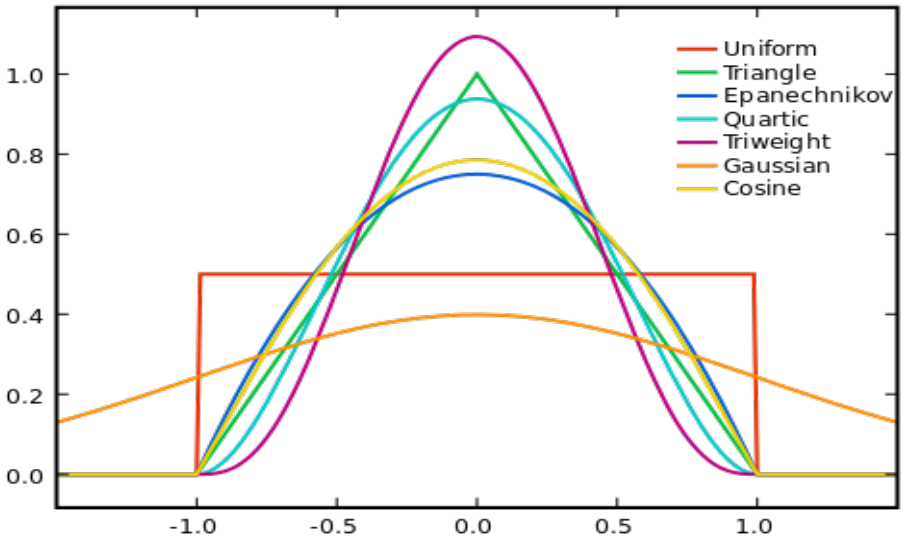

Figure 2.1. Different kernels

Some of the popular kernel functions include: uniform, triangular, epanechnikov, normal, and others, which are depicted in figure 2.1. Since the Gaussian (normal) kernel function has convenient mathematical properties, it is often used and so is the case in this project. The normal kernel is second-order kernel, suggesting it is a

proper, symmetric density function. One drawback of the Gaussian kernel is that its support runs over the entire real line; occasionally it is desirable that a kernel have compact support. Selection of the bandwidth is very important. If Gaussian basis functions are used to approximate univariate data, and the underlying density being estimated is Gaussian, the optimal choice for h is given by,

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \approx 1.06\hat{\sigma}n^{-1/5}$$

where $\hat{\sigma}$ is the standard deviation of the samples. We will see in Chapter 3 that when using this value of h, it does not work well in practice and using a grid search gives a much better result. Changing the bandwidth changes the shape of the kernel: a lower bandwidth means only points very close to the current position are given any weight, which leads to the estimate looking squiggly; a higher bandwidth means a shallow kernel where distant points can contribute. For a fixed h, kernel density estimates are not consistent. However, if the bandwidth decreases with sample size at an appropriate rate, then they are, regardless of which kernel is used. Therefore, choice of bandwidth has a larger impact on estimation quality than the choice of kernel. Bandwidth selection for kernel density estimation of heavy-tailed distributions is known to be relatively difficult.

In Figure 2.2 below, we see how different bandwidths for the Gaussian kernel work in estimating the density of the data. The thick black line represents the optimal bandwidth, $h = 0.344$. The jagged dotted line is the estimate of $p_o(x)$ when the bandwidth is halved, $h = 0.172$. The flatter, bell-shaped curve represents $h = 1.5$ which clearly oversmooths the data.

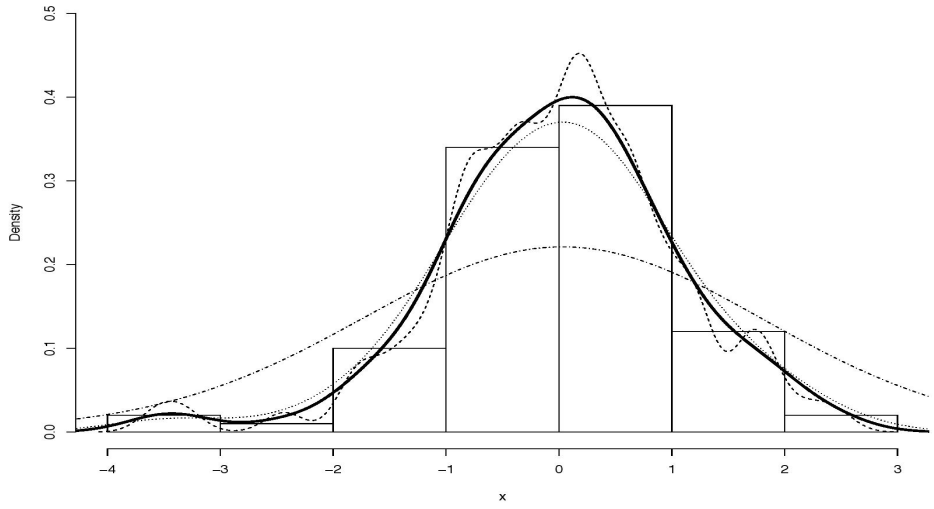

Figure 2.2. Kernel density estimates for various bandwidths

2.1.2 Multivariate Case

Let $X_1, X_2, ..., X_n$ be a sample of d-variate random vectors drawn from a common distribution described by the density function p. The kernel density estimate of p is defined as,

$$\hat{p_n}(x) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(x - x_i), \qquad x \in \mathbb{R}^d \tag{2.2}$$

where $\mathbf{H}$ is the bandwidth (or smoothing) matrix of size $d \times d$ which is symmetric and positive definite; K is the kernel function which is a symmetric multivariate density;

$$K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}\mathbf{x}).$$

The KDE accuracy does not depend on the choice of kernel, so we use the standard multivariate normal kernel in our experiments, which is given as,

$$K_{\mathbf{H}}(\mathbf{x}) = (2\pi)^{-d/2}|\mathbf{H}|^{-1/2} e^{-\frac{1}{2}\mathbf{x^T}\mathbf{H^{-1}}\mathbf{x}},$$

where $\mathbf{H}$ plays the role of the covariance matrix. But the choice of the bandwidth

matrix $\mathbf{H}$ is a crucial factor affecting the accuracy since it controls the amount and orientation of smoothing induced. A thumb of rule for value for $\mathbf{H}$ is, $\mathbf{H}_{ii}^{0.5}=n^{\frac{-1}{d+4}}\sigma_i$, where σ_i is the standard deviation of the i^{th} variable and all the other entries in $\mathbf{H}$ are zero i.e. the non diagonal elements are zero. But if the true density p is very different from the normal then this can result in an over-smoothed density. Similiar to the univariate case, the formula for calculating $\mathbf{H}$ did not help much and again defining $\mathbf{H} = a \cdot I$, where a is a constant and I is identity matrix, turned out to give a better result. We used grid search to find the best value of a. KDE suffers from the curse of dimensionality.

