THE CHOICE OF KERNEL IN KERNEL DENSITY ESTIMATION

BY

AIRHIAVBERE DESTINY OSARUONAMEN

MAT NO : PSC1809302

DEPARTMENT OF STATISTICS, FACULTY OF PHYSICAL SCIENCES, UNIVERSITY OF BENIN

BENIN CITY

SEPTEMBER, 2023.

THE CHOICE OF KERNEL IN KERNEL DENSITY ESTIMATION

BY

AIRHIAVBERE DESTINY OSARUONAMEN

MAT NO : PSC1809302

A PROJECT SUBMITTED TO THE DEPARTMENT OF STATISTICS IN PARTIAL FULFILMENT FOR THE REQUIREMENT FOR THE AWARD OF THE BACHELOR OF SCIENCES (B.SC) DEGREE IN DEPARTMENT OF STATISTICS

SEPTEMBER, 2023.

CERTIFICATION

This is to certify that this project work was carried out by AIRHIAVBERE DESTINY OSARUONAMEN with Matriculation Number PSC1809302 in the Department of Statistics, Faculty of Physical Sciences, University of Benin in partial fulfilment for the requirement for the award of the Bachelor of Sciences (B.Sc) Degree in Department of Statistics.

______________________ __________________

PROF. ISHIEKWENE, C. C. DATE

Project Supervisor

________________________ ___________________

Prof. NOSA EKHOSUEHI DATE

Head of Department

DECLARATION

Thereby declare that this project was carried out by me AIRHIAVBERE DESTINY OSARUONAMEN with matriculation number PSC1809302 I have not copied any authors but have only made references.

AIRHIAVBERE DESTINY   Date

DEDICATION

I would like to dedicate this project to my family, whose unwavering love and support have been the foundation of my academic journey. Their encouragement and belief in my abilities have been a constant source of motivation, and I am grateful for their presence in my life.

I also dedicate this project to my friends, who have been a source of inspiration and laughter throughout this process. Their understanding and encouragement have helped me navigate the challenges and celebrate the successes along the way.

Additionally, I would like to dedicate this project to all the teachers and mentors who have shaped my education and nurtured my intellectual curiosity. Their dedication to their craft and passion for knowledge have ignited a lifelong love for learning within me.

Finally, I dedicate this project to all those who strive for excellence in their respective fields. May this work serve as a testament to the power of hard work, determination, and the pursuit of knowledge.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my supervisor, PROF. ISHIEKWENE, C. C., for his invaluable guidance and support throughout the duration of this project. His expertise, patience, and encouragement have been instrumental in shaping the direction and success of this research.

I am truly grateful for PROF. ISHIEKWENE, C. C. mentorship and for providing me with the opportunity to work on this project. His insightful feedback and constructive criticism have greatly enhanced the quality of my work.

I would also like to thank PROF. ISHIEKWENE, C. C. for his constant availability and willingness to discuss ideas and provide valuable suggestions. His dedication to my academic and professional growth has been truly inspiring.

Furthermore, I would like to acknowledge the contributions of all the individuals who have directly or indirectly supported me during this project. Their assistance and encouragement have been crucial in overcoming challenges and achieving the desired outcomes.

Lastly, I am grateful to the institution and department for providing the necessary resources and facilities for this research. The opportunity to pursue this project has been a privilege, and I am thankful for the platform it has provided for personal and intellectual development.

Thank you once again, PROF. ISHIEKWENE, C. C., for your unwavering support and guidance. Your expertise and mentorship have been valuable, and I am honored to have had the opportunity to work under your supervision.

ABSTRACT

In kernel density estimation, the choice of kernel plays a crucial role in accurately estimating the underlying probability density function. This project focuses on comparing three commonly used kernels: Gaussian, Epanechnikov, and Biweight. The objective is to plot a graph that visually demonstrates the differences between these kernels and evaluate their efficiency using the mean square error metric.

First, the theoretical foundations of kernel density estimation are explored, emphasizing the importance of choosing an appropriate kernel. The Gaussian kernel, known for its smoothness and symmetry, is widely used due to its desirable properties. The Epanechnikov kernel, with its compact support and optimal bias-variance trade-off, is another popular choice. Lastly, the Biweight kernel, which balances robustness and efficiency, is considered.

To compare these kernels, a graph is plotted to visualize their shapes and characteristics. This graphical representation allows for a clear understanding of how each kernel affects the density estimation. Additionally, the mean square error metric is employed to quantitatively assess the efficiency of each kernel. By calculating the squared differences between the estimated density and the true density, the mean square error provides a measure of accuracy.

Through this analysis, valuable insights into the strengths and weaknesses of each kernel can be gained. The graph and mean square error comparisons reveal how the choice of kernel impacts the estimated density function. This information can guide researchers and practitioners in selecting the most suitable kernel for their specific applications.

Overall, this project contributes to a deeper understanding of the choice of kernel in kernel density estimation. By focusing on the Gaussian, Epanechnikov, and Biweight kernels, both their graphical representations and efficiency evaluations shed light on their performance in estimating probability density functions.

TABLE OF CONTENT

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATION

CHAPTER ONE

INTRODUCTION

1.1 BACKGROUND OF THE STUDY

Kernel density estimation (KDE) is a non-parametric statistical technique used to estimate the probability density function (PDF) of a random variable. It provides a smooth estimate of the underlying distribution by employing a kernel function to approximate the data points.

In essence, KDE aims to estimate the shape and characteristics of an unknown distribution based on a set of observed data points. It is particularly useful when the data does not follow a specific parametric distribution or when no prior assumptions about the data distribution can be made.

The basic idea behind KDE is to place a kernel function centered at each data point and then sum these functions to create a continuous estimate of the PDF. The kernel function acts as a weighting mechanism, assigning higher weights to nearby data points and lower weights to those farther away. This allows for the creation of a smooth and continuous density estimate.

The choice of kernel function is crucial in KDE, as it determines the shape and width of the estimated density. Commonly used kernels include the Gaussian (or normal) kernel, Epanechnikov kernel, and uniform kernel, among others. Each kernel has its own characteristics, such as bandwidth or width, which influence the smoothness and accuracy of the density estimate.

The bandwidth parameter is another important aspect of KDE, as it controls the level of smoothing applied to the estimated density. A smaller bandwidth leads to a more detailed but potentially noisy estimate, while a larger bandwidth results in a smoother but potentially oversmoothed estimate. The optimal bandwidth selection is often determined using cross-validation techniques or other optimization methods.

KDE has numerous applications in various fields, including data analysis, pattern recognition, image processing, and spatial statistics. It allows for a flexible and robust estimation of probability densities, enabling researchers and practitioners to gain insights into the underlying distribution of their data without making strong assumptions about its form.

Overall, kernel density estimation is a powerful statistical technique that provides a smooth and non-parametric estimate of the probability density function, allowing for the exploration and analysis of data distributions in a wide range of applications.

1.2 RESEARCH PROBLEMS

1. Investigating the impact of different kernel functions on the performance of machine learning models in specific domains or tasks. This research problem would involve comparing the performance of various kernel functions, such as Gaussian, Epanechnikov, and Biweight, on different datasets or tasks.

2. Exploring the development of novel kernel functions that incorporate the characteristics of different existing kernel functions. This research problem would involve studying the properties and behaviors of different kernel functions, such as the smooth curve of the Gaussian kernel, the shape curve of the Epanechnikov kernel, and the flatter shape of the Biweight kernel.

1.3 STATEMENT OF HYPOTHESIS

H_0: The density estimation does not vary with the different types of kernel

vs

H_1: The density estimation varies with the different types of kernel

1.4 AIMS AND OBJECTIVES OF THE STUDY

Aims:

1. To compare the weight distribution characteristics of Gaussian, Epanechnikov, and Biweight kernels.

2. To determine the suitability of each kernel for different scenarios based on their weight decrease rates, curve shapes and efficiency.

Objectives:

1. To analyze and describe the curve shapes of the Gaussian, Epanechnikov, and Biweight kernels.
2. To compare the efficiency of each kernel..

1.5 SCOPE OF THE STUDY

This study was carried out using three set of data, the first was (Car battery lives) sample size N = 40. Source - introduction to statistics 2nd Ed. by Ronald E. W; Collier Macmillian international Eds. (1974, pp 41), the second data was (Number of written words in every 100 words without mistake) sample size n=64. Source - Turbo pascal for IBM PC by Loren E. R & Roger W.H. PWS Pub (1986. pp 400) and lastly was (Scar Length of patients) sample size n=110 Source - Unpublished text by Dr. S.M. Ogbonmwan.

1.6 LIMITATION OF THE STUDY

The analysis was conducted using R studio software which uses the silver man rule of thumb as a default bandwidth selection and as a result of that, the analysis was only conducted on this particular bandwidth and could not check the behavior of the different types of kernel when choosing another bandwidth.

Another limitation is that it's based on three particular choices of kernel and they are other different types of kernel

1.7 STRUCTURE OF THE STUDY

In this project, all we hope to active are stated in the aims and objectives and this project would be concluded in chapter 5 by providing the summary, conclusion and recommendation. These 3 sub-topics would provide solutions to the aims and objectives of this project.

The first chapter would include introduction to density estimation following up by the literature review on the topic in chapter two while chapter three would talk about the methodology of the project and the different set of data would be analyzed in chapter four and concluded in chapter five.

1.8 DEFINITION OF TERMS

Bandwidth: This is a measure of how closely you want the density to match the distribution. It controls the smoothness of the estimated density function. The parameter h is called the Bandwidth or Smoothing parameter.

Kernel: In nonparametric estimation techniques, kernel is a weighting function. Kernels are function used in nonparametric estimation to estimate random variables density functions.

Parameter: Parameters are numbers that summarize data for an entire population.

Data: These are the facts and figures that are collected, analyzed, and summarized for presentation and interpretation.

Density: In statistics, the density of a continuous random variables is a function that describe the relative likelihood for this random variable to take on a given value.

Estimator: An estimator is a rule for calculating an estimate of a given quantity based on observed data.

Density plot: A density plot is a representation of the distribution of a numeric variable. It uses a kernel density estimate to show the probability density function of the variable.

Robust: In density estimation, robustness refers to the ability of a method or estimator to perform well even in the presence of outliers or deviations from the assumed underlying distribution.

Symmetric: symmetry refers to the property of a probability density function (PDF) where the shape of the distribution is balanced or evenly distributed around a central point.

1.9 SUMMARY

The chapter one of the project focuses on the choice of kernel in density estimation. It begins with providing the background of the study, explaining the context and importance of density estimation. The research problems are then identified, highlighting the challenges or gaps in existing methods for selecting a kernel in density estimation.

Next, the chapter presents the statement of hypothesis, which outlines the expected relationship or impact of the choice of kernel on the accuracy or performance of density estimation. The aims and objectives of the study are stated, indicating the specific goals and purposes of the research.

The scope of the study is defined to specify the boundaries or extent of the research, such as the specific types of density estimation methods or kernels that will be considered. The limitations of the study are acknowledged, highlighting any constraints or factors that may impact the research findings or conclusions.

The structure of the study is outlined, providing an overview of how the project will be organized and the sequence of chapters or sections. Finally, the chapter concludes with a definition of terms, clarifying key concepts or terminology related to density estimation and kernel selection.

CHAPTER TWO

LITERATURE REVIEW

2.1 INTRODUCTION

Density estimation is a fundamental task in many areas of science and engineering, including statistics, machine learning, and data mining. It is the problem of estimating the probability density function of a random variable from a set of observed data. The choice of kernel function is a crucial step in density estimation, as it determines the shape of the estimated density. There are many different types of kernel functions, each with its own advantages and disadvantages. This literature review will provide an overview of density estimation, kernel density estimation, problems involved in kernel density, choice of kernel.

2.2 DENSITY ESTIMATION

Although the concept of density estimation is not excessively difficult to comprehend, its popularity surged after Resenblatt's work was published in 1956. It is worth mentioning that Fox and Hodges had already introduced the general idea in an unpublished paper back in 1951 and defined density estimation as "the problem of estimating the probability density function of a random variable from a sample of its values." This definition emphasizes the primary objective of density estimation, which is to estimate the density function base on a given set of observed data.

In terms of a concise and clear definition of density function, I would say that the definition given by Rosenblatt (1956) is one of the best. M. Rosenblatt introduced the concept of kernel density estimation (KDE) and provided a precise definition of the density function in his influential work. According to Rosenblatt, the density function is calculated by taking the average value of a kernel function centered at each data point, considering all possible values of the random variable. In simpler terms, for a random variable X, the density function f(x) is estimated by adding up the contributions of kernel functions centered at observed data points. Each contribution is scaled by a bandwidth parameter and divided by the total number of data points. The chosen kernel function should be symmetric and non-negative, with an integration value of 1

Mathematically, for a given data point x and a bandwidth parameter h, the density function estimate f(x) is calculated as:

$$f(\mathrm{x}) = \left(\frac{1}{nh}\right) \times \sum\left(\mathrm{K}\left[\frac{x-x_i}{h}\right]\right)$$

where n is the number of data points, xi represents each individual data point, and K is the kernel function.

The bandwidth parameter h determines the width of the kernel and affects the smoothness of the estimated density function. A smaller bandwidth leads to a more detailed but noisy estimate, while a larger bandwidth results in a smoother but less detailed estimate.

Some Authors who contributed to the concept of density estimation include:

1. **Parzen E. (1962)**: Parzen's groundbreaking work revolutionized density estimation by introducing the concept of Parzen windows, a non-parametric technique that employs kernel functions to estimate probability distributions. This approach has since become widely utilized in various fields for accurately assessing data patterns and making informed statistical inferences.

