Non-Parametric Density Estimation and Regression

Rishab Acharya, Haoyang Wang,
Tonglu Wang, Yifei Wang,
Massimo Wu

Year 4 Project
School of Mathematics

University of Edinburgh

Abstract

This project delves into Kernel Density Estimation (KDE) and non-parametric regression, highlighting their utility in statistical analysis. Initially, we introduce KDE, discussing its effectiveness through metrics like Mean Squared Error (MSE) and Mean Integrated Squared Error (MISE), alongside asymptotic properties and bandwidth selection strategies such as rule-of-thumb, plug-in methods, and cross-validation. The selection of the Gaussian kernel is emphasized for its broad applicability.

Transitioning to non-parametric regression, comparing it to standard regression paradigms, focusing on Nadaraya-Watson, local polynomial regression, and splines. These are appraised based on their performance metrics and the process of selecting smoothing parameters, particularly highlighting the role of leave-one-out cross-validation.

Concluding, the project identifies future research directions, including extending KDE to higher dimensions, integrating with machine learning, developing methods for outlier-dense data, and testing on real-world datasets. It advocates for a holistic statistical approach, merging non-parametric techniques with other analytical methods to suit the complex data landscapes of various research domains.

Declaration

I declare that this thesis was composed by myself and that the work contained therein is my own, except where explicitly stated otherwise in the text.

(*Rishab Acharya, Haoyang Wang,*
Tonglu Wang, Yifei Wang,
Massimo Wu)

To Dr. Timothy Cannings, in recognition of your steadfast guidance and mentorship, this project is not just a culmination of our academic pursuits but a reflection of the enduring lessons you have imparted upon us. Your unwavering dedication to our growth and understanding has been the beacon that has navigated us through this academic voyage. With heartfelt gratitude, we dedicate this report to you, as a token of our appreciation for your invaluable contribution to our final year journey.

Contents

Chapter 1

Introduction

1.1 Background and Motivation

Parametric statistical methods operate under the assumption that sample data arise from a probability distribution defined by a fixed set of parameters. The predominant presumption is the normality of the data distribution. These methods are widely employed due to their statistical power. However, their applicability is sometimes limited by the stringent nature of their underlying assumptions, which may not always hold true. This limitation is particularly relevant when assessing the significance of differences within the data.