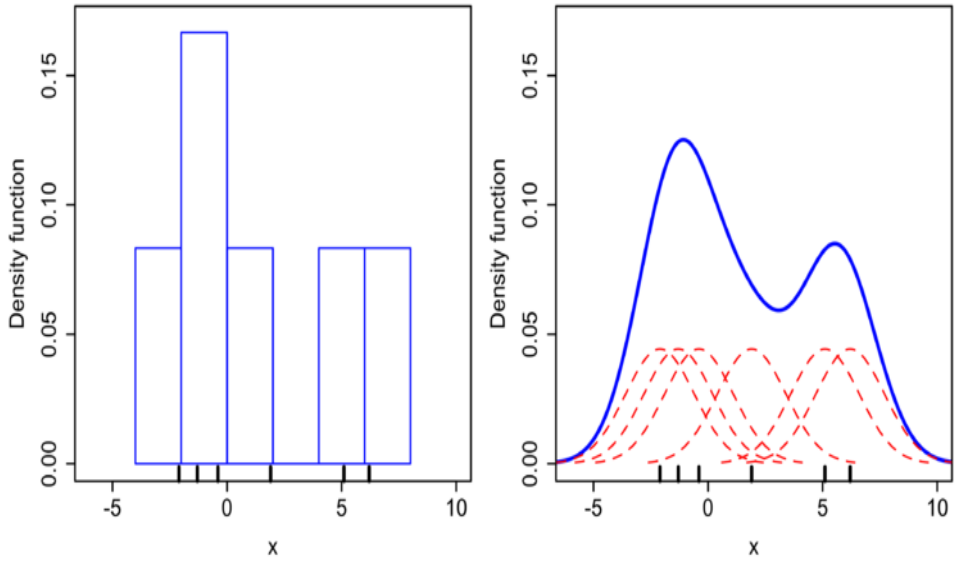

Figure 2.3. Histogram and kernel density estimator

In Figure 2.3, we compare the histogram (left) and kernel density estimate (right) constructed using data=(-2.1,-1.3,-0.4,1.9,5.1,6.2). The 6 individual kernels are the red dashed curves, with the kernel density estimate being the blue curves. For KDE, a normal kernel with variance 2.25 (indicated by the red dashed lines) is placed on each of the data points x_i. The kernels are summed to make the kernel density estimate (solid blue curve). The smoothness of KDE is evident compared to the discreteness of the histogram, as KDE converges faster to the true underlying density for continuous

random variables.

2.2 Score Matching Estimator

Score Matching Estimator (SME) involves choosing the minimizer of the (empirical) Fisher divergence between the unknown density p_o ($:= p_{f_o} \in \mathcal{P}$) and $p_f \in \mathcal{P}$ as the estimator where n iid random samples $(X_a)_{a=1}^n$ are drawn from p_o. The estimator is given as $\hat{p}$ where $\hat{p}$ is obtained by solving a simple finite dimensional linear system as shown in equation 2.3 below (refer to Theorem 5 in [4]). Note that the MLE is infeasible in practice due to the difficulty in handling $A(f)$ but SME is not. The consistency and convergence rates of $\hat{f} \in \mathcal{F}$ and $p_{\hat{f}}$ are given in [4]. Some assumptions on p_o, q_o and $\mathcal{H}$ related to the method is as follows,

- Ω is a non-empty open subset of $\mathbb{R}^d$ with piecewise smooth boundary $\partial\Omega := \bar{\Omega} \setminus \Omega$, where $\bar{\Omega}$ denotes the closure of Ω.
- p_0 is continuously extendible to $\bar{\Omega}$. k is twice continuously differentiable on $\Omega \times \Omega$ with continuous extension of $\partial^{\alpha,\alpha}k$ to $\bar{\Omega} \times \bar{\Omega}$ for $|\alpha| \leq 2$.
- $\partial_i\partial_{i+d}k(x,x)p_o(x) = 0$ for $x \in \partial\Omega$ and $\sqrt{\partial_i\partial_{i+d}k(x,x)}p_o(x)$=o($||x||_2^{1-d}$) as $x \in \Omega$, $||x||_2 \to \infty$ for all i $\in [d]$ where $[d]$:= $\{1, 2, ...d\}$ and k is a kernel function.

With the help of these assumptions, the problem of estimating p_o through the minimization of Fisher divergence gets simplified to the problem of estimating f_o using weighted least squares minimization in $\mathcal{H}$. This leads to the minimization of the regularized empirical weighted least squares to obtain an estimator $f_{\lambda,n}$ of f_o, which is then used to compute the estimate $p_{f_{\lambda,n}}$ of p_o, where $\lambda > 0$ is the penalty. Theorem 4 and 5 in [4] leads to SME and is given as,

$$f_{\lambda,n}(x) = -\frac{\hat{\xi}(x)}{\lambda} + \sum_{a=1}^{n}\sum_{i=1}^{d} \beta_{(a-1)d+i}\ \partial_i k(X_a, x), \tag{2.3}$$

where

$$\hat{\xi}(x) = \frac{1}{n}\sum_{a=1}^{n}\sum_{i=1}^{d}[\partial_i k(X_a, x)\ \partial_i \log q(X_a) + \partial_i^2 k(X_a, x)],$$

and

$$\beta = (\beta_{(a-1)d+i})_{a,i}$$

β is obtained by solving,

$$(G + n\lambda I)\beta = \frac{1}{\lambda}h,$$

where

$$(G)_{(a-1)d+i,(b-1)d+j} = \partial_i \partial_{j+d} k(X_a, X_b)$$

and

$$(h)_{(a-1)d+i} = \frac{1}{n}\sum_{a=1}^{n}\sum_{i=1}^{d}[\partial_i \partial_{j+d}^2 k(X_a, X_b) + \partial_i \partial_{j+d} k(X_a, X_b)\partial_j \log q_0(X - b)]$$

From (1.2) and (2.3) we get,

$$p_{f_{\lambda,n}}(x) = \frac{e^{(f_{\lambda,n}(x))} q_0(x)}{\int e^{(f_{\lambda,n}(x))} q_0(x)\, dx}. \tag{2.4}$$

It should be noted that G is a $nd \times nd$ matrix and h is a $nd \times 1$ matrix. Even though $f_{\lambda,n}$ involves solving simple linear system, it can be computationally taxing when n and d are large. On the other hand, MLE can be intractable due to the difficulty in handling the log-partition function but is statistically well-understood, with consistency and convergence rates established in general for the problem of density estimation [18]. However, for SME, computationally efficient methods exist to solve large linear systems and details on the consistency and convergence rates of SME under some smoothness conditions on f_o can be found in [4].