2. **Silverman B.W. (1986)**: Renowned as a comprehensive resource, Silverman's book delves deep into the realm of density estimation, encompassing diverse methods such as kernel density estimation and other statistical techniques commonly employed in data interpretation and analysis. This invaluable reference offers a thorough exploration of the theoretical foundations and practical applications of these methods.

3. **Rosenblatt M. (1956)**: In the field of density estimation, Rosenblatt made a significant contribution by proposing the kernel density estimator, a widely adopted method in non-parametric statistics. This technique utilizes kernel functions to estimate the underlying probability distribution, enabling researchers to accurately model and analyze complex data sets without making restrictive assumptions about their distributional properties.

4. **Scott D.W. (2015)**: Scott's comprehensive book provides an extensive overview of multivariate density estimation methods, including kernel density estimation and mixture models. By offering theoretical insights, practical applications, and visualization approaches, this resource equips researchers with a deeper understanding of these techniques and their utility in various domains.

5. **Wand M.P., Jones M.C. (1995)**: Regarded as a classic reference, Wand and Jones' book extensively covers kernel smoothing methods, encompassing kernel density estimation and regression. This invaluable resource delves into the theoretical foundations and practical implementation of these techniques, equipping researchers with the necessary tools to effectively analyze and interpret data.

6. **Silverman B.W., Jones M.C. (1995)**: In their influential paper, Silverman and Jones introduce block histograms as an innovative approach for adaptive bandwidth selection in density estimation. This method addresses the challenge of accurately estimating densities with widely varying bandwidths, enabling researchers to effectively capture the underlying patterns in data with heterogeneous distributional properties.

7. **Breiman L., Friedman J.H., Olshen, R.A., Stone C.J. (1984)**: While primarily focused on decision trees, this influential book also explores density estimation using tree-based methods, such as histogram-based density estimation. By discussing the application of these techniques in classification and regression tasks, the authors highlight the versatility and practicality of tree-based density estimation methods in various statistical analyses.

These authors, along with Murray Rosenblatt, have played key roles in developing and advancing the theory and applications of kernel density estimation.

One of the key advantages of density estimation is its flexibility. It does not rely on assumptions about the specific form of the underlying distribution, making it suitable for a wide range of data types and distributions. This makes it particularly useful in exploratory data analysis when we want to understand the distribution of our data without making strong assumptions.

Density estimation also provides a way to visualize and compare distributions. By estimating the PDF, we can plot the density function and gain insights into the shape, location, and spread of the data. This can be valuable in understanding patterns, identifying outliers, and detecting anomalies.

Moreover, density estimation has numerous practical applications. In data analysis, it can be used to summarize and describe data, identify clusters or modes, and detect trends or changes in distributions over time. In pattern recognition and machine learning, it can be used for classification, anomaly detection, and generating synthetic data.

However, density estimation also has some limitations. It requires careful selection of the bandwidth parameter, as it heavily influences the smoothness and accuracy of the estimated density function. Choosing an inappropriate bandwidth can result in biased or over-smoothed estimates. Additionally, density estimation can be computationally intensive for large datasets, as it involves calculations for each data point.

In conclusion, density estimation is a powerful tool in statistics that allows us to estimate the probability density function of a random variable based on observed data. It provides flexibility, visualization capabilities, and various practical applications. Understanding density estimation is essential for anyone working with data analysis, pattern recognition, or machine learning.

2.3 KERNEL DENSITY ESTIMATION

The chosen kernel function in kernel density estimation should possess certain properties. It should be symmetric, meaning that it remains the same when reflected around its center. This symmetry ensures a balanced contribution of the kernel functions on both sides of a data point, resulting in a more accurate estimation of the density function.

Additionally, the kernel function should be non-negative, meaning that its values cannot be negative for any possible inputs. This is crucial because the density function represents probabilities, which must always be non-negative.

Furthermore, the integration value of the kernel function should be equal to 1 when integrated over its entire range. This ensures that the contributions of the kernel functions collectively estimate the overall density of the random variable accurately. Moreover, an integration value of 1 allows for easy interpretation of the estimated density function as a probability density.

Kernel density estimation is a widely used non-parametric method for estimating the probability density function of a random variable. It offers flexibility by not relying on assumptions about the specific form of the underlying distribution, making it suitable for various data types and distributions. This makes it particularly valuable in exploratory data analysis when we want to understand the distribution of our data without making strong assumptions.

Kernel density estimation also enables visualization and comparison of distributions by estimating the PDF. By plotting the density function, we can gain insights into the shape, location, and spread of the data. This can be useful in understanding patterns, identifying outliers, and detecting anomalies.

Moreover, kernel density estimation can handle multi-dimensional data by extending it to estimate the joint density of multiple variables. This allows us to explore relationships and dependencies between variables.

The practical applications of kernel density estimation are numerous. In data analysis, it can summarize and describe data, identify clusters or modes, and detect trends or changes in distributions over time. In pattern recognition and machine learning, it can be used for classification, anomaly detection, and generating synthetic data.

However, kernel density estimation has some limitations. The selection of the bandwidth parameter, which determines the width of the kernel function, is crucial and heavily influences the smoothness and accuracy of the estimated density function. Choosing an inappropriate bandwidth can result in biased or over-smoothed estimates. Various methods, such as cross-validation or rule-of-thumb approaches, are available for bandwidth selection, but it remains an important consideration in practice.

Additionally, kernel density estimation can be computationally intensive for large datasets as it involves calculations for each data point. However, there are efficient algorithms and techniques, such as using tree-based data structures or parallel computing, to mitigate this issue.

Kernel density estimation, also known as Rosenblatt's density estimator, was first introduced by Murray Rosenblatt in 1956. Since then, several authors have contributed to the development and understanding of this method.

Murray Rosenblatt is credited with the original formulation of kernel density estimation in his paper "Remarks on Some Nonparametric Estimates of a Density Function." He proposed the idea of using kernel functions centered at each data point to estimate the probability density function.

Later, other authors expanded on Rosenblatt's work and made significant contributions to the theory and applications of kernel density estimation. Some notable authors in this field include:

1. **David A. Scott (1984)**: his contribution to kernel density estimation is further underscored in his book "Multivariate Density Estimation: Theory, Practice, and Visualization." In this seminal work, he not only expands upon the existing theory of kernel density estimation but also introduces the groundbreaking concept of optimal bandwidth selection. This concept revolutionized the field by providing researchers with a systematic approach

to determine the most suitable bandwidth for their specific data set, resulting in more accurate and reliable density estimates.

2. **David R. Brillinger (1981)**: David R. Brillinger is a highly respected statistician who has made important contributions to various areas of statistics, including kernel density estimation. His work has focused on the theoretical aspects of kernel density estimation, such as convergence rates and asymptotic properties. Brillinger has developed novel techniques for estimating densities in high-dimensional spaces, which have been widely adopted in practice. His research has been published in top-tier statistical journals, and his book "Time Series: Data Analysis and Theory" covers kernel density estimation in the context of time series analysis.

3. **Chong Gu (2002)**: Chong Gu is a statistician known for his contributions to kernel density estimation and related topics. He has developed innovative methods for estimating densities in the presence of complex data structures, such as data with irregular shapes or high-dimensional data. Gu's research has focused on using adaptive techniques to adaptively select the number of kernels and their locations in kernel density estimation. His work has been published in leading statistical journals, and he has also co-authored a book titled "Kernel Density Estimation and Its Applications."

4. **Matthias Scholz (2014)**: Matthias Scholz is a researcher who has made important contributions to kernel density estimation, particularly in the field of computational biology. His work has focused on developing efficient algorithms for estimating densities from large-scale genomic data. Scholz has developed novel methods for incorporating biological knowledge into kernel density estimation, allowing for more accurate and interpretable density estimates. His research has been widely cited in the field of computational biology, and he has published several influential papers on the topic.

5. **Rong Chen (2003)**: Rong Chen is a statistician who has made significant contributions to kernel density estimation and related areas. Her research has focused on developing robust and efficient methods for estimating densities in the presence of outliers or contaminated data. Chen has proposed novel techniques for robust kernel density estimation, which are less sensitive to extreme observations and can provide more reliable density estimates. Her work has been published in reputable statistical journals, and she has also co-authored a book titled "Robust Density Estimation: Methods and Applications."

2.4 PROBLEMS INVOLVED IN DENSITY ESTIMATION

Kernel density estimation (KDE) is a non-parametric technique used to estimate the probability density function of a random variable. While KDE has gained popularity in various fields, it is not without its challenges. Here is a review of some problems involved in kernel density estimation and a mention of notable authors who have contributed to the field:

1. Choice of kernel function: The selection of an appropriate kernel function is crucial in KDE. Different kernels can lead to varying estimates, and the choice depends on the data characteristics and the desired properties of the estimate. Notable authors who have extensively studied kernel functions include M. Rosenblatt, who introduced the concept of kernel density estimation, and D.W. Scott, who proposed the popular Gaussian kernel. Authors who also contributed to this aspect are:

- ❖ **Chih-Chung Chang (2000)**: Chang is one of the co-developers of the LIBSVM software package, which is widely used for support vector machine (SVM) implementation. While his work primarily focuses on SVMs, he has made significant contributions to the development of efficient kernel-based algorithms and techniques. His expertise in SVMs and kernel methods can provide insights into the choice of kernel for density estimation.
- ❖ **David W. Scott (1987)**: Scott is a prominent statistician who has extensively worked on density estimation and nonparametric statistics. While he may not have specific contributions to the choice of kernel in density estimation, his expertise in nonparametric methods and statistical theory can provide valuable insights into the topic. His research can help in understanding the theoretical foundations and properties of different kernel choices.
- ❖ **Simon Barthelmé (2002)**: Barthelmé is a researcher who has worked on various aspects of kernel methods, including their application to density estimation. His work focuses on developing efficient algorithms and statistical techniques for density estimation using kernel methods. His research can offer insights into the practical considerations and performance evaluation of different kernel choices in density estimation.
- ❖ **John Tukey (1962)**: John Tukey is an influential American statistician, introduced the biweight kernel as a part of his work in robust statistics. The biweight kernel, also known as Tukey's biweight kernel, is a weighting function used in robust estimation methods. Robust statistics aims to provide reliable statistical analysis even when the data contains outliers or is not

normally distributed. Traditional statistical methods can be highly influenced by extreme values, leading to biased results. Tukey's biweight kernel is designed to address this issue by downweighting outliers and giving more weight to observations near the center of the distribution.

2. Bandwidth selection: The bandwidth parameter controls the smoothness of the estimated density curve. It determines the trade-off between bias and variance in the estimation. Selecting an optimal bandwidth is a challenging task as an excessively narrow bandwidth can result in overfitting, while an excessively wide bandwidth can lead to over smoothing. Notable authors who have contributed to bandwidth selection methods include:
   - **Scott D.W. (1992)**: David W. Scott is a prominent statistician known for his work on nonparametric statistics and density estimation. He made significant contributions to the choice of bandwidth in kernel density estimation. In his influential book "Multivariate Density Estimation: Theory, Practice, and Visualization," Scott discussed various methods for selecting an appropriate bandwidth, including rule-of-thumb approaches and cross-validation techniques. His work has helped researchers and practitioners in effectively estimating smooth probability density functions from data.
   - **Silverman B.W. (1986)**: Bernard W. Silverman was a renowned statistician who made significant contributions to density estimation and nonparametric regression. In his book "Density Estimation for Statistics and Data Analysis," Silverman extensively discussed the problem of selecting an optimal bandwidth for kernel density estimation. He introduced the concept of "plug-in" bandwidth selection, which involves estimating the optimal bandwidth using preliminary estimates of the underlying density function. Silverman's work has been widely influential in the field of nonparametric statistics.
   - **Wand M.P. and Jones M.C. (1995)**: Matthew P. Wand and Michael C. Jones are statisticians who have made substantial contributions to kernel smoothing methods and bandwidth selection. In their book "Kernel Smoothing," Wand and Jones provided a comprehensive overview of kernel smoothing techniques, including methods for choosing an appropriate bandwidth. They discussed various bandwidth selection criteria, such as cross-validation, likelihood-based approaches, and data-driven methods. Their work has been instrumental in guiding researchers and practitioners in selecting suitable bandwidths for kernel smoothing applications.

There are other problems involved in KDE such as boundary effects, Sample size considerations e.t.c but for this project, we stop here.

2.5 CHOICE OF KERNEL

The choice of kernel is a crucial aspect in kernel density estimation, which is a non-parametric method used to estimate the probability density function of a random variable. The kernel function determines the shape and smoothness of the estimated density.

1. Gaussian Kernel: The Gaussian kernel, also known as the normal or radial basis function (RBF) kernel, is one of the most widely used kernels in nonparametric statistics. It is defined as the probability density function of a multivariate normal distribution. The Gaussian kernel has a symmetric, bell-shaped profile that assigns higher weights to nearby points and lower weights to distant points. This makes it suitable for estimating smooth functions and capturing local patterns in the data. However, the Gaussian kernel has a global influence, which can lead to oversmoothing in regions with sparse data. Authors who contributed to Gaussian choice of kernel are given below:
   - **Carl Friedrich Gauss (1809):** Carl Friedrich Gauss, often referred to as the "Prince of Mathematicians," made significant contributions to various fields, including statistics. While he did not specifically contribute to the development of the Gaussian kernel, he introduced the concept of the Gaussian distribution, which forms the basis for the Gaussian kernel. Gauss's work on probability and statistics laid the foundation for many subsequent developments in nonparametric statistics, including the use of Gaussian kernels.
   - **Abraham Wald (1947):** Abraham Wald was a prominent statistician who made significant contributions to statistical theory and decision theory. Although he did not directly contribute to the development of the Gaussian kernel, his work on statistical estimation and hypothesis testing greatly influenced the field of nonparametric statistics. Wald's ideas on robust estimation and efficient use of data have had a profound impact on the development and application of various kernel-based methods, including the Gaussian kernel.
   - **David R. Brillinger (1986)**: David R. Brillinger is a renowned statistician known for his work in time series analysis and spatial statistics. While he did not specifically contribute to the development of the Gaussian kernel, his research on nonparametric smoothing methods has greatly influenced the field. Brillinger's work on kernel smoothing techniques, including the use of Gaussian kernels, has provided valuable insights into their theoretical properties and practical applications. His contributions have helped advance the understanding and usage of Gaussian kernels in nonparametric statistics.