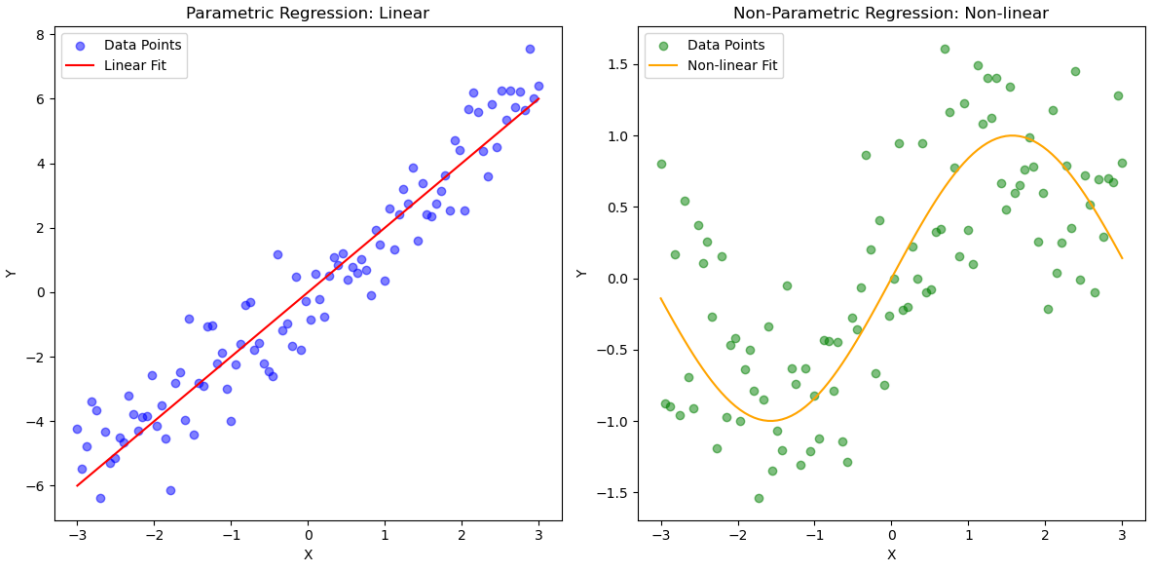


Figure 1: On the left, we have **Parametric Regression: Linear** which showcases a linear relationship between X and Y, where a straight line (red) is fitted to the data points (blue). This exemplifies a parametric approach, assuming a linear model. On the right, we have **Non-Parametric Regression: Non-linear** demonstrating a non-linear relationship, with a curve (orange) fitting the non-linearly distributed data points (green). This represents a non-parametric approach, where the model adapts to the data's inherent structure without assuming a predefined form. These plots underscore the basic differences between parametric and non-parametric methods, showing how each approach is suited to different types of data relationships.

In contrast, non-parametric statistical methods renounce the necessity for data to conform to a normal distribution. They do not rely on numerical data alone; instead, they can handle

data that is ordinal or categorical in nature, making use of rankings or presence of the shared attributes. Our motivation to study non-parametric methods, including kernel density estimation (KDE) and regression, stems from their flexibility. They discard the need for data normality, accommodating various data types, including ordinal and categorical. This adaptability is crucial for analyzing data structures that elude parametric models, enabling a more nuanced understanding of data relationships and distributions.

Non-parametric statistical methods like kernel density estimation in **Chapter 2** and non-parametric regression in **Chapter 3**, are indispensable in the field data analysis due to their ability to handle a wide range of data structures without requiring strict assumptions. They offer flexibility across diverse research fields, making them critical tools for exploratory data analysis, modeling, and inference when parametric approaches are not suitable.

1.2 History

The early development of statistical methods can be traced back to the 17^{th} and 18^{th} centuries, particularly with the discovery of the normal distribution, when Gauss [1809] and Laplace [1812], laid the foundations of probability theory. Into the 19^{th} century, methods like least squares were further developed for the analysis of measurement errors. The early 20^{th} century marked a significant milestone in parametric statistics with Fisher [1925]'s introduction of the concept of maximum likelihood estimation, along with ANOVA and hypothesis testing. By the mid-20^{th} century, parametric statistics was solidified with advancements in experimental design, estimation theory, and hypothesis testing, becoming mainstream in statistical analysis.

Non-parametric methodologies gained prominence as a response to instances where data failed to conform to the pre-requisites of parametric testing, heralding the initial adoption of such approaches. With the advancement of computational technology in the mid-20^{th} century, the feasibility of employing these methods significantly improved, leading to foundational tests by Wilcoxon [1945] and Mann and Whitney [1947]. The latter half of the 20^{th} century witnessed considerable advancements, particularly in the field of density estimation and regression analysis, signaling a phase of further evolution. Moreover, the advent and evolution of computing technology have facilitated the broad implementation of non-parametric methods, notably within data science and machine learning, symbolizing a distinctive aspect of the modern computing epoch.

1.3 Report Structure

This report offers an in-depth examination of Kernel Density Estimation (KDE) and non-parametric regression techniques. It is specifically designed to cater to third- and fourth-year undergraduate students or peers who have not previously been exposed to these critical non-parametric methods. The objective is to bridge the knowledge gap and provide a foundational understanding of how KDE and non-parametric regression can be applied to analyse data without assuming a predetermined form for the underlying distribution. Hence, the focal point of this report is predominantly on kernel density estimation.

We start with the basic formulation of KDE in section 2.1, providing intuitive understanding and examples to ease the reader into the concept. In section 2.2 we focus on the theoretical analysis of KDE's performance, evaluating it under various error criteria, including the mean integrated squared error (MISE) and asymptotic MISE. The purpose of this section is to understand how well kernel density estimation approximates the true underlying density function given some conditions/assumptions. Building upon the error criteria discussed, we introduce and derive

various bandwidth selection methods in section 2.3 such as rule-of-thumb choices or plug-in and cross-validation methods aimed at minimizing the error. We conclude with a simulation study that compares KDE's performance across different bandwidth selectors, illustrating their impact on KDE's accuracy and effectiveness by making use of the mathematical machinery developed in the previous sections. In section 2.4 we explore the significance of kernel choice in KDE, emphasizing the role of kernel differentiability in some bandwidth selection. Through comparative analysis with the theoretically optimal Epanechnikov kernel, we argue the relative insignificance of kernel choice for practical applications, advocating for the Gaussian kernel due to its versatility and suitability across most scenarios. Finally, this chapter concludes with a brief discussion on some advanced topics in section 2.5 such as multivariate KDE and KDE with boundary correction, hinting at potential areas for future research and exploration.