2.3 Approximate Matching Estimator

Approximate Matching Estimator (AME) involves n iid random samples $(X_a)_{a=1}^n$ that are drawn from p_o and $\hat{f}$ is obtained by solving a simpler finite dimensional linear system than that of SME as shown in Equation (2.5) below. AME is feasible in practice as it is a linear combination of kernel functions in RKHS and therefore easier to solve compared to SME. The consistency and convergence rates of AME are yet to be established.

The AME is of the form,

$$\tilde{f}_{\lambda,n}(x) = \sum_{a=1}^{n} \alpha_a k(x, X_a) \tag{2.5}$$

where

$$\hat{\alpha} = \arg\min_{\alpha} \ \alpha^T A \alpha + 2\alpha^T b,$$

and

$$A_{ij} = \frac{1}{n} \sum_{a=1}^{n} \sum_{l=1}^{d} \ \partial_l k(X_a, x_i) \ \partial_l k(X_a, x_j),$$

with

$$b_i = \frac{1}{n} \sum_{a=1}^{n} \sum_{l=1}^{d} [\partial_l^2 k(X_a, x_i) + \partial_l \log q_o(X_a) \partial_l k(X_a, x_i)].$$

This means $\hat{\alpha}_{min} = -A^{-1}b$.

From (1.2) and (2.5) we get,

$$p_{\tilde{f}_{\lambda,n}}(x) = \frac{e^{(\tilde{f}_{\lambda,n}(x))} q_o(x)}{\int e^{(\tilde{f}_{\lambda,n}(x))} q_o(x)\, dx} \tag{2.6}$$

It is clear that the computational cost of SME is $O(n^3 d^3)$ whereas for AME it's $O(n^3)$.

Chapter 3

Simulation and Results

In this chapter, we discuss the simulation setting for density estimation in univariate and multivariate scenarios. We will see the results for each scenario at different simulation settings. In both the scenarios the performance metric used are the score objective function and the correlation (see Equations 3.1 and 3.2).

As mentioned before, computation cost for KDE is $O(n)$ where as for the SME and AME it is $O(n^3d^3)$ and $O(n^3)$ respectively. Therefore, when compared to KDE other two methods look computationally costly, and hence raises questions about the applicability of these other two methods. In the simulations below, we can clearly see that as d increases, SME and AME perform much better than KDE and the gap between the methods increases as d grows. This is because KDE performs poorly even for moderate d [19]. We don't consider MLE since it does not give practically feasible estimators [20].

3.1 Univariate scenario

We check the performance of these three methods under univariate scenario to study if there is any difference in the methods when the dimension is not a factor. In the univariate scenario we solve the problem for estimating different distributions using the base measure of the exponential family q_o as a normal distribution $N(0, 100)$ with kernel being Gaussian, which is defined as $k(x, y) = e^{-s*||x-y||^2}$ (s is the tuning

parameter). We estimate standard normal, exponential, Cauchy and mixture of normal distribution in this scenario, which are defined as follows:

Normal Distribution: $X \sim N(\mu, \sigma^2)$ *where* $\mu \in \mathbb{R}$ and $\sigma^2 > 0$.

$$p_o(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{(x-\mu)^2}{\sigma^2}\right)}, \quad x \in \mathbb{R}$$

Exponential Distribution: $X \sim exp(\lambda)$ *where* $\lambda > 0$

$$p(x) = \lambda e^{(-\lambda x)}, \quad x \in \mathbb{R}$$

It can be shown that Normal and Exponential distribution belong to P.

Cauchy Distribution: $X \sim cauchy(\mu, \gamma)$ *where* $\mu \in \mathbb{R}$, and $\gamma > 0$

$$p(x) = \frac{1}{\pi\gamma \left[1 + \left(\frac{x-\mu}{\gamma}\right)^2\right]}, \quad x \in \mathbb{R}$$

Cauchy distribution does not belong to $\mathcal{P}$ i.e. it is a misspeicifed case. For simulation, the parameter values are $\mu = 0$ and $\gamma = 2$

Mixture of Normal distribution:

$X \sim \alpha N(\mu_1 = 0, \sigma_1 = 1) + (1 - \alpha) N(\mu_2 = 4, \sigma_2 = 2)$

$$p(x) = \alpha\left(\frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{1}{2}\left(\frac{(x-\mu_1)^2}{\sigma_1^2}\right)}\right) + (1-\alpha)\left(\frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{1}{2}\left(\frac{(x-\mu_2)^2}{\sigma_2^2}\right)}\right), \quad x \in \mathbb{R}$$

It turns out mixture of normal does not belong to $\mathcal{P}$.Hence, this is also a misspec-

ified case.

The tuning parameters s (bandwidth) and λ (penalty) are selected using 5-fold cross-validation (CV) where we do a grid search over values of s and λ. The range of values for bandwidth parameter s is (0.1,2) with a step of 0.2 and regularization parameter λ is (0,1) with a step of 0.1 but in some cases we have gone out of the range if it was possible to get a much better performance. Since the distribution used are univariate, we can plot the densities to get an idea of the performance of the estimators. We chose n ranging from 100 to 1000.

We check the performance of the methods by, the objective function for the score matching method and the correlation of the estimator with the true density function, which are defined below.

$$\hat{J}(p) = \int_{\Omega} \left(\frac{1}{2} |\partial \log p(x)|^2 + \partial \log p(x) \right) p_o(x) \, dx; \tag{3.1}$$

$$cor(p, p_o) := \frac{E_R[p(X)p_o[(X)]]}{\sqrt{E_R[p(X)^2]E_R[p_o(X)^2]}}, \tag{3.2}$$

where R is a probability distribution and for R, we use empirical distribution based on 10000 random samples drawn iid from p_o(x). Monte-carlo method is used to get good estimates of the above two equations. Lower the $\hat{J}(p)$ and higher the correlation, more closer the estimator is to the true density.

The Table below gives the 5-fold CV score of all the distributions under each of the methods. In the table, KDE_{h^*} indicates that the value of h was taken as the recommended value, i.e. $h \approx 1.06\hat{\sigma}n^{-1/5}$. But, as we can see from Table 3.1 that KDE_{h^*} performs the worst in general. In most of the cases AME performs better than the KDE and SME, can be seen this from Table 3.1 and Figure 3.1. Though in the Figure 3.1, it is difficult to say which is the best method between AME and

KDE in the univariate scenario, we can see better distinction between them when we compare the score function objective and correlation.