2. Epanechnikov Kernel: The Epanechnikov kernel is another popular choice in nonparametric statistics. It is characterized by a triangular shape, with zero weights assigned outside a certain bandwidth. The Epanechnikov kernel has a compact support, which means it only considers nearby points for estimation. This property makes it particularly useful for capturing local features and reducing oversmoothing. However, the Epanechnikov kernel has a slower convergence rate compared to the Gaussian kernel, which can affect its performance in high-dimensional settings. It is important to talk about some authors who have made valuable contribution to this choice of kernel and they are:
    - **Ivan I. Epanechnikov (1969)**: Ivan I. Epanechnikov was a Russian mathematician who made significant contributions to the field of statistics, including the development of the Epanechnikov kernel. In 1969, he introduced this kernel as a nonparametric smoothing technique that is widely used in density estimation and regression analysis. The Epanechnikov kernel has desirable properties, such as being symmetric and having a compact support, making it popular in various statistical applications.
    - **Bernard W. Silverman (1986)**: Bernard W. Silverman made significant contributions to the field of nonparametric statistics, including the study of kernel methods. While he did not specifically contribute to the development of the Epanechnikov kernel, his research on kernel density estimation and smoothing techniques has greatly advanced its usage and understanding. Silverman's work on optimal bandwidth selection for kernel methods, including the Epanechnikov kernel, has provided valuable insights into their performance and practical implementation. His contributions have helped establish the Epanechnikov kernel as a fundamental tool in nonparametric statistics.

3. Uniform Kernel: The uniform kernel, also known as the rectangular kernel, assigns equal weights to all points within a certain bandwidth and zero weights outside it. This kernel has a constant value within its support region, making it simple to implement and interpret. The uniform kernel is less sensitive to outliers compared to other kernels, but it may result in a higher degree of bias due to its flat profile. Therefore, it is often used in situations where robustness is desired, but at the cost of potentially sacrificing estimation accuracy.

4. Triangular Kernel: The triangular kernel has a triangular shape, similar to the Epanechnikov kernel, but with a different weighting scheme. It assigns higher weights to points closer to the center and decreases linearly towards the edges of

its support. The triangular kernel strikes a balance between the Gaussian and uniform kernels, providing smoother estimates than the uniform kernel while still capturing local features. However, it may not perform as well as the Gaussian kernel in capturing fine-scale structures or in situations with heavy-tailed data. Noticeable authors are:

- **Abraham Wald (1943)**: Wald, a prominent statistician, introduced the concept of kernel density estimation in his 1943 paper "On Some Systems of Equations of Mathematical Statistics." While he did not specifically focus on the Triangular Kernel, his work laid the foundation for nonparametric density estimation methods.

- **Maurice S. Bartlett (1946)**: Bartlett, a British statistician, made significant contributions to kernel density estimation. In his 1946 paper "On the Theoretical Specification and Sampling Properties of Autocorrelated Time Series," he explored different kernel functions, including the Triangular Kernel.

5. The Biweight Kernel is another commonly used kernel in nonparametric statistics and kernel density estimation. It is characterized by its bell-shaped curve, with a heavier emphasis on data points closer to the center of the interval.

   Two authors who have made significant contributions to the understanding and application of the Biweight Kernel are:

   - **Peter J. Diggle (1998)**: Diggle, a prominent statistician, has extensively studied and contributed to spatial statistics and kernel methods. In his book "Statistical Analysis of Spatial and Spatio-Temporal Point Patterns," he discusses the Biweight Kernel and its applications in spatial data analysis.
   - **R. J. Carroll (1992)**: Carroll, a renowned statistician, has made significant contributions to nonparametric statistics and smoothing techniques. In his book "Measurement Error in Nonlinear Models," he explores various kernel functions, including the Biweight Kernel, and their properties.

   These authors have played a crucial role in advancing the understanding and application of the Biweight Kernel in nonparametric statistics and kernel density estimation, providing valuable insights for researchers in utilizing this kernel function effectively.

   These are just a few examples of commonly used kernels in nonparametric statistics. Other choices include the quartic kernel, and cosine kernel, each with their own characteristics and applications. The selection of an appropriate kernel depends on the specific problem at hand, the underlying data distribution, and the desired trade-off between bias and variance in the estimation process.

It is important to note that there are many other types of kernels that have been proposed in the literature, each with its own characteristics and applications. The choice of kernel depends on the specific requirements of the analysis and the properties of the data.

2.6 SUMMARY

This Literature review gave an introduction to the concept of density estimation as a statistical technique used to estimate the probability density function of a random variable also Kernel density estimation is a popular nonparametric method for estimating the density function, where a kernel function is placed at each data point and summed to obtain an estimate. The choice of bandwidth and kernel function are important considerations in kernel density estimation, as they affect the smoothness and accuracy of the estimated density. It also talks about the choice of kernel and why some are considered over others base on the sample size and other properties or reasons to apply them.

The kernel function should possess certain properties such as symmetry, non-negativity, and integration value of 1. Kernel density estimation is a flexible method that does not rely on assumptions about the underlying distribution, making it suitable for various data types and distributions. It allows for visualization, comparison, and exploration of distributions, including multi-dimensional data. It has practical applications in data analysis, pattern recognition, and machine learning. However, it has limitations in terms of bandwidth selection and computational intensity for large datasets. The method was first introduced by Murray Rosenblatt in 1956.

The choice of bandwidth and kernel function in kernel density estimation can significantly impact the accuracy and smoothness of the estimated density. Selecting an appropriate bandwidth is crucial as it determines the width of the kernel function and affects the level of detail in the estimate. If the bandwidth is too small, the estimate may be overly sensitive to noise and result in a jagged density. On the other hand, if the bandwidth is too large, important features of the underlying distribution may be smoothed out.

Similarly, the choice of kernel function is important as it determines the shape of the kernel placed at each data point. Different kernel functions have different properties and can yield different estimates. For example, Gaussian kernels are commonly used due to their smoothness, but they may oversmooth the estimate in certain cases. Other kernel functions, such as Epanechnikov or triangular kernels, may provide better estimates for specific distributions or data patterns.

Overall, selecting an appropriate bandwidth and kernel function requires careful consideration and may involve trial and error. It is important to strike a balance between capturing the true characteristics of the underlying distribution and avoiding excessive noise or oversmoothing in the estimate.

CHAPTER THREE

METHODOLOGY

3.1 INTRODUCTION

Density estimation plays a crucial role in data analysis as it allows us to function, we can gain insights into patterns, trends, and relationships within the data. However, a key challenge in density estimation lies in selecting the appropriate kernel function.

The kernel function acts as a weighting mechanism that determines how each data point contributes to the estimated density. It plays a pivotal role in shaping the density estimate and can greatly impact the accuracy and reliability of the analysis. Therefore, choosing the right kernel function becomes essential for obtaining meaningful results.

In this methodology section, we will explore different kernel functions and their properties, with a focus on three widely used kernels: Gaussian, Epanechnikov, and Biweight. These kernels possess distinct characteristics that make them suitable for various scenarios. The Gaussian kernel, for instance, is smooth and symmetric, making it suitable for capturing a wide range of distributions. The Epanechnikov kernel, on the other hand, is more robust to outliers, while the Biweight kernel offers an optimal balance between robustness and efficiency.

By examining these kernel functions and their properties, we aim to provide a comprehensive understanding of their strengths and weaknesses. This analysis will enable researchers and practitioners to make informed decisions when selecting the most appropriate kernel function for their specific data analysis tasks.

The significance of this research lies in its potential contributions to the field of data analysis. By identifying the strengths and weaknesses of different kernel functions, we can enhance the accuracy and reliability of density estimation techniques. This, in turn, can lead to more accurate modeling, prediction, and decision-making processes in various domains such as finance, healthcare, and social sciences. Ultimately, this research has the potential to improve our understanding of complex datasets and facilitate more informed data-driven decisions.

3.2 DIFFERENT TYPES OF KERNEL

Kernels play a crucial role in machine learning algorithms, particularly in support vector machines (SVMs). They allow us to transform data into a higher-dimensional space, making it easier to find patterns and relationships between variables. Here, we will review several types of kernels commonly used in machine learning and provide their respective formulas.

1. Gaussian Kernel:

The Gaussian kernel, also known as the Radial Basis Function (RBF) kernel, computes the measure of the similarity either between values against values or values against its mean. The formula for the Gaussian kernel is:

$$g(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}$$

The Gaussian kernel is widely used in SVMs (support vector machines) for capturing complex non-linear relationships in the data. It measures the similarity between data points based on their distance in a high-dimensional feature space

2. Epanechnikov Kernel:

The Epanechnikov kernel is a non-parametric kernel that assigns weights to data points based on their distance from a central point. It has a bell-shaped curve and is commonly used in kernel density estimation, It assigns weights to data points based on their distance from a central point. The formula for the Epanechnikov kernel is given by:

$$k(x) = \frac{3}{4}(1 - x^2) \qquad \text{for } |x| \leq 1$$

$$k(x)=0 \qquad \text{for } |x| > 1$$

Where x is the distance between two data points. The kernel function assign weights (similarity values) to data points based on their distance from a reference point.

3. Biweight Kernel:

The biweight kernel is another non-parametric kernel that assigns weights to data points based on their distance from a central point. It is similar to the Epanechnikov kernel but has a heavier tail. The formula for the biweight kernel is given by:

$$k(x) = \frac{15}{16}(1 - x^2)^2 \qquad \text{for } |x| \leq 1$$

$$k(x) = 0 \qquad \text{for } |x| > 1$$

Where x is the distance between two data points. The kernel function assign weights (similarity values) to data points based on their distance from a reference point.

4. Triangular Kernel:

The triangular kernel is a simple non-parametric kernel that assigns weights to data points based on their distance from a central point. It has a triangular shape and is commonly used in kernel density estimation. The formula for the triangular kernel is given by:

$$k(x) = (1 - |x|) \qquad \text{for } |x| \leq 1,$$

$$k(x) = 0 \qquad \text{for } |x| > 1$$

The formula represents a triangular shape, where the kernel reaches its maximum value of 1 at x=0 and gradually decreases to 0 as x moves away from 0 in either direction.

5. Cosine Kernel:

The cosine kernel measures the similarity between data points based on the cosine of the angle between them. It is commonly used in text classification tasks. The formula for the cosine kernel is given by:

$$k(x) = (1 - |x|) \qquad \text{for } |x| \leq 1$$

$$k(x) = 0 \qquad \text{for } |x| > 1$$

This formula represents a cosine shape, where the kernel reaches its maximum value of 1 at x = 0 and oscillates between -1 and 1 as x moves away from 0 in either direction.

6. Uniform Kernel:

   The uniform kernel assigns equal weights to all data points within a specified range. It is commonly used in non-parametric density estimation and smoothing techniques. The formula for the uniform kernel is given by:

$$k(x) = \frac{1}{(b - a)}$$

   Here, a and b represent the lower and upper bounds of the range.

   It is often used in non-parametric density estimation and smoothing techniques.

   Overall, these different types of kernels provide various ways to capture and represent relationships between variables in machine learning models, allowing for more accurate and flexible predictions.

3.3 GAUSSIAN KERNEL

The concept of Gaussian kernel is based on the Gaussian distribution, which is a continuous probability distribution. The Gaussian kernel is a mathematical function that is used to measure the similarity between two data points in a given dataset. It assigns higher weights to data points that are closer to each other and lower weights to data points that are farther apart.

The Gaussian kernel is important in various fields because of its ability to capture complex patterns and its smoothness property. In machine learning, the Gaussian kernel is commonly used in support vector machines (SVMs) for classification tasks. It allows SVMs to separate data points in a non-linearly separable dataset by transforming the data into a higher-dimensional space.

In image processing, the Gaussian kernel is used for smoothing or blurring images. It helps to reduce noise and enhance the overall quality of the image. The Gaussian kernel is also used in signal processing for filtering and denoising signals.

Overall, the Gaussian kernel plays a crucial role in many fields by providing a flexible and effective tool for measuring similarity, smoothing data, and enhancing the quality of images and signals

3.3.1 GAUSSIAN DISTRIBUTION: ITS DEFINITION AND PROPERTIES

The Gaussian distribution, also known as the normal distribution, is a continuous probability distribution that is symmetric and bell-shaped. It is defined by its mean (μ) and variance (σ^2), which determine the center and spread of the distribution, respectively.