Chapter 3 delves into non-parametric regression techniques. We begin with Section 3.1, revisiting foundational concepts and prevalent approaches in parametric regression. The narrative then transitions to Section 3.2, presenting a suite of non-parametric methods such as Nadaraya-Watson and local polynomial regression, along with spline regression, enriched by their mathematical underpinnings. Sections 3.3 and 3.4 further expand on non-parametric regression by discussing the assessment of estimator performance and the selection of tsmoothing parameters, with a focus on leave-one-out cross-validation (LOOCV), which stands as a pivotal element in this statistical domain.

The code utilized for generating the plots and conducting the simulation study is accessible at `https://github.com/abalone88/nonparam-stats-proj`

We also have an Appendix, and much of the content in it is dedicated to enriching the main report by providing deeper insights, demonstrations, and detailed mathematical justifications that are essential for a comprehensive understanding of the topics discussed, without overshadowing the primary objective of the report.

Chapter 2

Kernel Density Estimation

Kernel density estimation (KDE) is a crucial method for estimating the probability density function, providing a significant advantage in analyzing probability distributions compared to traditional histograms. Unlike histograms, which are less smooth and rely on binning data, KDE offers a smooth approximation of the probability density function (pdf), taking into account the location of every sample point and more effectively indicating the presence of multiple modes. When applied in two dimensions, KDE outperforms 2D histograms even further by eliminating the need to specify the orientation of bins. Two key factors are essential in KDE: the shape of the kernel function and the smoothness coefficient, with the latter being especially important to the technique's effectiveness.

Consider $\{x_i\}_{i=1}^n$ independent and identically distributed, sampled from an unknown probability density function (pdf) f. The objective of density estimation, as discussed herein, is to construct an estimate $\hat{f}$ of the true density function f based on the observed data.

In the field of density estimation, one approach is the parametric method, which assumes the data derived from a specified parametric family of distributions, such as the normal distribution with mean μ and variance σ^2. In this scenario, the density f could be estimated by computing $\hat{\mu}$ and $\hat{\sigma}^2$ from the data and substituting these estimates into the formula for the normal density function. Nonetheless, this chapter emphasizes non-parametric approaches that make less stringent assumptions about the data's distribution. Although it is presumed that the distribution has a probability density f, the estimation of f is primarily guided by the data itself, rather than restricting f to a predetermined parametric family.
Kernel density estimation (KDE) is probably the most well-known non-parametric density estimation method as it is robust to outliers in the data, making them more reliable in practice where the data is noisy. It is used to create a smooth curve that represents the distribution of any given dataset.
It is clear that in the example shown in Figure 2 with the income distribution, the simple parametric approach fails to capture the distribution's skewness and bimodality. While it's possible to use a more complex parametric approach, for example, fitting with a mixture of Gaussian, this approach itself has its challenges and is also specific to this case. Hence this illustrates the flexibility in the use of kernel density estimation as it can generalise the estimation process which does not need to be tailored to specific cases. In addition, individual data points are shown as vertical black lines, illustrating the KDE's ability to fit individual observations, which is critical when the dataset is small. This comparison highlights the advantage of KDE in accurately representing complex data distributions, particularly when the true underlying distribution deviates significantly from the normal distribution assumed by the parametric

method.