Table 3.1. Cross-Validation error for different methods at $n = 500$

Distribution	KDE	KDE_{h^*}	SME	AME
Exponential	**0.06145074**	0.1448237	0.306359	0.08188853
Normal	0.0001884844	0.00019400	0.00491352	**8.593506e-05**
Cauchy	6.972625e-04	0.00708302	0.00053151	**3.226371e-05**
Mixture of Normal	0.0003089972	0.00439203	0.00168175	**0.000180419**

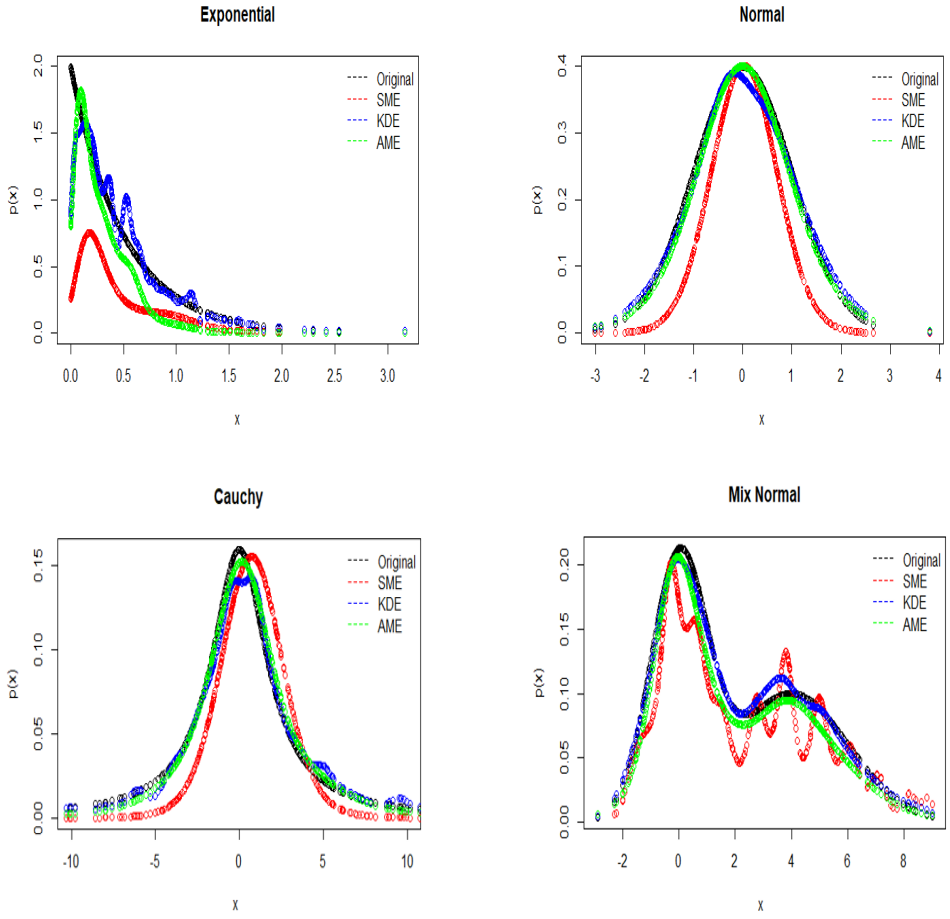

Figure 3.1. Estimated densities at $n = 500$

Figure 3.1 is a plot of estimated densities at $n = 500$ for each of the methods, where the true sample is denoted in black. From the plot is it clear that SME does not perform as well as the others.

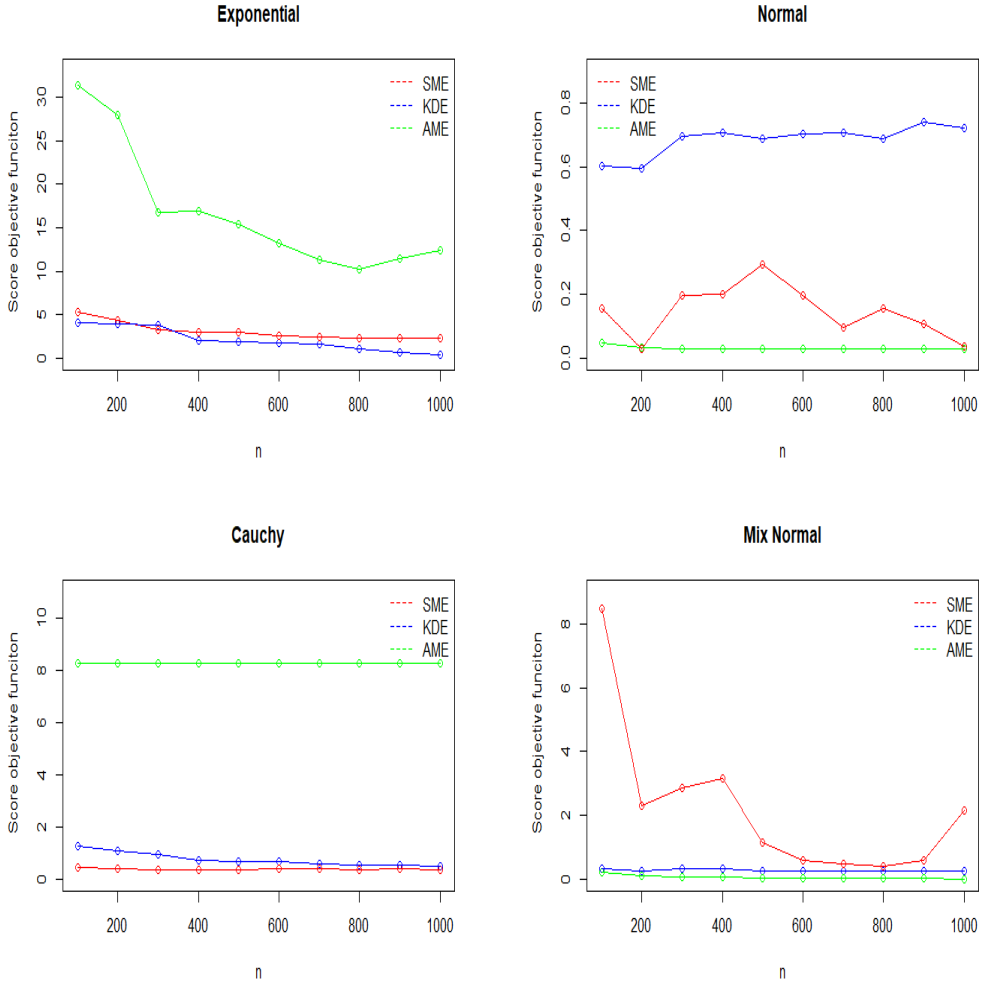

Figure 3.2. Score objective function for different distributions.