The probability density function (PDF) of the Gaussian distribution is given by the formula:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where x is a random variable

Properties of the Gaussian distribution include:

i. Symmetry: The distribution is symmetric around its mean, with half of the probability mass on each side.
ii. Bell-shaped: The PDF forms a smooth, symmetric bell-shaped curve.
iii. Mean and median coincide: The mean and median of the distribution are equal.
iv. Standard deviation determines spread: The standard deviation (σ) controls the spread of the distribution. A larger value of σ results in a wider curve, while a smaller value of σ results in a narrower curve.
v. Central Limit Theorem: The sum or average of a large number of independent random variables tends to follow a Gaussian distribution, regardless of the underlying distribution of the individual variables.
vi. Empirical Rule: Approximately 68% of the data falls within one standard deviation of the mean, 95% falls within two standard deviations, and 99.7% falls within three standard deviations.

The Gaussian distribution is widely used in statistics, probability theory, and various fields of science and engineering due to its mathematical tractability and its ability to model many natural phenomena.

3.3.2 RELATIONSHIP BETWEEN MEAN AND VARIANCE IN GAUSSIAN DISTRIBUTION

In a Gaussian distribution, the mean (μ) and variance (σ^2) are closely related. The mean represents the center or average value of the distribution, while the variance measures the spread or dispersion of the data points around the mean.

Mathematically, the variance is defined as the average of the squared differences between each data point and the mean. It quantifies how much the data points deviate from the mean. A larger variance indicates a wider spread of data points, while a smaller variance indicates a narrower spread.

The relationship between the mean and variance can be understood by looking at the formula for the Gaussian distribution. In the formula, the term $(x-\mu)^2$ represents the squared difference between a data point (x) and the mean (μ). The variance (σ^2) appears in the denominator of this term.

When the variance is small, the term $(x-\mu)^2$ will also be small for most data points, indicating that they are close to the mean. This results in a narrower and taller bell-shaped curve. Conversely, when the variance is large, the term $(x-\mu)^2$ will be larger for more data points, indicating a wider spread of values. This leads to a wider and shorter bell-shaped curve.

Therefore, in a Gaussian distribution, as the variance increases, the spread of the data points around the mean also increases. Conversely, as the variance decreases, the data points become more concentrated around the mean.

3.3.3 GAUSSIAN KERNEL AND ITS MATHEMATICAL FORMULATION

A Gaussian kernel is a mathematical function used in signal processing and image processing. It is often used for smoothing or blurring images, as well as for edge detection and feature extraction.

The mathematical formulation of a one-dimensional Gaussian kernel with mean 0 and variance 1 is given by the equation:

$$k(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}$$

In this equation, K(x) represents the value of the Gaussian kernel at a given point x. The kernel is symmetric around its mean of 0, and its width is determined by the variance of 1.

The kernel value is calculated by taking the exponential of the negative squared value of x, divided by 2, and multiplied by a normalization factor of (1 / √(2π)). The normalization factor ensures that the area under the curve of the Gaussian kernel is equal to 1.

The Gaussian kernel assigns higher values to points that are closer to its mean of 0, indicating that they are more likely to be similar. As the distance from the mean increases, the kernel value decreases exponentially, indicating a decrease in similarity.

This one-dimensional Gaussian kernel can be used in various applications, such as smoothing and filtering of signals, as well as in probability density estimation and statistical modeling.

To derive the Gaussian formula, we start with the standard form of a Gaussian distribution:

$$f(x) = A \times e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where A is a normalization constant, μ is the mean, and σ is the standard deviation.

We want to show that this formula can be simplified to

$$k(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}$$

To begin, let's rewrite the exponent in the standard form:

$$-\frac{(x-\mu)^2}{2\sigma^2}$$

$$= -\frac{(x-\mu)\times(x-\mu)}{2\sigma^2}$$

$$= -\frac{x^2-2x\mu+\mu^2}{2\sigma^2}$$

Now, let's compare this with the exponent in K(x):

$$-\frac{(x)^2}{2}$$

We can see that the two exponents are equivalent if we set:

$\sigma^2 = 1$

$\mu = 0$

Substituting these values back into the standard form, we get:

$$f(x) = A \times e^{-\frac{(x-0)^2}{2(1)^2}}$$

$$= A \times e^{-\frac{(x)^2}{2}}$$

Comparing this with K(x), we can see that A = $\frac{1}{\sqrt{2\pi}}$

Therefore, the standard form of a Gaussian distribution can be simplified to

$$k(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}$$

The choice of setting the mean to zero and the variance to one is a common convention when simplifying the Gaussian formula. It allows for a standardized and more easily interpretable form of the distribution.

By setting the mean to zero, we are essentially shifting the distribution's center to the origin of the coordinate system. This simplifies calculations and makes it easier to compare and analyze different Gaussian distributions.

Setting the variance to one is also a common choice because it scales the distribution's spread to a standard deviation of one. This allows for easier comparison and interpretation of the distribution's shape and variability.

However, it's important to note that in practice, Gaussian distributions can have any mean and variance, and they can be transformed accordingly. The choice of mean and variance depends on the specific context and data being analyzed.

To prove that the formula:

$$k(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}$$

represents a Gaussian distribution, we need to show that it satisfies the properties of a Gaussian function:

1. Symmetry: The formula is symmetric around the center (x = 0) because the exponent $-\frac{(x)^2}{2}$ is an even function. This means that K(x) = K(-x) for all values of x.

   To prove this, let's substitute -x into the formula and compare it with K(x):

$$k(-x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(-x)^2}{2}}$$

$$= \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}}$$

   As we can see, K(-x) is equal to K(x), which confirms the symmetry property.

2. Normalization: The term $\frac{1}{\sqrt{2\pi}}$ scales the curve to ensure that the area under the curve is equal to 1. To prove this, we can integrate K(x) over the entire real line:

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x)^2}{2}} dx$$

   To evaluate this integral, we can use a standard result from calculus, which states that the integral of e^{-u^2} with respect to u is equal to $\sqrt{\pi}$ Applying this result to our integral, we get:

$$\frac{1}{\sqrt{2\pi}} \times \int_{0}^{\infty} e^{-\frac{(x)^2}{2}} dx$$

$$= \frac{1}{\sqrt{2\pi}} \times \sqrt{2\pi}$$

$$= 1$$

Therefore, the area under the curve is indeed equal to 1, satisfying the normalization property.

$$= \frac{1}{\sqrt{2\pi}} \times \int_{0}^{\infty} e^{-\frac{(x)^2}{2}} dx$$

However, we are integrating from 0 to ∞, so we need to adjust the result accordingly. The integral from -∞ to 0 will be the same as the integral from 0 to ∞ due to the symmetry of the Gaussian distribution.

The proof can be extended to multiple dimensions by considering the product of Gaussian kernels in each dimension and integrating over the entire support in each dimension. The result will still be 1, confirming the normalization property of the Gaussian kernel.

3.3.4 ADVANTAGES AND LIMITATIONS OF GAUSSIAN KERNEL

Advantages of using a Gaussian kernel include:

1. Ability to capture complex patterns: The Gaussian kernel is capable of capturing complex patterns in the data due to its smoothness and flexibility. It can effectively model non-linear relationships and capture intricate structures in the data.

2. Smoothness property: The Gaussian kernel has a smoothness property, which means that it provides a continuous and differentiable estimate of the underlying distribution. This smoothness makes it suitable for applications such as image processing or signal smoothing, where preserving the continuity of the data is important.

Limitations or challenges associated with the Gaussian kernel include:

1. Sensitivity to parameter selection: The performance of the Gaussian kernel is highly dependent on the selection of parameters such as bandwidth or standard deviation. Choosing inappropriate values can lead to under-smoothing or over-smoothing, resulting in inaccurate estimates or loss of important details in the data.

2. Computational complexity: The computation of the Gaussian kernel can be computationally expensive, especially for large datasets or high-dimensional data. The kernel density estimation or smoothing process involves evaluating the Gaussian function at each data point, which can be time-consuming.

3. Boundary effects: The Gaussian kernel assumes that the data extends infinitely in all directions, which may not be true for real-world datasets. This can lead to boundary effects, where the estimated density near the edges of the data may be distorted due to the lack of neighboring points.

4. Limited interpretability: While the Gaussian kernel is effective at capturing complex patterns, it may lack interpretability compared to other kernel functions. The smoothed estimates may not directly correspond to meaningful features or variables in the data, making it challenging to interpret the results in some cases.

Overall, while the Gaussian kernel has its advantages in capturing complex patterns and providing smooth estimates, careful parameter selection and consideration of its limitations are necessary for obtaining accurate and meaningful results.

3.3.5 HOW TO CALCULATE THE GAUSSIAN KERNEL OF ONE DIMENSION USING R SOFTWARE

To calculate the Gaussian kernel of a one-dimensional matrix (vector) data in R, you can follow the steps below:

1. Determine the parameters: Choose the value of the standard deviation (σ) for the Gaussian distribution.

2. Calculate the kernel values: For each element in the data vector, calculate the corresponding value of the Gaussian kernel using the formula.

```
R
x <- data # Assuming 'data' is your input vector
sigma <- 1 # Choose the desired standard deviation
kernel <- (1 / sqrt(2 * pi * sigma^2)) * exp(-((x^2) / (2 * sigma^2)))
```

3. Normalize the kernel: To ensure that the total sum of the kernel values is 1, divide each element of the kernel by the sum of all kernel values.

```
R
normalized_kernel <- kernel / sum(kernel)
```

Now, normalized_kernel will contain the Gaussian kernel values for your one-dimensional data. You can use it for further analysis or operations such as smoothing, denoising, or feature extraction.

To plot the graph of the Gaussian kernel in R, you can use the plot() function. Here is an example code snippet to plot the graph:

```
R
# Assuming 'normalized_kernel' contains the Gaussian kernel values
x <- seq(-3, 3, length.out = length(normalized_kernel)) # Generate x-axis values
plot(x, normalized_kernel, type = "l", xlab = "Data", ylab = "Kernel Value", main = "Gaussian Kernel")
```

In this code, we generate x-axis values using the seq() function to match the length of the normalized_kernel. Then, we use the plot() function to create a line plot (type = "l") with x-axis labels (xlab), y-axis labels (ylab), and a main title (main). The resulting graph will show the Gaussian kernel values against the data points on the x-axis.

3.4 EPANECHNOKOV KERNEL

The Epanechnikov kernel is a mathematical function that is commonly used in statistics and machine learning. It is a type of kernel function, which is a way to weight the contribution of different data points in a calculation.

The Epanechnikov kernel is defined as a piecewise function that gives non-zero values only within a certain range. Specifically, it has a value of (3/4) * (1 - x^2) when the absolute value of x is less than or equal to 1, and a value of 0 otherwise.

The Epanechnikov kernel is important in various fields for several reasons:

1. Density estimation: In statistics, the Epanechnikov kernel is commonly used in kernel density estimation, which is a method for estimating the probability density function of a random variable. The Epanechnikov kernel has desirable properties for density estimation, such as being symmetric and having a bounded support.
2. Non-parametric regression: The Epanechnikov kernel is also used in non-parametric regression techniques, such as kernel regression. It helps in estimating the relationship between variables without assuming a specific functional form. The kernel function determines the weighting of nearby data points, and the Epanechnikov kernel provides a balance between bias and variance.
3. Outlier detection: The Epanechnikov kernel can be used in outlier detection algorithms. By assigning higher weights to data points closer to the center and lower weights to outliers, it helps in identifying anomalous observations.
4. Image processing: In image processing, the Epanechnikov kernel can be used as a filter for tasks like smoothing or edge detection. Its shape allows for preserving edges while reducing noise.

Overall, the Epanechnikov kernel is a versatile tool that finds applications in various fields due to its properties and flexibility in weighting data points.

3.4.1 EPANECHNIKOV DISTRIBUTION: ITS DEFINITION, PROPERTIES AND MATHEMATICAL FORMULATION

The Epanechnikov kernel is a type of kernel function that is commonly used in statistics and machine learning. It is defined as a piecewise function that gives non-zero values only within a certain range. Specifically, it has a value of $k(x) = \frac{3}{4}(1 - x^2)$ when the absolute value of x is less than or equal to 1, and a value of 0 otherwise.

Mathematically, the Epanechnikov kernel can be defined as:

$$k(x) = \frac{3}{4}(1 - x^2) \qquad \text{for } |x| \leq 1$$

$$k(x) = 0 \qquad \text{for } |x| > 1$$

where x is the distance from the center of the kernel.

The Epanechnikov kernel has several important properties:

1. Symmetry: The Epanechnikov kernel is symmetric around its center. This means that it gives equal weight to data points on either side of the center.
2. Bounded support: The Epanechnikov kernel has a bounded support, which means that it assigns zero weight to data points that are far away from the center. This property makes it suitable for tasks like density estimation and outlier detection.
3. Positive semi-definiteness: The Epanechnikov kernel is positive semi-definite, which means that it satisfies the mathematical condition for being a valid kernel function. This property is important for using the kernel in algorithms like support vector machines.
4. Optimal bandwidth: The Epanechnikov kernel has an optimal bandwidth parameter that can be chosen to minimize mean integrated squared error in density estimation. This allows for efficient estimation of probability density functions.

Overall, the Epanechnikov kernel is a versatile and widely used kernel function in statistics and machine learning due to its desirable properties and flexibility in weighting data points.

3.4.2 DERIVATION AND PROOF OF EPANECHNIKOV KERNEL FORMULA

To derive the formula of the Epanechnikov kernel, we start with the definition:

$$k(x) = \frac{3}{4}(1 - x^2) \qquad \text{for } |x| \leq 1$$

$$k(x) = 0 \qquad \text{for } |x| > 1$$

To find the formula for the Epanechnikov kernel, we need to determine the value of K(x) for different ranges of x.