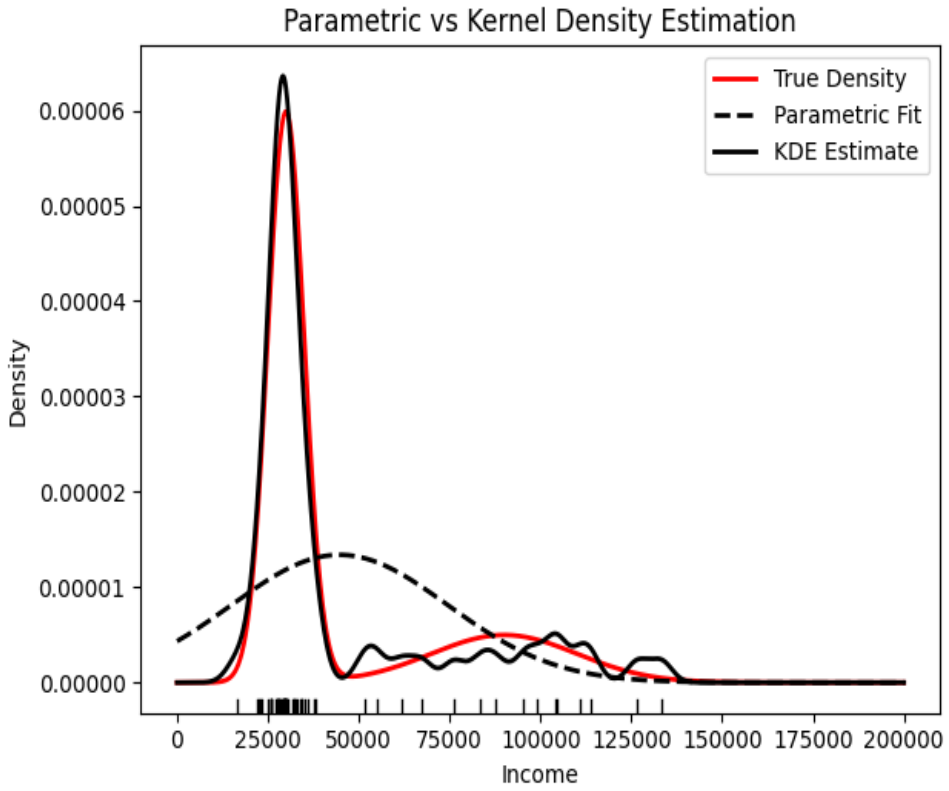


Figure 2: Comparison of density estimation methods using a mixture of two normal distributions for household income distribution. The red curve (True Density) is based on a synthetic income dataset generated from a mixture of two normal distributions: 45 data points from $N(30000, 5000^2)$ and 15 data points from $N(90000, 20000^2)$, simulating a bimodal income distribution with a pronounced peak at lower income and a long tail towards higher income. The black dashed line represents the Parametric Fit, assuming a single normal distribution with a mean and standard deviation estimated from the data. The solid black line is the kernel density estimate, closely following the true density and adapting to the skewness and long tail of the income distribution.

2.1 Formulation of Kernel Density Estimation

We denote the PDF of a uni-variate continuous random variable by X by f_X, and define

$$Pr(a \leq X \leq b) = \int_a^b f_X(x)\, dx,$$

to be the probability that X falls within the interval (a, b) where $a, b \in \mathbb{R}$ such that $a < b$. Thus, the PDF represents the likelihood of X taking on specific (range of) values, providing a smooth representation of the data distribution. Below we define the relevant components used in KDE and discuss the intuition afterwards.

Definition 1. *A function $K : \mathbb{R} \to \mathbb{R}$ is called a **kernel** if it satisfies the following conditions:*

1. *Non-negativity: $K(x) \geq 0$ for all $x \in \mathbb{R}$.*
2. *Normalisation: $\int_{-\infty}^{\infty} K(x)\, dx = 1$.*
3. *Symmetry: $K(x) = K(-x)$ for all $x \in \mathbb{R}$.*

Remark 1. *A natural choice for a kernel is the PDFs of symmetric distributions, such as the Gaussian PDF with mean zero, which inherently satisfies the above definition.*