Figure 3.2 above and Figure 3.3 below, we can see that, KDE does really well irrespective of the distribution being used. AME does better than SME when correlation is used as the objective. An anomaly is the score objective graph of normal distribution where KDE is not performing well.

Note: As n increases we have more information and hence the score function should gradually decrease and the correlation should increase. This behaviour is mostly observed in all the graphs.

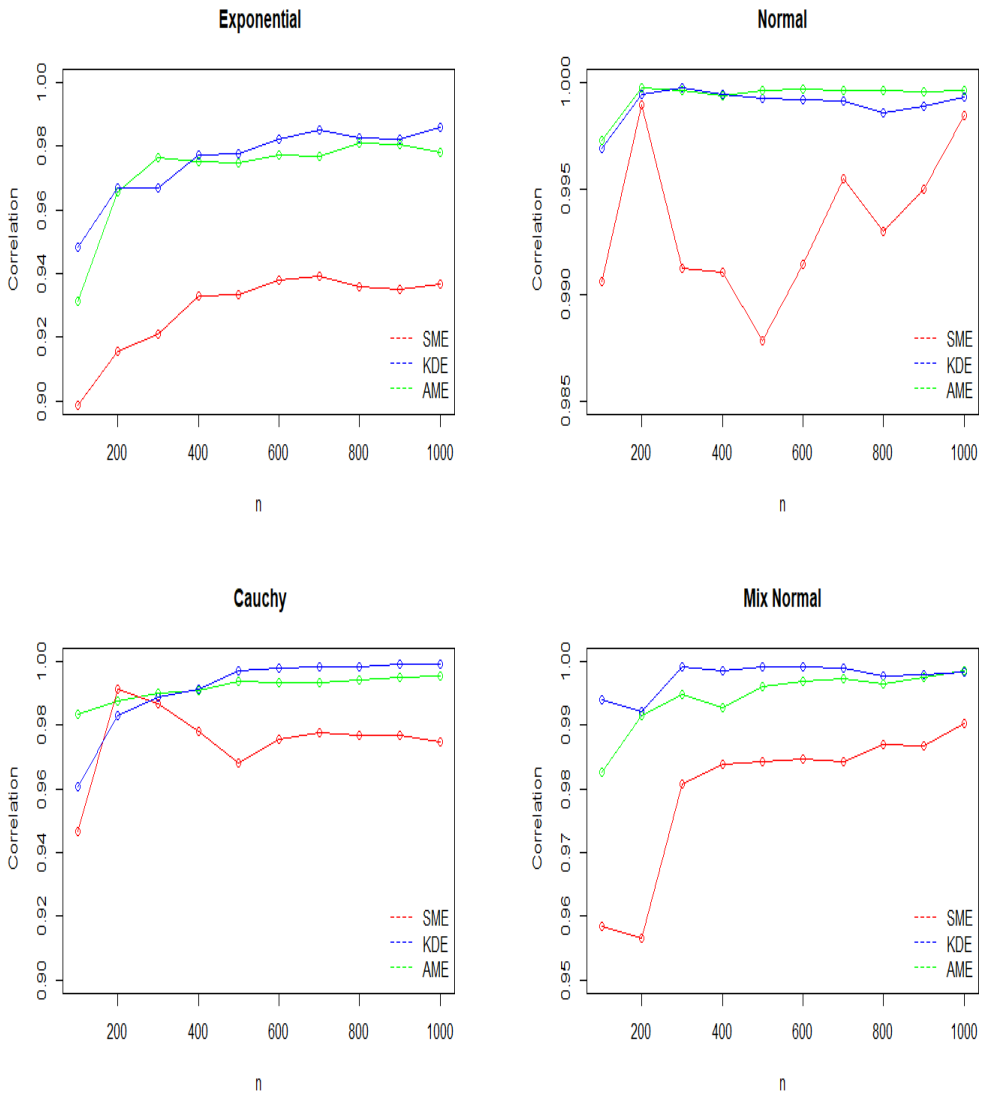

Figure 3.3. Correlation plot for different distributions.

In Figure 3.4 we can see that KDE is the fastest among the three methods, which

makes sense since computation cost for Kernel Density Estimation is $O(n)$ where as for SME is $O(n^3d^3)$ and for AME it is also $O(n^3)$. Even though SME is faster in the univariate scenario compared to AME, later we will see that this changes in the multivariate scenario.

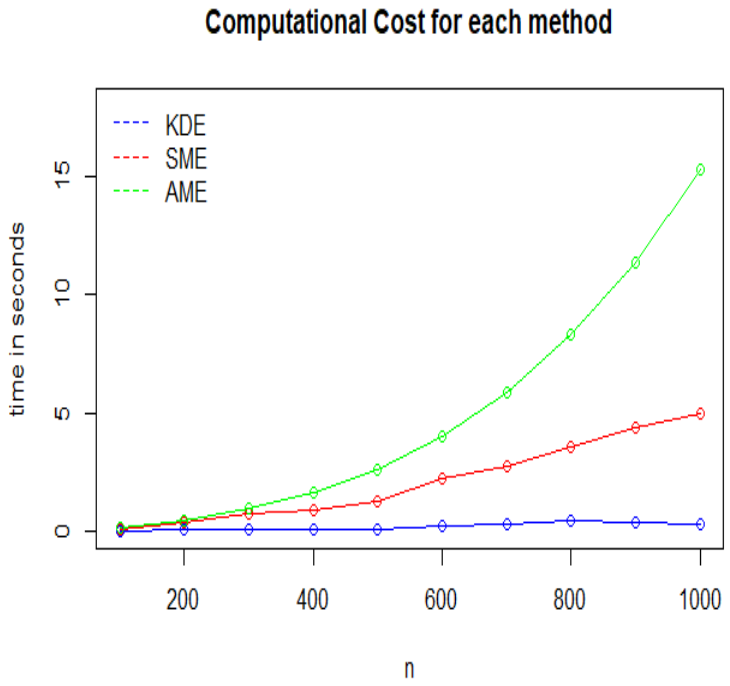

Figure 3.4. Computational time for different methods.

3.2 Multivariate scenario

We consider two simulation settings in the multivariate scenario, one for estimating a multivariate normal and another for estimating a mixture of Gaussians. Multivariate normal belongs to the exponential family and hence is a well-specified case whereas the mixture of Gaussians is not. We check the performance of all the methods in both the cases as we increase the value of d. We refer readers to [6, 21] to see the superiority of working with the infinite dimensional family using SME.