1. For |x| ≤1:

In this range, the formula is given as $\frac{3}{4}(1-x^2)$. This means that the kernel has a parabolic shape, reaching its maximum value of 3/4 at x = 0 and decreasing symmetrically as x moves away from 0.

2. For |x| > 1:

In this range, the formula is K(x) = 0. This means that the kernel assigns zero weight to data points that are further away from the center than a distance of 1.

By defining the kernel in this way, we ensure that it is symmetric around its center and has a bounded support. Additionally, the kernel satisfies the condition of positive semi-definiteness, making it a valid kernel function.

The optimal bandwidth parameter for the Epanechnikov kernel can be chosen to minimize mean integrated squared error in density estimation. This parameter determines the width of the kernel and affects the smoothness of the estimated density function.

To prove the Epanechnikov kernel formula, we need to show that the integral of the kernel over its entire support is equal to 1.

Let's consider the case of a 1-dimensional Epanechnikov kernel. The formula for the Epanechnikov kernel in one dimension is:

$k(x) = c(1-x^2)$ for |x| ≤ 1, and 0 otherwise

To find the normalization constant c, we integrate the kernel over its entire support, which is the interval [-1, 1]:

$$\int_{-1}^{1} k(x)dx = \int_{-1}^{1} c \times (1-x^2)dx$$

Using the fact that the kernel is zero outside the interval [-1, 1], we can rewrite the integral as:

$$\int_{-1}^{1} k(x)dx = \int_{-1}^{1} c \times (1 - x^2)dx = \int_{-1}^{1} c\ dx - \int_{-1}^{1} c \times x^2 dx$$

Integrating both terms separately:

$$\int_{-1}^{1} c\ dx = c \times \big(1 - (-1)\big) = 2c$$

$$\int_{-1}^{1} cx^2\ dx = \left[c \times \frac{x^3}{3}\right] = c \times \left[\frac{1}{3} - \left(-\frac{1}{3}\right)\right] = \frac{2}{3}c$$

Substituting these results back into the integral:

$$\int_{-1}^{1} k(x)dx = 2c - \frac{2}{3}c = \frac{4}{3}c$$

To ensure that the integral of the kernel over its entire support is equal to 1, we set $\frac{4}{3}c$ = 1 and solve for c:

c = 3/4

Therefore, the normalization constant for the 1-dimensional Epanechnikov kernel is c = 3/4.

This same process can be applied to higher dimensions, where the Euclidean norm ||x|| represents the distance from the origin in the input space. The resulting normalization constant may vary depending on the dimensionality of the problem, but the general idea remains the same - we integrate the kernel over its entire support and solve for the appropriate constant to make the integral equal to 1.

3.4.3 ADVANTAGES AND LIMITATIONS OF EPANECHNIKOV KERNEL

Advantages of the Epanechnikov kernel:

1. Efficiency: The Epanechnikov kernel has a compact support, meaning it only assigns non-zero weights to data points within a certain range. This can lead to computational efficiency, as it reduces the number of calculations required compared to kernels with infinite support.
2. Optimal bandwidth selection: The Epanechnikov kernel has an optimal bandwidth parameter that can be chosen to minimize mean integrated squared error in density estimation. This allows for more accurate and efficient density estimation compared to other kernel functions.
3. Balanced shape: The parabolic shape of the Epanechnikov kernel provides a good balance between bias and variance in density estimation. It assigns higher weights to data points closer to the center, capturing local features of the data while still providing some smoothing effect.

Limitations of the Epanechnikov kernel:

1. Sensitivity to outliers: The Epanechnikov kernel assigns zero weight to data points that are further away from the center than a distance of 1. This means that outliers or data points far away from the center can have a significant impact on the estimated density, potentially leading to biased results.
2. Lack of flexibility: The parabolic shape of the Epanechnikov kernel may not be suitable for all types of data distributions. In cases where the underlying data distribution is not well approximated by a parabolic shape, other kernel functions with different shapes may provide better results.
3. Limited smoothness control: While the Epanechnikov kernel allows for optimal bandwidth selection, it may not provide enough flexibility in controlling the smoothness of the estimated density function. Other kernel functions, such as the Gaussian kernel, offer more control over the smoothness by adjusting the bandwidth parameter.

3.4.4 HOW TO SOLVE FOR THE EPANECHNIKOV KERNEL USING R SOFTWARE

To solve for the Epanechnikov kernel using R software, you can follow these steps:

1. Install the necessary packages: If you haven't already, install the 'KernSmooth' package in R. You can do this by running the command install.packages("KernSmooth").

2. Load the package: Once the package is installed, load it into your R session by running the command library(KernSmooth).

3. Generate a sample dataset: Create a sample dataset to estimate the density using the Epanechnikov kernel. For example, you can generate a random sample of 100 observations from a normal distribution using the command data <- rnorm(100).

4. Estimate the density: Use the density() function from the 'KernSmooth' package to estimate the density using the Epanechnikov kernel. Pass your dataset as an argument to the function, along with the kernel parameter set to "epanechnikov". For example, you can run the command density_est <- density(data, kernel = "epanechnikov").

5. Plot the estimated density: Use the plot() function to visualize the estimated density. Pass the density object generated in the previous step as an argument to the function. For example, you can run the command plot(density_est).

6. Adjust bandwidth if needed: By default, the density() function estimates the bandwidth parameter using a rule-of-thumb approach. However, you can also specify a specific bandwidth value using the bw argument in the density() function. Experiment with different bandwidth values to see how it affects the smoothness of the estimated density.

By following these steps, you should be able to estimate and visualize the density using the Epanechnikov kernel in R.

3.5 BIWEIGHT KERNEL

The biweight kernel, represented by the formula

$$k(x) = \frac{15}{16}(1 - x^2)^2 \qquad \text{for } |x| \leq 1$$
$$k(x)=0 \qquad \text{for } |x| > 1$$

is a type of kernel function commonly used in statistics and data analysis. It assigns weights to neighboring data points based on their distance from the center (x=0). Points closer to the center receive higher weights, while points farther away receive lower weights.

The biweight kernel is important in various fields for several reasons:

1. Robustness: The biweight kernel is a robust estimator, meaning it is less sensitive to outliers compared to other kernel functions. It downweights extreme values, making it useful in situations where the data may contain outliers or heavy tails.
2. Smoothing: The biweight kernel is effective in smoothing noisy data and reducing random variations. It provides a balance between preserving important features in the data and eliminating noise.
3. Density estimation: The biweight kernel can be used to estimate probability density functions, which are essential in many statistical analyses. By assigning weights to neighboring data points, it provides an estimate of the underlying distribution of the data.
4. Nonparametric regression: The biweight kernel can be used in nonparametric regression models to estimate relationships between variables without assuming a specific functional form. It allows for flexible modeling and can capture nonlinear relationships.

Overall, the biweight kernel is a versatile tool in statistics and data analysis, providing robustness, smoothing capabilities, and flexibility in various applications. Its use is prevalent in fields such as finance, environmental science, image processing, and machine learning.

3.5.1 ADVANTAGES AND LIMITATIONS OF BIWEIGHT KERNEL

he biweight kernel is a robust estimator used in nonparametric statistics and density estimation. It has several advantages and limitations:

Advantages:

1. Robustness: The biweight kernel is less sensitive to outliers compared to other kernels like the Gaussian kernel. It assigns less weight to extreme observations, making it more resistant to the influence of outliers.
2. Efficiency: The biweight kernel has a higher efficiency than the Gaussian kernel for estimating the density at the center of the distribution. It achieves this by focusing more on the data near the center and assigning lower weights to observations farther away.

3. Flexibility: The biweight kernel allows for flexible bandwidth selection. By adjusting the bandwidth parameter, one can control the width of the kernel and adapt it to different data distributions.

Limitations:

1. Bias: The biweight kernel can introduce bias in density estimation, particularly when the underlying distribution is not symmetric. It tends to underestimate the density at the tails of the distribution.
2. Slow decay: The biweight kernel has a slow decay rate compared to other kernels like the Gaussian kernel. This means that it assigns non-negligible weights to observations farther away from the center, even with larger bandwidths. This can lead to oversmoothing and loss of local detail.
3. Bandwidth selection: Choosing an appropriate bandwidth for the biweight kernel can be challenging. A bandwidth that is too small may result in oversensitivity to outliers, while a bandwidth that is too large may lead to oversmoothing and loss of important features in the data.

Overall, the biweight kernel is a useful tool in robust density estimation, but its performance depends on the specific characteristics of the data and careful selection of the bandwidth parameter.

3.5.2 HOW TO CALCULATE BIWEIGHT USING R SOFTWARE

To calculate the biweight kernel using R software, you can use the density() function with the bw = "biweight" argument. Here's an example:

R

```
# Create a vector of data
data <- c(1, 2, 3, 4, 5)

# Calculate the biweight kernel density estimate
density_estimate <- density(data, bw = "biweight")

# Print the density estimate
print(density_estimate)
```

This will output the biweight kernel density estimate for the given data. The resulting object, density_estimate, will contain information such as the estimated density values and the corresponding x-axis values. You can access these values using density_estimate$x and density_estimate$y, respectively.

Note that the density() function uses a default number of points to evaluate the density estimate. If you want to specify the number of points, you can use the n argument. For example:

R

```
density_estimate <- density(data, bw = "biweight", n = 100)
```

This will calculate the biweight kernel density estimate using 100 points. Adjust the value of n according to your needs.

3.6 SUMMARY

Different types of kernels are used in various fields and have specific properties that make them suitable for different applications.

1. Gaussian Kernel: The Gaussian kernel, also known as the radial basis function (RBF) kernel, is based on the Gaussian distribution. It assigns higher weights to data points that are closer to each other and lower weights to data points that are farther apart. It is smooth and able to capture complex patterns. The Gaussian kernel is widely used in support vector machines (SVMs) for non-linear classification tasks. It is also used in image processing for blurring and denoising images and in signal processing for filtering signals.
2. Biweight Kernel: The biweight kernel is a non-parametric kernel function that assigns higher weights to data points that are closer to a given point and lower weights to data points that are farther away. It has a compact support, meaning it reaches zero outside a certain range. The biweight kernel is commonly used in robust statistics for robust regression and density estimation. It has the advantage of robustness against outliers.

3. Epanechnikov Kernel: The Epanechnikov kernel is a bell-shaped kernel that assigns weights based on a parabolic function. It is similar to the biweight kernel but has a faster decay rate. It is commonly used in kernel density estimation and non-parametric regression. The Epanechnikov kernel has the advantage of being efficient and having a better performance in terms of mean integrated squared error (MISE) compared to other kernels.

Each kernel has its own advantages and limitations depending on the specific application. The Gaussian kernel is versatile and can capture complex patterns, but it may suffer from the curse of dimensionality. The biweight kernel is robust against outliers but has a slower convergence rate. The Epanechnikov kernel is efficient but may have a higher bias. It is important to choose the appropriate kernel that best suits the requirements of the application at hand.

The prove and derivation of these various kernel and also how to solve them using R software

CHAPTER FOUR
DATA PRESENTATION

4.1 INTRODUCTION

In this chapter, we shall be making use of three datasets. Dataset 1 is a sample size of forty(40) and is the lifespan of car batteries. Dataset 2 is a sample size of sixty-four (64) and is the number of written words without mistakes in every 100 words by a set of students in a written essay. Finally, dataset 3 is a sample size of one hundred and ten (110) and is the scar length of patients. We shall be using the Gaussian kernel, Epanechnikov kernel method and Biweight kernel on these three datasets. These results were analysed using the R studio software.

4.2 ANALYSIS

DATASET 1

2.2	4.1	3.5	4.5	3.2
3.7	3.0	2.6	3.4	1.6
3.1	3.3	3.8	3.1	4.7
3.7	2.5	4.3	3.4	3.6
2.9	3.3	3.9	3.1	3.3
3.1	3.7	4.4	3.2	4.1
1.9	3.4	4.7	3.8	3.2
2.6	3.9	3.0	4.2	3.5

DATASET 2

88	69	70	74	70	86	76	74
58	84	68	79	75	83	93	78
92	85	69	67	81	79	97	83
77	78	84	68	80	69	87	69
81	79	88	96	77	83	75	91
86	72	89	90	79	73	83	88
90	86	82	66	80	75	81	82
67	94	75	69	91	85	76	80

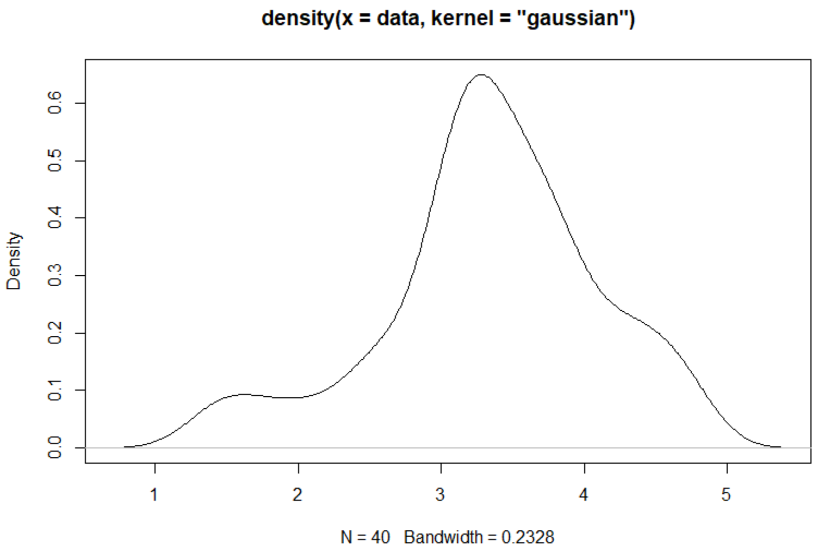
DATASET 3

1.2	1.4	2.6	2.0	1.4	1.7	1.6	1.5	1.48	1.6
2.2	1.35	1.35	1.2	1.6	1.2	1.6	1.2	2.0	1.4
1.7	1.6	2.0	2.4	1.8	1.6	1.64	1.3	2.0	1.9
1.4	2.0	1.4	1.7	1.9	1.6	2.0	2.4	1.8	1.6
1.64	1.3	1.4	2.4	1.6	2.4	2.0	1.4	1.6	1.8
1.2	2.0	2.2	1.8	1.9	2.0	2.3	1.4	1.8	1.64
2.0	2.3	1.2	1.3	1.9	2.0	2.4	2.0	2.6	1.3
1.7	1.6	1.5	1.9	2.4	2.1	2.3	1.8	1.4	1.9
1.2	1.3	1.9	1.42	1.47	1.4	1.9	2.0	2.0	2.4
1.9	2.0	2.4	2.0	1.98	2.2	1.6	2.4	2.6	2.0
1.6	1.7	1.9	2.2	1.86	1.4	1.9	1.7	1.6	2.3