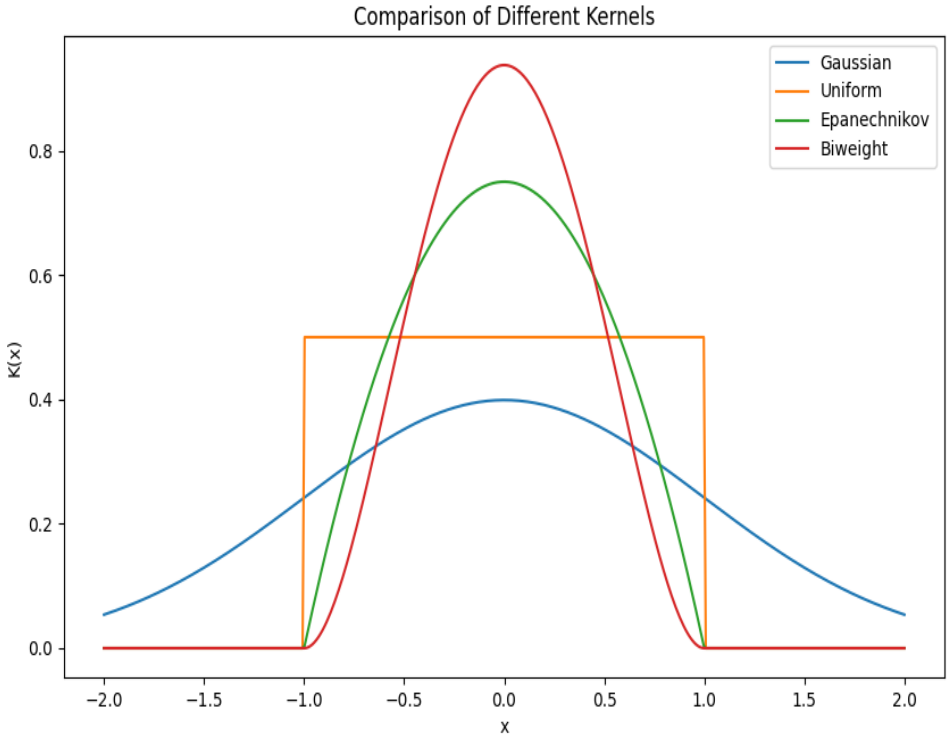


Figure 3: Illustration of some univariate kernels that satisfy definition 1: Gaussian kernel with $\sigma = 1$; the uniform kernel; the Epanechnikov kernel and the biweight kernel. We discuss the choice of kernel in section 2.4.

Definition 2. *The bandwidth, denoted as* $\mathbf{h}$*, is a positive real number that scales the kernel function.*

Remark 2. *The choice of bandwidth is a big topic on its own and we provide a detailed discussion in section 2.3.*

Definition 3. *Given data* $X_1, \ldots, X_n$ *independent and identically distributed from some density function* f*, a kernel* K *and a bandwidth* $h > 0$*, the* ***kernel density estimator*** *is defined to be*

$$\hat{f}_n(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h} K\left(\frac{x - X_i}{h}\right)$$

Intuitively, the kernel density estimator works by placing a smooth, symmetric kernel, like a weighted bump, on each data point and then summing these kernels to form a density curve as shown in Figure 4. Hence the kernel density estimator embodies the principle that each data point provides information about the location of the underlying distribution. By aggregating this information across all points through the use of kernels, KDE can smooth out randomness and noise, thus offering a clear picture of the data distribution. This also motivates the symmetry property of a kernel as stated in definition 1: we want the estimation to not be biased in either direction from the point where each kernel is centered.

On the other hand, the width of these kernels, or bumps, is controlled by the bandwidth, which determines how much influence each data point has on the shape of the density curve. A smaller bandwidth means each data point has a more localized impact, leading to a bumpier density estimate that closely follows the individual data points. In contrast, a larger bandwidth smooths out the curve, which can be better for highlighting the general shape of the data distribution but may obscure finer details.

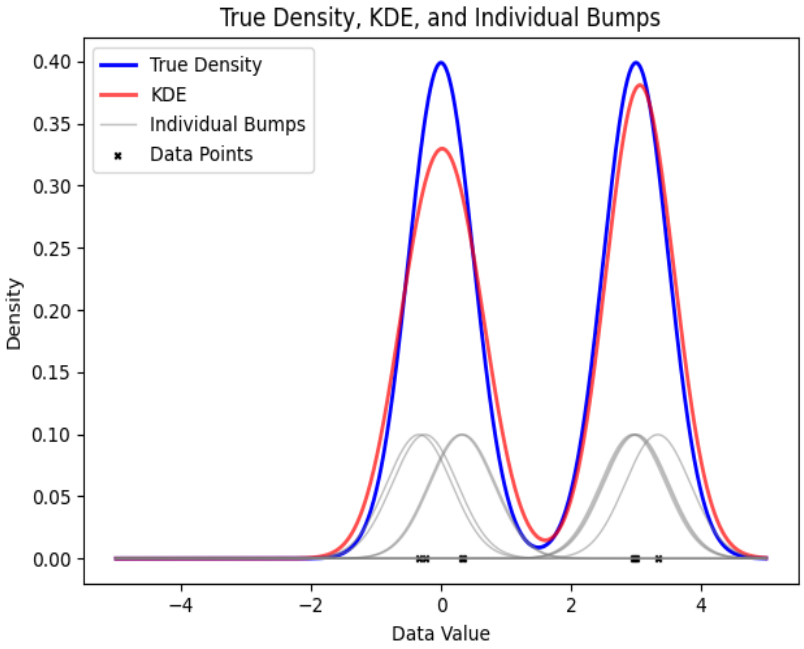


Figure 4: Illustration of the kernel density estimator as the sum of individual bumps, each centered at a data point, where the data points are drawn from a mixture of Gaussian distribution using a Gaussian kernel and fixed bandwidth.