For the multivariate normal, we just solve the problem of estimating a standard multivariate normal distribution, $N(0, I_d)$ on $\mathbb{R}^d$. We use a Gaussian kernel $= e^{\left(-s*||x-y||_2^2\right)}$. For the mixture of Gaussians case,:

$$p_o(x) = \frac{1}{2}N(\alpha, I_d) + \frac{1}{2}N(\beta, I_d)$$

where $I_d \in \mathbb{R}^d$, α=2 and β=5.

The base measure q_o is chosen to be multivariate normal distribution $N(0,100I_d)$ for both cases and again, the values for bandwidth parameter s and regularization parameter λ are found using a grid search, by minimizing the 5-fold CV error. We take a range of values for s and λ as in the univariate scenario and also in some cases we have gone out of the range if it was possible to get a much better performance. We increase n from 100 to 1000 and d from 1 to 20.

Again, we check how good the methods are using the objective function for the score matching method (3.3) and the correlation of the estimator with the true density function (3.2).

$$\hat{J}(p) = \sum_{i=1}^{d} \int_{\Omega} \left(\frac{1}{2} |\partial_i \log\ p(x)|^2 + \partial_i^2 \log\ p(x) \right) p_o(x)\ dx \tag{3.3}$$

We can see from Figure 3.5 that both the methods perform better than KDE for both fixed d, varying n and fixed n, varying d. But there is not much difference between SME and AME. Similar conclusion can be draw even from the correlation plots in Figure 3.6, though it seems like AME performs slightly better in case of the multivariate normal. We can see that SME and AME have an advantage over KDE as the dimensionality increases.

Note: As n increases we have more information and hence the Score objective function should gradually decrease and the correlation should increase. We could

expect the inverse behaviour when d increases.

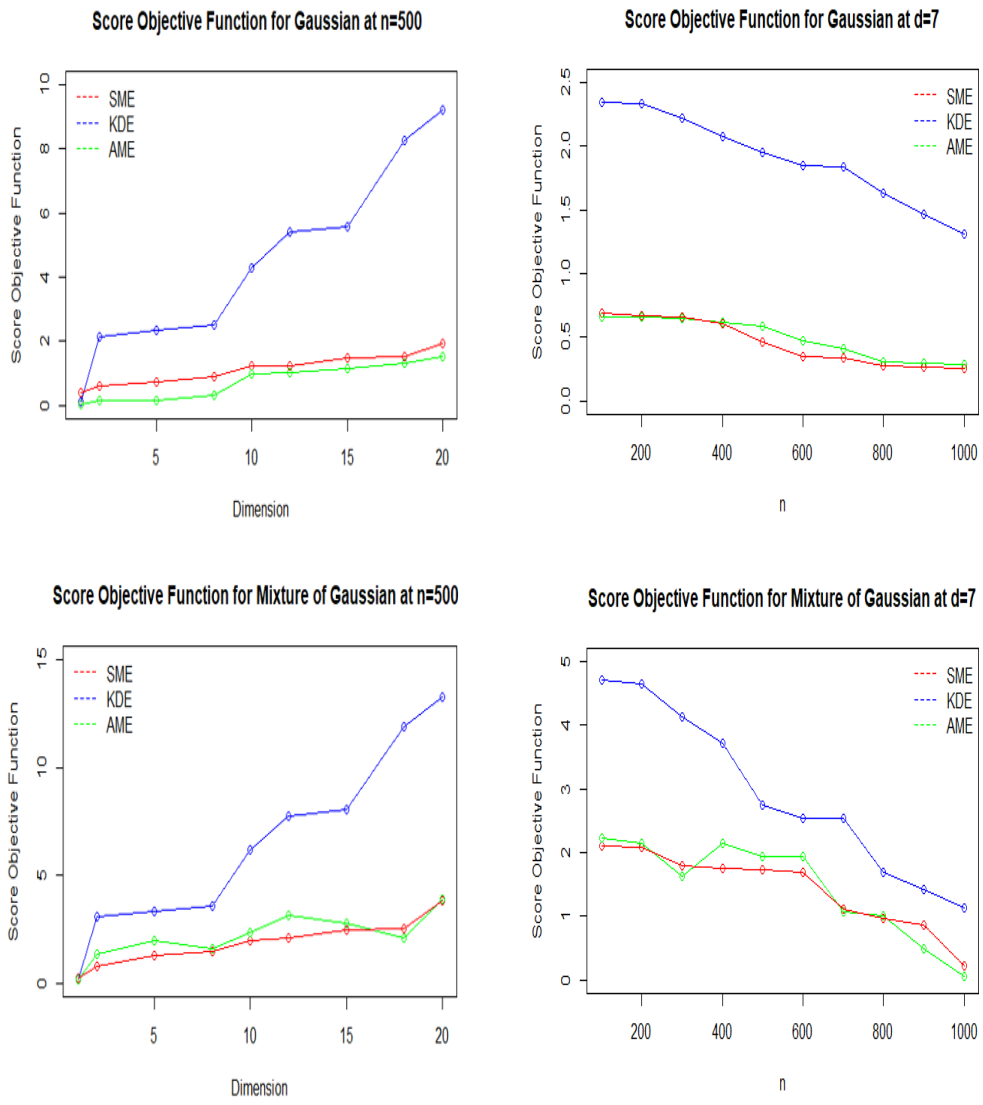

Figure 3.5. Score objective function for different cases.

In Figure 3.7 below, we can see that KDE is still the fastest among the methods since computation cost for KDE is $O(n)$ where as for the SME is $O(n^3d^3)$ and for AME it is $O(n^3)$. The gap has increased compared to the univarite scenario. But we can see that in the multivariate scenario the AME is much faster than SME.