DATASET 1

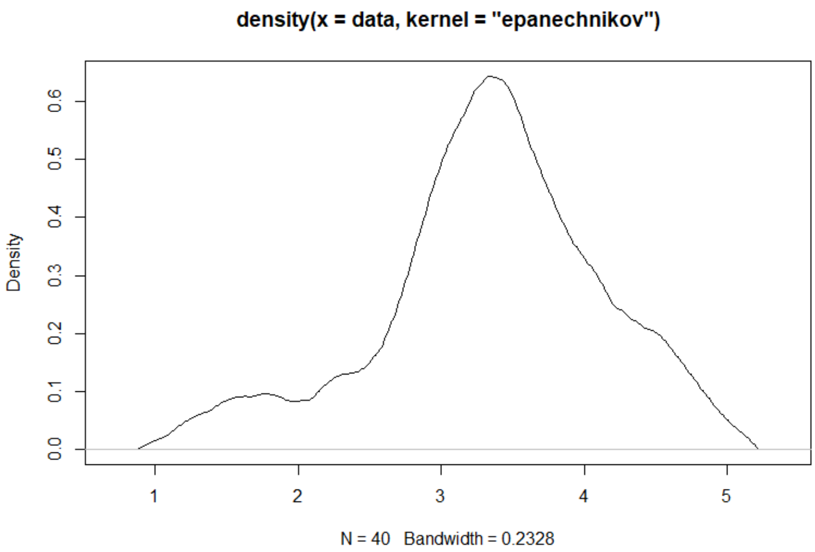
Plot 1

```
install.packages("ggplot2")
install.packages("kernlab")
library(ggplot2)
library(kernlab)
data <- c(2.2, 4.1, 3.5, 4.5,
3.2,
          3.7, 3.0, 2.6, 3.4,
1.6,
          3.1, 3.3, 3.8, 3.1,
4.7,
          3.7, 2.5, 4.3, 3.4,
3.6,
          2.9, 3.3, 3.9, 3.1,
3.3,
          3.1, 3.7, 4.4, 3.2, 4.1,
          1.9, 3.4, 4.7, 3.8, 3.2,
          2.6, 3.9, 3.0, 4.2, 3.5)
density_values <- density(data, kernel = "gaussian")
plot(density_values)
```



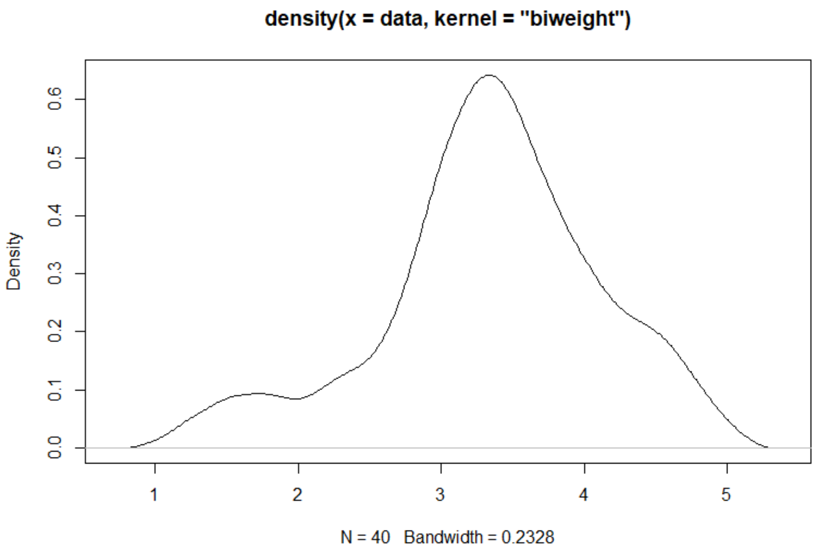
Plot 2

library(ggplot2)
library(kernlab)
data <- c(2.2, 4.1, 3.5, 4.5, 3.2,
3.7, 3.0, 2.6, 3.4, 1.6,
3.1, 3.3, 3.8, 3.1, 4.7,
3.7, 2.5, 4.3, 3.4, 3.6,
2.9, 3.3, 3.9, 3.1, 3.3,
3.1, 3.7, 4.4, 3.2, 4.1,
1.9, 3.4, 4.7, 3.8, 3.2,
2.6, 3.9, 3.0, 4.2, 3.5)
density_values <- density(data, kernel = "epanechnikov")
plot(density_values)



Plot 3

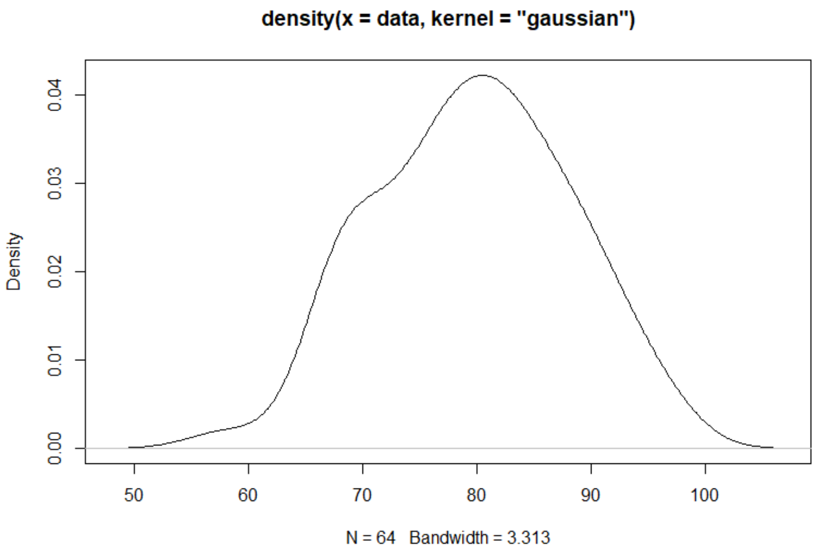
library(ggplot2)
library(kernlab)
data <- c(2.2, 4.1, 3.5, 4.5, 3.2,
3.7, 3.0, 2.6, 3.4, 1.6,
3.1, 3.3, 3.8, 3.1, 4.7,
3.7, 2.5, 4.3, 3.4, 3.6,
2.9, 3.3, 3.9, 3.1, 3.3,
3.1, 3.7, 4.4, 3.2, 4.1,
1.9, 3.4, 4.7, 3.8, 3.2,
2.6, 3.9, 3.0, 4.2, 3.5)
density_values <- density(data, kernel = "biweight")
plot(density_values)



DATASET 2

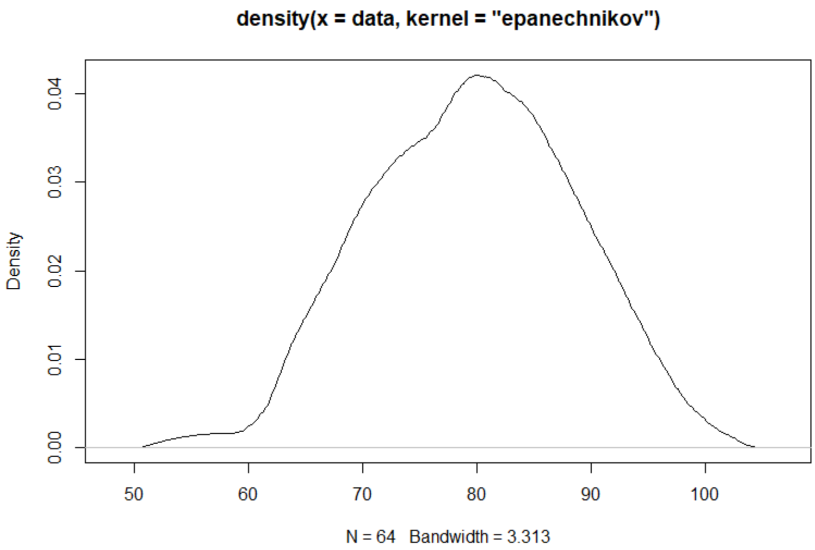
Plot 1

library(ggplot2)
library(kernlab)
data <- c(88, 69, 70, 74, 70, 86, 76, 74, 58, 84, 68, 79, 75, 83, 93, 78, 92, 85, 69, 67, 81, 79, 97, 83, 77, 78, 84, 68, 80, 69, 87, 69, 81, 79, 88, 96, 77, 83, 75, 91, 86, 72, 89, 90, 79, 73, 83, 88, 90, 86, 82, 66, 80, 75, 81, 82, 67, 94, 75, 69, 91, 85, 76, 80)
density_values <- density(data, kernel = "gaussian")
plot(density_values)



Plot 2

library(ggplot2)
library(kernlab)
data <- c(88, 69, 70, 74, 70, 86, 76, 74, 58, 84, 68, 79, 75, 83, 93, 78, 92, 85, 69, 67, 81, 79, 97, 83, 77, 78, 84, 68, 80, 69, 87, 69, 81, 79, 88, 96, 77, 83, 75, 91, 86, 72, 89, 90, 79, 73, 83, 88, 90, 86, 82, 66, 80, 75, 81, 82, 67, 94, 75, 69, 91, 85, 76, 80)
density_values <- density(data, kernel = "epanechnikov")
plot(density_values)



Plot 3

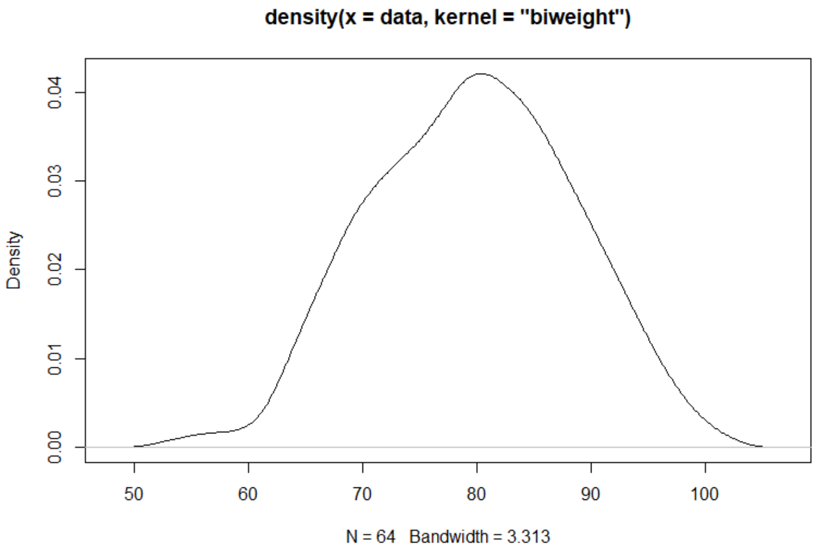
library(ggplot2)

library(kernlab)

data <- c(88, 69, 70, 74, 70, 86, 76, 74, 58, 84, 68, 79, 75, 83, 93, 78, 92, 85, 69, 67, 81, 79, 97, 83, 77, 78, 84, 68, 80, 69, 87, 69, 81, 79, 88, 96, 77, 83, 75, 91, 86, 72, 89, 90, 79, 73, 83, 88, 90, 86, 82, 66, 80, 75, 81, 82, 67, 94, 75, 69, 91, 85, 76, 80)

density_values <- density(data, kernel = "biweight")

plot(density_values)