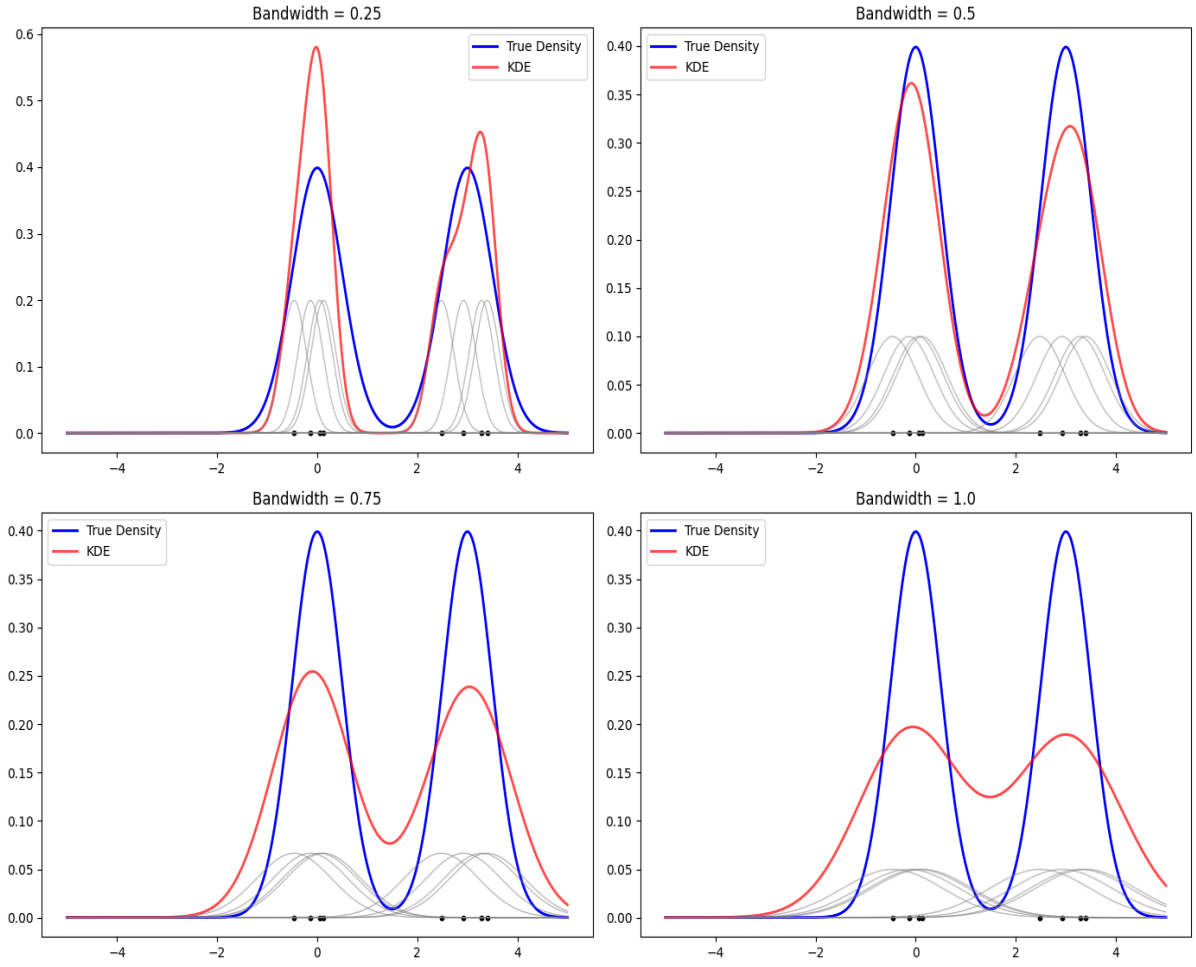


Figure 5: Illustration of the choice of bandwidth on the kernel density estimator.