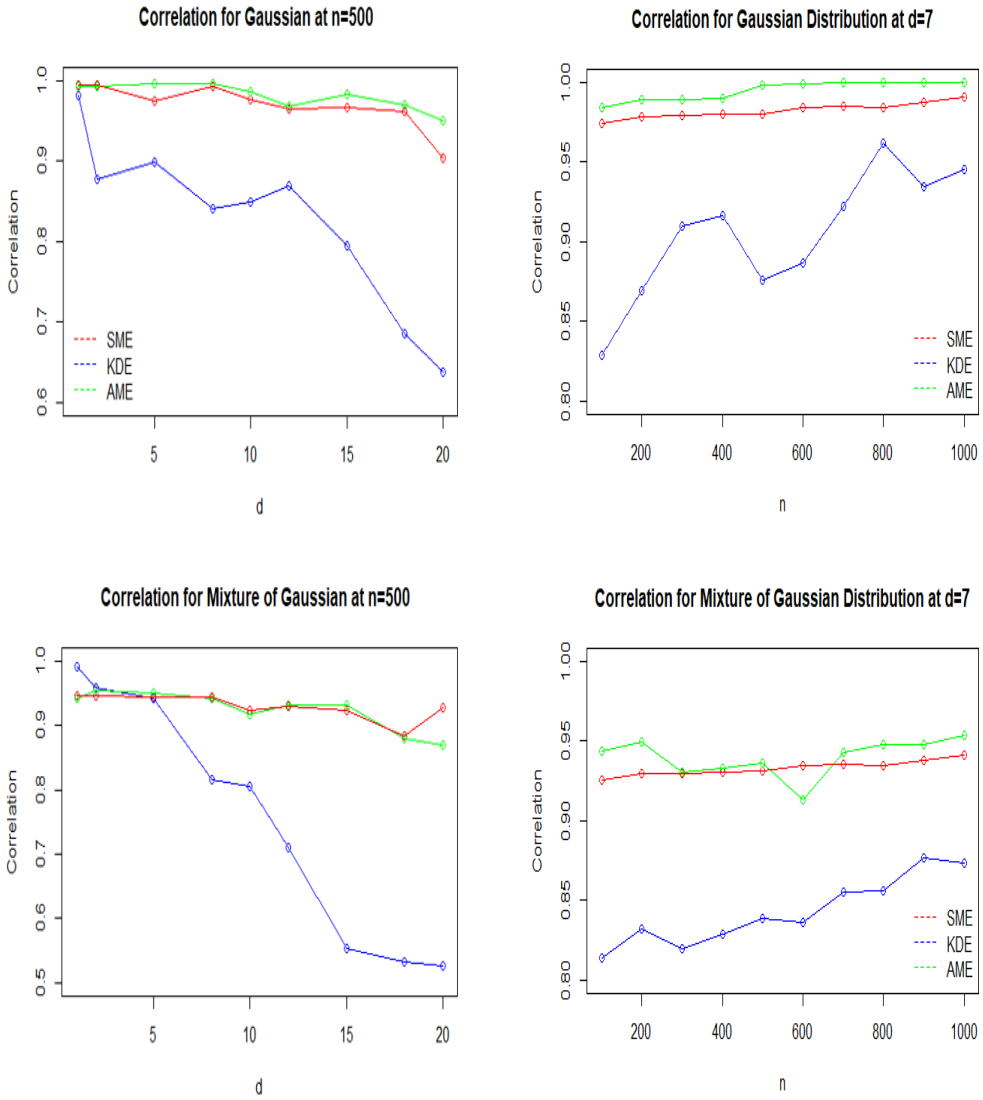

Figure 3.6. Correlation plot for different cases.

It is clear that we are considering simple scenarios of estimating some univariate distribution, multivariate Normal and mixture of Gaussian using AME and SME to demonstrate their superior performance over KDE as the dimension, d increases. But the goal here is not to construct density estimators that improves upon KDE but to provide a modeling technique of approximating an unknown density by a

rich parametric family of densities with the parameter being infinite dimensional in contrast to the classical approach of finite dimensional approximation. Basically a method which can still provide good results in cases where KDE fails, especially when d is large. We can see from Figure 3.7 that KDE still has the least computationally cost but AME is considerably faster than SME as the dimensionality increases.

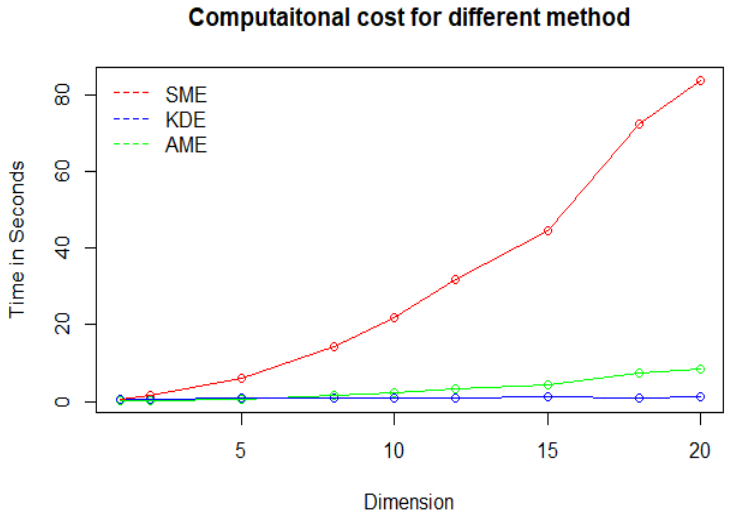

Figure 3.7. Computational time for different methods.

Chapter 4

Conclusion and Future work

Based on our study, we can conclude that Score matching estimator (SME) and Approximate matching estimator (AME) provide computationally efficient alternatives to maximum likelihood based estimators, which suffer from the computational intractability of the log-partition function and also empirically outperforms the kernel density estimator, especially in the high dimensional scenarios. It is important to note that SME and AME perform well in well-specified and misspecified cases. Even though we have used normal as the base measure, we noted in our experiments that the estimators perform well if the base measure is changed. Hence, the methods are quiet robust.

There is still space for further improvement and discoveries. First, since these estimators are computationally expensive compared to KDE, it is important to study either alternate estimators or improve implementations of the current estimators for better applicability. The consistency and convergence rates for the AME is still open and is an interesting question to consider in the future.

Appendix | Calculations

A kernel k is said to be m-times continuously differentiable if $\partial^{\alpha,\alpha}k : \mathcal{X} \times \mathcal{X} \longrightarrow \mathbb{R}$ exists and is continuous for all $\alpha \in \mathbb{N}_o^d$ with $|\alpha| \leq m$ where $\partial^{\alpha,\alpha} := \partial_1^{\alpha_1}...\partial_d^{\alpha_d}\partial_{d+1}^{\alpha_{d+1}}...\partial_{2d}^{\alpha_{2d}}$. If $\partial^{\alpha,\alpha}k$ exists and is continuous, then $\partial^{\alpha}k(x,\cdot) = \partial_1^{\alpha_1}...\partial_d^{\alpha_d}k(x,\cdot) = \frac{\partial^{|\alpha|}}{\partial_1^{\alpha_1}...\partial_d}k((x_1,...,x_d),\cdot) \in \mathcal{H}_k$ with $x = (x_1,...,x_d)$ and for every $f \in \mathcal{H}_k$, we have $\partial^{\alpha}f(x) = \langle \partial^{\alpha}k(x,\cdot), f\rangle_{\mathcal{H}_k}$ and $\partial^{\alpha,\alpha}k(x,x') = \langle \partial^{\alpha}k(x,\cdot), \langle \partial^{\alpha}k(x',\cdot)\rangle_{\mathcal{H}_k}$.