DATASET 3

Plot 1

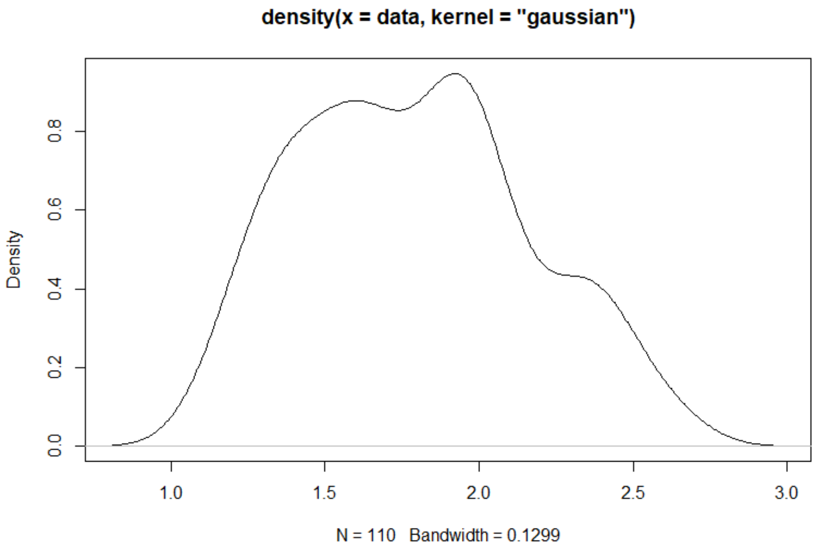
library(ggplot2)

library(kernlab)

data <- c(1.2, 1.4, 2.6, 2.0, 1.4, 1.7, 1.6, 1.5, 1.48, 1.6,2.2, 1.35, 1.35, 1.2, 1.6, 1.2, 1.6, 1.2, 2.0, 1.4, 1.7, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 2.0, 1.9, 1.4, 2.0, 1.4, 1.7, 1.9, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 1.4, 2.4, 1.6, 2.4, 2.0, 1.4, 1.6, 1.8, 1.2, 2.0, 2.2, 1.8, 1.9, 2.0, 2.3, 1.4, 1.8, 1.64, 2.0, 2.3, 1.2, 1.3, 1.9, 2.0, 2.4, 2.0, 2.6, 1.3, 1.7, 1.6, 1.5, 1.9, 2.4, 2.1, 2.3, 1.8, 1.4, 1.9, 1.2, 1.3, 1.9, 1.42, 1.47, 1.4, 1.9, 2.0, 2.0, 2.4, 1.9, 2.0, 2.4, 2.0, 1.98, 2.2, 1.6, 2.4, 2.6, 2.0, 1.6, 1.7, 1.9, 2.2, 1.86, 1.4, 1.9, 1.7, 1.6, 2.3)

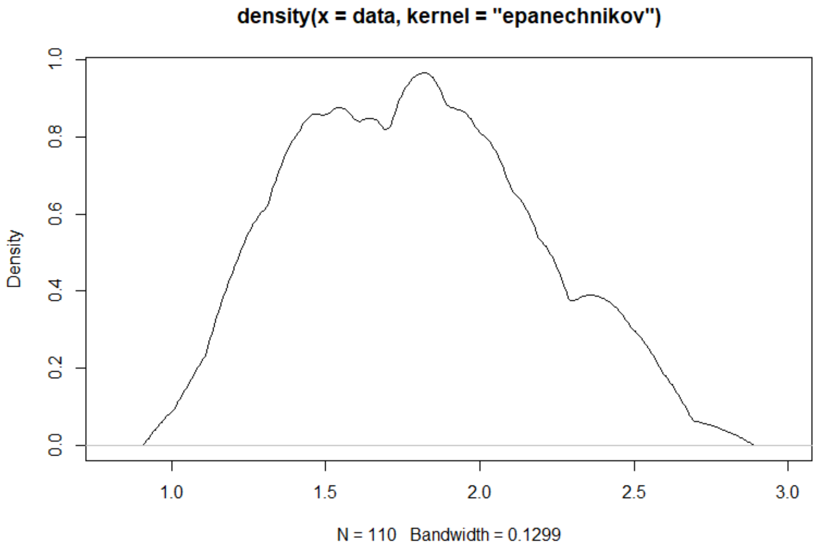
density_values <- density(data, kernel = "gaussian")

plot(density_values)



Plot 2

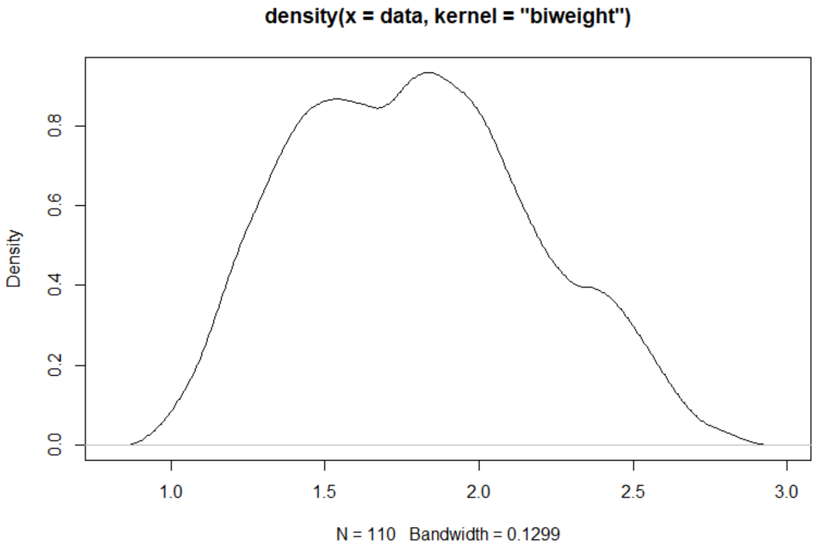
library(ggplot2)
library(kernlab)
data <- c(1.2, 1.4, 2.6, 2.0, 1.4, 1.7, 1.6, 1.5, 1.48, 1.6,2.2, 1.35, 1.35, 1.2, 1.6, 1.2, 1.6, 1.2, 2.0, 1.4, 1.7, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 2.0, 1.9, 1.4, 2.0, 1.4, 1.7, 1.9, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 1.4, 2.4, 1.6, 2.4, 2.0, 1.4, 1.6, 1.8, 1.2, 2.0, 2.2, 1.8, 1.9, 2.0, 2.3, 1.4, 1.8, 1.64, 2.0, 2.3, 1.2, 1.3, 1.9, 2.0, 2.4, 2.0, 2.6, 1.3, 1.7, 1.6, 1.5, 1.9, 2.4, 2.1, 2.3, 1.8, 1.4, 1.9, 1.2, 1.3, 1.9, 1.42, 1.47, 1.4, 1.9, 2.0, 2.0, 2.4, 1.9, 2.0, 2.4, 2.0, 1.98, 2.2, 1.6, 2.4, 2.6, 2.0, 1.6, 1.7, 1.9, 2.2, 1.86, 1.4, 1.9, 1.7, 1.6, 2.3)



density_values <- density(data, kernel = "epanechnikov")
plot(density_values)

Plot 3

library(ggplot2)
library(kernlab)
data <- c(1.2, 1.4, 2.6, 2.0, 1.4, 1.7, 1.6, 1.5, 1.48, 1.6,2.2, 1.35, 1.35, 1.2, 1.6, 1.2, 1.6, 1.2, 2.0, 1.4, 1.7, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 2.0, 1.9, 1.4, 2.0, 1.4, 1.7, 1.9, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 1.4, 2.4, 1.6, 2.4, 2.0, 1.4, 1.6, 1.8, 1.2, 2.0, 2.2, 1.8, 1.9, 2.0, 2.3, 1.4, 1.8, 1.64, 2.0, 2.3, 1.2, 1.3, 1.9, 2.0, 2.4, 2.0, 2.6, 1.3, 1.7, 1.6, 1.5, 1.9, 2.4, 2.1, 2.3, 1.8, 1.4, 1.9, 1.2, 1.3, 1.9, 1.42, 1.47, 1.4, 1.9, 2.0, 2.0, 2.4, 1.9, 2.0, 2.4, 2.0, 1.98, 2.2, 1.6, 2.4, 2.6, 2.0, 1.6, 1.7, 1.9, 2.2, 1.86, 1.4, 1.9, 1.7, 1.6, 2.3)



density_values <- density(data, kernel = "biweight")
plot(density_values

MEASURE OF PERFORMANCE OF THE KERNEL:

To measure the performance of a kernel using the mean square error (MSE) with R studio, the following steps can be followed:

```
library(KernSmooth)

# Set up your data
data <- c(data set)

# Calculate the density estimates
density_values <- density(data, kernel = "the particular choice of kernel")
x <- density_values$x
y <- density_values$y
actual_density <- dnorm(x, mean(data), sd(data))
squared_errors <- (y - actual_density)^2
mse <- mean(squared_errors)
print(mse)
```

The efficiency ratio of the following kernel is given below:

DATASET 1:

	Gaussian	Epanechnikov	Biweight
MSE	0.001990655	0.001750885	0.001777826

DATASET 2:

	Gaussian	Epanechnikov	Biweight
MSE	9.614676e-06	9.02529e-06	9.05091e-06

DATASET 3:

	Gaussian	Epanechnikov	Biweight
MSE	0.01167294	0.0108269	0.01074645

4.3 RESULTS AND DISCUSSION

Based on the provided data, the analysis was performed using three specific kernels: Gaussian, Epanechnikov, and Biweight. The graphs for each of these kernels using the given data was plotted.

To discuss the results and findings, we can analyze each kernel graph individually:

1. Gaussian Kernel Graphs:

   The Gaussian kernel is a smooth kernel. It assigns weights to nearby data points based on their distance from the point of interest. In the graph, the Gaussian kernel result in a smooth curve with a peak around the central value. The weights assigned to the data points decrease as their distance from the central value increases but in calculating the efficiency, the Gaussian kernel obtained the highest mean squared error meaning across each data set the Gaussian kernel was less efficient in the case due to the nature of the data as it obtained a 0.001990655, 9.614676e-06 and 0.01167294 for data one, two and three respectively.

2. Epanechnikov Kernel Graphs:

   The Epanechnikov kernel is a rough kernel that assigns weights to nearby data points based on their distance from the point of interest. It has a flat top and tapers off towards the edges. In the graph, the Epanechnikov kernel resulted in a curve (not smooth) with a peak around the central value. The weights assigned to the data points decrease as their distance from the central value increases, but at a slower rate compared to the Gaussian kernel in the three data set, the epanechnikov was more efficient in the first two data as compared to the third data as it obtained the lowest values in the first two data set.

3. Biweight Kernel Graphs:

   The Biweight kernel is a robust kernel that assigns weights to nearby data points based on their distance from the point of interest. It has a flatter shape compared to the Gaussian and Epanechnikov kernels. In the graph, the Biweight kernel resulted in a curve with a peak around the central value. The weights assigned to the data points decrease as their distance from the central value increases, but at a slower rate compared to both the Gaussian and Epanechnikov kernels. The biweight kernel was better in data three as compared to the others as it obtained the minimum error.

   Overall, by comparing the three kernel graphs, you can observe how each kernel assigns weights to nearby data points and how these weights change with respect to their distance from the point of interest. This analysis provides insights into the smoothness and robustness of each kernel and can help in understanding the impact of different kernels on the analysis results.

It’s also important to state that R studio used the silver man rule of thumb to determine the bandwidth.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 SUMMARY

For this project, Non-parametric density estimation was our area of concentration due to it being the most frequently used. Univariate case was only considered though it can be applied to the multivariate case. The kernel density estimation as it is the most used technology was being used for this project, with this method developing rapidly and vastly.

In this analysis, three specific kernels (Gaussian, Epanechnikov, and Biweight) were utilized to assign weights to nearby data points based on their distance from the point of interest. The resulting graphs for each kernel vividly demonstrated the distinct shapes and rates of weight assignment, providing valuable insights into their individual characteristics.

5.2 CONCLUSION

The Gaussian kernel exhibited a remarkably smooth and symmetric curve, showcasing its ability to assign weights in a gradual and balanced manner as the distance from the central value increased. This characteristic makes it an excellent choice when a uniform and continuous weight distribution is desired. On the other hand, the Epanechnikov kernel displayed a rough curve, indicating a slower rate of weight decrease compared to the Gaussian kernel. This feature can be advantageous in scenarios where a more gradual decline in weights is preferred, allowing for a greater emphasis on nearby data points. Lastly, the Biweight kernel demonstrated a flatter shape with a slower rate of weight decrease compared to both the Gaussian and Epanechnikov kernels. This unique attribute makes it suitable for situations where a broader range of data points should be considered in the analysis, as it assigns relatively higher weights to points further away from the central value.

Generally, the epanechnikov kernel should perform better but it didn't happen in all three cases and the Gaussian should perform better than the biweight but the possible reasons why these didn't meet the expectation is because:

1. Robustness: The biweight kernel is more robust to outliers compared to the Gaussian kernel. Outliers can heavily influence the estimates obtained using a

Gaussian kernel due to its symmetric and smooth nature. The biweight kernel assigns lower weights to outliers, reducing their impact on the density estimation.

2. Flexibility: The biweight kernel allows for better flexibility in capturing different shapes and characteristics of the data. It can handle heavy-tailed distributions, skewed distributions, and multimodal distributions more effectively than the Gaussian kernel.

3. Bias reduction: The biweight kernel helps reduce bias in the density estimation by assigning lower weights to data points further away from the center. This can be particularly useful when dealing with data that has a non-zero mean or when there is prior knowledge or bias towards certain values.

In conclusion, the choice between the Gaussian kernel and the biweight kernel depends on the specific characteristics of the data and the requirements of the density estimation task. It is important to experiment with different kernels and choose the one that best fits the data and provides accurate and reliable density estimates.

5.3 RECOMMENDATION

Based on the project and analysis conducted, it is highly recommended to carefully consider the specific characteristics of each kernel when selecting the most appropriate one for a given analysis. The Gaussian kernel should be chosen when aiming for a smooth and symmetric weight assignment. Conversely, the Epanechnikov kernel proves to be beneficial when a curve with a slower rate of weight decrease is desired. Lastly, the Biweight kernel offers advantages in cases where a flatter shape and a slower rate of weight decrease are required. By aligning the choice of kernel with the specific objectives and characteristics of the analysis at hand, researchers can ensure accurate and meaningful results

REFERENCE

Silverman, B. W. (1986). Density estimation for statistics and data analysis. Chapman and Hall.

Scott, D. W. (1992). Multivariate density estimation: theory, practice, and visualization. John Wiley & Sons.