As shown in Figure 5, the choice of bandwidth significantly affects the KDE's appearance and how well it approximates the true underlying density of the data.

2.2 Analysis of KDE Performance

To evaluate the performance of kernel density estimation, it is necessary to define suitable error criteria that can measure how close the estimated density $\hat{f}$ is to the actual density f_X, given independent $X_1, \dots X_n, \sim f_X$. We will later use the results obtained in this section to guide bandwidth selection in section 2.3.

2.2.1 Mean Squared Error

One of the most frequently used *local error criterion* is the Mean Squared Error (MSE), due to its decomposition into a variance and squared bias term.

Definition 4. *For $x \in \mathbb{R}$, the **mean squared error** of a kernel density estimator $\hat{f}$ with respect to an unknown true density function f_X is given by*

$$MSE(\hat{f}) = \mathbb{E}\left[(\hat{f}(x) - f_Z(x))^2\right]$$

Lemma 1. *The MSE can be written as a sum of the variance of $\hat{f}$ and its squared bias.*

Proof. Write

$$\begin{aligned}
\mathrm{MSE}(\hat{f}) &= \mathbb{E}[(\hat{f} - f_X)^2] \\
&= \mathbb{E}\left[\left(\hat{f} - \mathbb{E}[\hat{f}] + \mathbb{E}[\hat{f}] - f_X\right)^2\right] \\
&= \mathbb{E}\left[\left(\hat{f} - \mathbb{E}[\hat{f}]\right)^2\right] + 2\mathbb{E}\left[\left(\hat{f} - \mathbb{E}[\hat{f}]\right)\left(\mathbb{E}[\hat{f}] - f_X\right)\right] + \mathbb{E}\left[\left(\mathbb{E}[\hat{f}] - f_X\right)^2\right] \\
&= \mathrm{Var}(\hat{f}) + \left(\mathrm{bias}(\hat{f})\right)^2,
\end{aligned}$$

since $\mathbb{E}\left[\left(\hat{f} - \mathbb{E}[\hat{f}]\right)\left(\mathbb{E}[\hat{f}] - f_X\right)\right] = 0$. This completes the proof. □

Remark 3. *The MSE decomposition is a typical demonstration of the bias-variance tradeoff. It's often true that with more complex models, the bias decreases while the variance increases, leading to a tradeoff in determining the optimum level to minimize the MSE.*

We also introduce a handy concept of kernels that will help in analyzing KDE performance.

Definition 5. *A kernel K is said to be of* **m-*th order*** *if:*

- *$m = 0$: K satisfies the normalization condition in definition 1.*
- *$m = 1$: K satisfies condition of zero-th order and $\int_{-\infty}^{\infty} xK(x)\,dx = 0$.*
- *$m \geq 2$: K satisfies conditions for all orders up to $m - 1$ and*

$$\begin{aligned}
&\int_{-\infty}^{\infty} x^j K(x)\,dx = 0, \quad \text{for } j = 1, 2, \dots, m-1, \\
&\int_{-\infty}^{\infty} x^m K(x)\,dx \neq 0
\end{aligned}$$

Example 1. *The Gaussian kernel, given by $K(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}x^2\right)$ is a second order kernel since it is non-negative by definition; integrates to 1 since it's a probability density function; its first moment $\int xK(x)\,dx$ is zero since it has mean zero and it's variance is positive.*

Next, we show that the error of KDE is bounded with respect to MSE, adapted from Proposition 1.1 and 1.2 of Tsybakov [2009]. We provide a discussion of the assumptions after the proof.

Theorem 1. *Let $X_1, \ldots, X_n \overset{iid}{\sim} f_X$ and define $\hat{f}_n(x)$ to be a kernel density estimator as in definition 3 with given bandwidth $h > 0$ and a second-order kernel K. Further assume $f_X(x) \leq C_X$, $f_X''(x) \leq M$ are bounded for all x; $R(K) = \int_{-\infty}^{\infty} K^2(u)\,du < \infty$, and finite variance $\sigma_K^2 = \int u^2 K(u)\,du < \infty$. Then the MSE is bounded:*

$$MSE(\hat{f}_n(x)) \leq \frac{C_X R(K)}{nh} + \frac{h^4 M^2 \sigma_K^4}{4}$$

Proof. Fix $x \in \mathbb{R}$. Then

$$\begin{aligned}\mathrm{MSE}(\hat{f}_n(x)) &= \mathbb{E}\left[\left(\hat{f}_n(x) - f_X(x)\right)^2\right], \\ &= \mathrm{Var}\left[\hat{f}_n(x)\right] + \left(\mathrm{bias}\left[\hat{f}_n(x)\right]\right)^2\end{aligned}$$