Below is a list of calculations used in SME and AME under the case where the kernel is Gaussian (of the form $e^{(-s\|x-y\|^2)}$ with $s > 0$) :

- $k(x,y) = e^{(-s\|x-y\|_2^2)} = e^{(-s\sum_{i=1}^{d}(x-y)^2)}$
- $\partial_j k(x,y) = -2 \cdot s(x_j - y_j) \cdot e^{(-s\sum_{i=1}^{d}(x_i-y_i)^2)}$ and
  $\partial_j^2 k(x,y) = 2 \cdot s \cdot e^{(-s\sum_{i=1}^{d}(x_i-y_i)^2)} \cdot (2 \cdot s \cdot (x_j - y_j)^2 - 1)$
- $\partial_j \partial_{j+d} k(x,y) = 2 \cdot s \cdot e^{(-s\sum_{i=1}^{d}(x_i-y_i)^2)} \cdot (1 - 2 \cdot s \cdot (x_j - y_j)^2)$ and
  $\partial_j \partial_{l+d} k(x,y) = -4 \cdot s^2 \cdot e^{(-s\sum_{i=1}^{d}(x_i-y_i)^2)}(x_j - y_j)(x_l - y_l)$
- $\partial_j \partial_{j+d}^2 k(x,y) = 4 \cdot s^2(x_j - y_j) \cdot e^{(-s\sum_{i=1}^{d}(x_i-y_i)^2)} \cdot (3 - 2 \cdot s \cdot (x_j - y_j)^2)$ and
  $\partial_j \partial_{l+d}^2 k(x,y) = 4 \cdot s^2(x_j - y_j) \cdot e^{(-s\sum_{i=1}^{d}(x_i-y_i)^2)} \cdot (3 - 2 \cdot s \cdot (x_l - y_l)^2)$

- $\partial_j^2 \partial_{j+d} k(x,y) = 4 \cdot s^2 (x_j - y_j) \cdot e^{(-s \sum_{i=1}^{d} (x_i - y_i)^2)} \cdot (2 \cdot s \cdot (x_j - y_j)^2 - 3)$ and
  $\partial_j^2 \partial_{l+d} k(x,y) = 4 \cdot s^2 (x_l - y_l) \cdot e^{(-s \sum_{i=1}^{d} (x_i - y_i)^2)} \cdot (2 \cdot s \cdot (x_l - y_l)^2 - 1)$

- $\partial_j^2 \partial_{j+d}^2 k(x,y) = 4 \cdot s^2 (x_j - y_j) \cdot e^{(-s \sum_{i=1}^{d} (x_i - y_i)^2)} \cdot (3 + 4 \cdot s^2 \cdot (x_j - y_j)^4 - 12 (x_j - y_j)^2)$ and
  $\partial_j^2 \partial_{l+d}^2 k(x,y) = 4 \cdot s^2 (x_l - y_l) \cdot e^{(-s \sum_{i=1}^{d} (x_i - y_i)^2)} \cdot (1 - 2 \cdot s \cdot (x_j - y_j)^2) \cdot (1 - 2 \cdot s \cdot (x_l - y_l)^2)$

Bibliography

[1] Canu, S. and A. Smola (2006) "Kernel methods and the exponential family," *Neurocomput.*, **69**(7-9), pp. 714–720.

[2] Fukumizu, K. (2009) *Exponential manifold by reproducing kernel Hilbert spaces*, Cambridge University Press, pp. 291–306.

[3] Aronszajn, N. (1950) "Theory of reproducing kernels," *Transactions of the American Mathematical Society*, **68**(3), pp. 337–404.

[4] Sriperumbudur, B., K. Fukumizu, A. Gretton, A. Hyvärinen, and R. Kumar (2017) "Density estimation in infinite dimensional exponential families," *Journal of Machine Learning Research*, **18**(57), pp. 1–59.

[5] van der Vaart, A. W. and J. H. van Zanten (2008) "Rates of contraction of posterior distributions based on Gaussian process priors," *Ann. Statist.*, **36**(3), pp. 1435–1463.

[6] Strathmann, H., D. Sejdinovic, S. Livingstone, Z. Szabo, and A. Gretton (2015) "Gradient-free Hamiltonian monte carlo with efficient kernel exponential families," NIPS'15.

[7] Fukumizu, K., A. Gretton, X. Sun, and B. Schölkopf (2008) "Kernel Measures of Conditional Dependence," in *Advances in Neural Information Processing Systems 20*, Curran Associates, Inc., pp. 489–496.

[8] Gretton, A., K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola (2012) "A Kernel two-sample test," *J. Mach. Learn. Res.*, **13**, pp. 723–773.

[9] Brown, L. D. (1986) "Fundamentals of Statistical Exponential Families with Applications in Statistical Decision Theory," *Lecture Notes-Monograph Series*, **9**, pp. i–279.

[10] Tsybakov, A. B. (2008) *Introduction to Nonparametric Estimation*, 1st ed., Springer Publishing Company, Incorporated.

[11] Hyvärinen, A. (2005) "Estimation of non-normalized statistical models by score matching," *J. Mach. Learn. Res.*, **6**, pp. 695–709.

[12] Hyvarinen, A. (2007) "Some extensions of score matching," *Computational Statistics Data Analysis*, **51**, pp. 2499–2512.

[13] Dawid, A. (2007) "The geometry of proper scoring rules," *Annals of the Institute of Statistical Mathematics*, **59**, pp. 77–93.

[14] Parry, M., A. P. Dawid, and S. Lauritzen (2012) "Proper local scoring rules," *Ann. Statist.*, **40**(1), pp. 561–592.

[15] Johnson, O. (2004) *Information Theory and the Central Limit Theorem*, Imperial College Press.

[16] Ley, C. and Y. Swan (2013) "Steins density approach and information inequalities," *Electronic Communications in Probability [electronic only]*, **18**.

[17] Forbes, P. G. and S. Lauritzen (2015) "Linear estimating equations for exponential families with application to Gaussian linear concentration models," *Linear Algebra and its Applications*, **473**, pp. 261 – 283, special issue on Statistics.

[18] van de Geer, S. (2009) *Empirical Processes in M-Estimation*, Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press.

[19] Wasserman, L. (2006) *All of Nonparametric Statistics (Springer Texts in Statistics)*, Springer-Verlag, Berlin, Heidelberg.

[20] Barron, A. R. and C.-H. Sheu (1991) "Approximation of density functions by sequences of exponential families," *Ann. Statist.*, **19**(3), pp. 1347–1369.

[21] Sun, S., M. Kolar, and J. Xu (2015) "Learning structured densities via infinite dimensional exponential families," in *Advances in Neural Information Processing Systems 28*, Curran Associates, Inc., pp. 2287–2295.