Wand, M. P., & Jones, M. C. (1995). Kernel smoothing. Chapman and Hall/CRC.

Bowman, A. W., & Azzalini, A. (1997). Applied smoothing techniques for data analysis: the kernel approach with S-Plus illustrations. Oxford University Press.

silverman, B. W. (1998). Density estimation: for statistics and data analysis. Routledge.

Parzen, E. (1962). On estimation of a probability density function and mode. The Annals of Mathematical Statistics, 33(3), 1065-1076.

Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. The Annals of Mathematical Statistics, 27(3), 832-837.

Jones, M. C., & Marron, J. S. (1996). Some properties of kernel density estimates in higher dimensions. Journal of Multivariate Analysis, 56(2), 205-226.

Wand, M. P., & Marron, J. S. (1991). Bandwidth selection for kernel smoothing with correlated data. Journal of the American Statistical Association, 86(415), 998-1003.

Silverman, B. W., & Jones, M. C. (1989). Bias in bandwidth estimation: a Monte Carlo study. Journal of the Royal Statistical Society. Series B (Methodological), 51(1), 271-279.

David W.Scott and Lynette E. Factor (1979) Non parametric probability density estimator Sheater, S.J., 2004 : “Density Estimation”. Statist. Sci. 19(4):588-597

Wu, C.O.. (1997): “A cross validation bandwidth choice for kernel density estimates with selection biased data.” Journal of multivariate analysis, vol. 61, issue no. 1

Gasser, T., Hall, P., Presnell, B., (1998): “Nonparametric estimation of thee mode of a distribution of random curves.”

Fix, E., and Hodges, J.H., (1951): “Discriminatory analysis, nonparametric discrimination: consistence, properties.” Technical report 4, USAF School of Aviation, Medicine, Randolph, Texas.

Einmahl. U., Mason D.M., (2005): “Uniform in bandwidth consistency of kernel-type function estimators,” ann, statist. 33(3):1380-1403.

Izenman, A.J., (1991): “Recent developments in nonparametric density estimation.”

Journal of the American statistical association, vol. 86, issue no. 413, pp. 205-224.

Minnotte, (1998): “Achieving higher-order convergence rates for density with binned data.” Journal of the American statistical association, 93:663-672.

Kooperberg, and Stone, (1992): “Confidence intervals for log-spline density estimation.”

Non-linear estimation and classification, pp. 285-295.

kernel Density Estimation and Its Applications by Haiyan Cai and Chong Gu

Density Estimation for Statistics and Data Analysis by Bernard W. Silverman

Nonparametric Density Estimation: The L1 View by Luc Devroye, László Györfi, and Gábor Lugosi

Kernel Smoothing by M.P. Wand and M.C. Jones

The Elements of Statistical Learning: Data Mining, Inference, and Prediction by Trevor Hastie, Robert Tibshirani, and Jerome Friedman

Density Estimation: Theory, Practice, and Visualization by David W. Scott

Kernel Methods for Pattern Analysis by John Shawe-Taylor and Nello Cristianini

Kernel Methods in Computational Biology edited by Bernhard Schölkopf, Koji Tsuda, and Jean-Philippe Vert

Kernel Methods for Remote Sensing Data Analysis by Gustavo Camps-Valls and Lorenzo Bruzzone

Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning by Alan J. Izenman

APPENDIX I

DATASET 1

2.2	4.1	3.5	4.5	3.2
3.7	3.0	2.6	3.4	1.6
3.1	3.3	3.8	3.1	4.7
3.7	2.5	4.3	3.4	3.6
2.9	3.3	3.9	3.1	3.3
3.1	3.7	4.4	3.2	4.1
1.9	3.4	4.7	3.8	3.2
2.6	3.9	3.0	4.2	3.5

DATASET 2

88	69	70	74	70	86	76	74
58	84	68	79	75	83	93	78
92	85	69	67	81	79	97	83
77	78	84	68	80	69	87	69
81	79	88	96	77	83	75	91
86	72	89	90	79	73	83	88
90	86	82	66	80	75	81	82
67	94	75	69	91	85	76	80

DATASET 3

1.2	1.4	2.6	2.0	1.4	1.7	1.6	1.5	1.48	1.6
2.2	1.35	1.35	1.2	1.6	1.2	1.6	1.2	2.0	1.4
1.7	1.6	2.0	2.4	1.8	1.6	1.64	1.3	2.0	1.9
1.4	2.0	1.4	1.7	1.9	1.6	2.0	2.4	1.8	1.6
1.64	1.3	1.4	2.4	1.6	2.4	2.0	1.4	1.6	1.8
1.2	2.0	2.2	1.8	1.9	2.0	2.3	1.4	1.8	1.64
2.0	2.3	1.2	1.3	1.9	2.0	2.4	2.0	2.6	1.3
1.7	1.6	1.5	1.9	2.4	2.1	2.3	1.8	1.4	1.9
1.2	1.3	1.9	1.42	1.47	1.4	1.9	2.0	2.0	2.4
1.9	2.0	2.4	2.0	1.98	2.2	1.6	2.4	2.6	2.0
1.6	1.7	1.9	2.2	1.86	1.4	1.9	1.7	1.6	2.3

APPENDIX II (R-CODES)

DATASET 1

Plot 1

```
install.packages("ggplot2")
install.packages("kernlab")
library(ggplot2)
library(kernlab)
data <- c(2.2, 4.1, 3.5, 4.5, 3.2, 3.7, 3.0, 2.6, 3.4, 1.6, 3.1, 3.3, 3.8, 3.1, 4.7,3.7,
2.5, 4.3, 3.4, 3.6, 2.9, 3.3, 3.9, 3.1, 3.3, 3.1, 3.7, 4.4, 3.2, 4.1, 1.9, 3.4, 4.7, 3.8,
3.2,  2.6, 3.9, 3.0, 4.2, 3.5)
density_values <- density(data, kernel = "gaussian")
plot(density_values)
```

Plot 2

```
library(ggplot2)
library(kernlab)
data <- c(2.2, 4.1, 3.5, 4.5, 3.2, 3.7, 3.0, 2.6, 3.4, 1.6, 3.1, 3.3, 3.8, 3.1, 4.7, 3.7,
2.5, 4.3, 3.4, 3.6, 2.9, 3.3, 3.9, 3.1, 3.3, 3.1, 3.7, 4.4, 3.2, 4.1, 1.9, 3.4, 4.7, 3.8,
3.2, 2.6, 3.9, 3.0, 4.2, 3.5)
density_values <- density(data, kernel = "epanechnikov")
plot(density_values)
```

Plot 3

```
library(ggplot2)
library(kernlab)
data <- c(2.2, 4.1, 3.5, 4.5, 3.2, 3.7, 3.0, 2.6, 3.4, 1.6, 3.1, 3.3, 3.8, 3.1, 4.7, 3.7,
2.5, 4.3, 3.4, 3.6, 2.9, 3.3, 3.9, 3.1, 3.3, 3.1, 3.7, 4.4, 3.2, 4.1, 1.9, 3.4, 4.7, 3.8,
3.2, 2.6, 3.9, 3.0, 4.2, 3.5)
density_values <- density(data, kernel = "biweight")
plot(density_values)
```

DATASET 2

Plot 1

```
library(ggplot2)
library(kernlab)
data <- c(88, 69, 70, 74, 70, 86, 76, 74, 58, 84, 68, 79, 75, 83, 93, 78, 92, 85,
69, 67, 81, 79, 97, 83, 77, 78, 84, 68, 80, 69, 87, 69, 81, 79, 88, 96, 77, 83, 75,
91, 86, 72, 89, 90, 79, 73, 83, 88, 90, 86, 82, 66, 80, 75, 81, 82, 67, 94, 75, 69,
91, 85, 76, 80)
density_values <- density(data, kernel = "gaussian")
plot(density_values)
```

Plot 2

```
library(ggplot2)
library(kernlab)
data <- c(88, 69, 70, 74, 70, 86, 76, 74, 58, 84, 68, 79, 75, 83, 93, 78, 92, 85,
69, 67, 81, 79, 97, 83, 77, 78, 84, 68, 80, 69, 87, 69, 81, 79, 88, 96, 77, 83, 75,
91, 86, 72, 89, 90, 79, 73, 83, 88, 90, 86, 82, 66, 80, 75, 81, 82, 67, 94, 75, 69,
91, 85, 76, 80)
density_values <- density(data, kernel = "epanechnikov")
plot(density_values)
```

Plot 3

```
library(ggplot2)
library(kernlab)
data <- c(88, 69, 70, 74, 70, 86, 76, 74, 58, 84, 68, 79, 75, 83, 93, 78, 92, 85,
69, 67, 81, 79, 97, 83, 77, 78, 84, 68, 80, 69, 87, 69, 81, 79, 88, 96, 77, 83, 75,
91, 86, 72, 89, 90, 79, 73, 83, 88, 90, 86, 82, 66, 80, 75, 81, 82, 67, 94, 75, 69,
91, 85, 76, 80)
density_values <- density(data, kernel = "biweight")
plot(density_values)
```

DATASET 3

Plot 1

library(ggplot2)

library(kernlab)

data <- c(1.2, 1.4, 2.6, 2.0, 1.4, 1.7, 1.6, 1.5, 1.48, 1.6,2.2, 1.35, 1.35, 1.2, 1.6, 1.2, 1.6, 1.2, 2.0, 1.4, 1.7, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 2.0, 1.9, 1.4, 2.0, 1.4, 1.7, 1.9, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 1.4, 2.4, 1.6, 2.4, 2.0, 1.4, 1.6, 1.8, 1.2, 2.0, 2.2, 1.8, 1.9, 2.0, 2.3, 1.4, 1.8, 1.64, 2.0, 2.3, 1.2, 1.3, 1.9, 2.0, 2.4, 2.0, 2.6, 1.3, 1.7, 1.6, 1.5, 1.9, 2.4, 2.1, 2.3, 1.8, 1.4, 1.9, 1.2, 1.3, 1.9, 1.42, 1.47, 1.4, 1.9, 2.0, 2.0, 2.4, 1.9, 2.0, 2.4, 2.0, 1.98, 2.2, 1.6, 2.4, 2.6, 2.0, 1.6, 1.7, 1.9, 2.2, 1.86, 1.4, 1.9, 1.7, 1.6, 2.3)

density_values <- density(data, kernel = "gaussian")

plot(density_values)

Plot 2

library(ggplot2)

library(kernlab)

data <- c(1.2, 1.4, 2.6, 2.0, 1.4, 1.7, 1.6, 1.5, 1.48, 1.6,2.2, 1.35, 1.35, 1.2, 1.6, 1.2, 1.6, 1.2, 2.0, 1.4, 1.7, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 2.0, 1.9, 1.4, 2.0, 1.4, 1.7, 1.9, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 1.4, 2.4, 1.6, 2.4, 2.0, 1.4, 1.6, 1.8, 1.2, 2.0, 2.2, 1.8, 1.9, 2.0, 2.3, 1.4, 1.8, 1.64, 2.0, 2.3, 1.2, 1.3, 1.9, 2.0, 2.4, 2.0, 2.6, 1.3, 1.7, 1.6, 1.5, 1.9, 2.4, 2.1, 2.3, 1.8, 1.4, 1.9, 1.2, 1.3, 1.9, 1.42, 1.47, 1.4, 1.9, 2.0, 2.0, 2.4, 1.9, 2.0, 2.4, 2.0, 1.98, 2.2, 1.6, 2.4, 2.6, 2.0, 1.6, 1.7, 1.9, 2.2, 1.86, 1.4, 1.9, 1.7, 1.6, 2.3)

density_values <- density(data, kernel = " epanechnikov")

plot(density_values)

Plot 3

library(ggplot2)

library(kernlab)

data <- c(1.2, 1.4, 2.6, 2.0, 1.4, 1.7, 1.6, 1.5, 1.48, 1.6,2.2, 1.35, 1.35, 1.2, 1.6, 1.2, 1.6, 1.2, 2.0, 1.4, 1.7, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 2.0, 1.9, 1.4, 2.0, 1.4, 1.7, 1.9, 1.6, 2.0, 2.4, 1.8, 1.6, 1.64, 1.3, 1.4, 2.4, 1.6, 2.4, 2.0, 1.4, 1.6, 1.8, 1.2, 2.0, 2.2, 1.8, 1.9, 2.0, 2.3, 1.4, 1.8, 1.64, 2.0, 2.3, 1.2, 1.3, 1.9, 2.0, 2.4, 2.0, 2.6, 1.3, 1.7, 1.6, 1.5, 1.9, 2.4, 2.1, 2.3, 1.8, 1.4, 1.9, 1.2, 1.3, 1.9, 1.42, 1.47, 1.4, 1.9, 2.0, 2.0, 2.4, 1.9, 2.0, 2.4, 2.0, 1.98, 2.2, 1.6, 2.4, 2.6, 2.0, 1.6, 1.7, 1.9, 2.2, 1.86, 1.4, 1.9, 1.7, 1.6, 2.3)

density_values <- density(data, kernel = "biweight")

plot(density_values)

FOR EFFICIENCY:

To measure the performance of a kernel using the mean square error (MSE) with R studio, the following steps can be followed:

```
library(KernSmooth)

# Set up your data
data <- c(data set)

# Calculate the density estimates
density_values <- density(data, kernel = "the particular choice of kernel")
x <- density_values$x
y <- density_values$y
actual_density <- dnorm(x, mean(data), sd(data))
squared_errors <- (y - actual_density)^2
mse <- mean(squared_errors)
print(mse)
```