Now we measure the RHS. The variance of the kernel density estimator is

$$\begin{aligned}\mathrm{Var}\left[\hat{f}_n(x)\right] &= \mathrm{Var}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K\left(\frac{x - X_i}{h}\right)\right], \\ &= \frac{1}{n^2h^2}\sum_{i=1}^{n}\mathrm{Var}\left[K\left(\frac{x - X_i}{h}\right)\right], \\ &= \frac{1}{nh^2}\mathrm{Var}\left[K\left(\frac{x - X_i}{h}\right)\right].\end{aligned}$$

We attempt to find a bound on the variance on the kernel K. We have

$$\begin{aligned}\mathrm{Var}\left[K\left(\frac{x - X_i}{h}\right)\right] &= \mathbb{E}\left[\left(K\left(\frac{x - X_i}{h}\right) - \mathbb{E}\left[K\left(\frac{x - X_i}{h}\right)\right]\right)^2\right], \\ &= \mathbb{E}\left[K^2\left(\frac{x - X_i}{h}\right)\right] - \left(\mathbb{E}\left[K\left(\frac{x - X_i}{h}\right)\right]\right)^2.\end{aligned}$$

Since the second term is always non-negative we have

$$\begin{aligned}\mathrm{Var}\left[K\left(\frac{x - X_i}{h}\right)\right] &\leq \mathbb{E}\left[K^2\left(\frac{x - X_i}{h}\right)\right] = \int_{-\infty}^{\infty} K^2\left(\frac{x - X_i}{h}\right) f_X(z)\,dz, \\ &\leq h\int_{-\infty}^{\infty} K^2(u) f_X(x - uh)\,du, \\ &\leq C_X h\int_{-\infty}^{\infty} K^2(u)\,du \leq C_X h R(K).\end{aligned}$$

Similarly, we measure the bias:

$$\mathrm{bias}\left[\hat{f}_n(x)\right] = \mathbb{E}[\hat{f}_n(x)] - f_X(x) = \mathbb{E}\left[\frac{1}{h}K\left(\frac{x - X_i}{h}\right) - f_X(x)\right].$$

Since $\int_{-\infty}^{\infty} \frac{1}{h}K(\frac{x-z}{h}) f_X(x)\,dz = f_X(x)$ which can be verified using substitution, it follows that

$$\mathrm{bias}\left[\hat{f}_n(x)\right] = \int_{-\infty}^{\infty}\frac{1}{h}K\left(\frac{x - z}{h}\right)\left[f_X(z) - f_X(x)\right]dz.$$

Rewriting the integral again using the substitution $u = (x - z)/h$ we have

$$\text{bias}\left[\hat{f}_n(x)\right] = -\int_{-\infty}^{\infty} K(u)\Big[f_X(x - hu) - f_X(x)\Big]\, du.$$

Applying Taylor's theorem on $f_X(x - hu) - f_X(x)$ with first order expansion we obtain

$$\text{bias}\left[\hat{f}_n(x)\right] = \int_{-\infty}^{\infty} K(u)\Big[-huf'_X(x) + \frac{1}{2}h^2u^2f''_X(\tilde{x})\Big]\, du, \quad \text{where } \tilde{x} \in (x, x - hu).$$

By assumption the second derivative $f''_X(\tilde{x}) \leq M$ is bounded. Hence we have

$$\text{bias}\left[\hat{f}_n(x)\right] \leq -huf'_X(x)\int_{-\infty}^{\infty} K(u)\, du + \frac{1}{2}h^2M\int_{-\infty}^{\infty} u^2K(u)\, du \leq \frac{h^2M\sigma_K^2}{2}.$$

where in the last step we used $\int uK(u)\, du = 0$ and finite variance of a second-order kernel. The result follows by combining the two bounds:

$$\begin{aligned}
\text{MSE}(\hat{f}_n(x)) &= \text{Var}\left[\hat{f}_n(x)\right] + \left(\text{bias}\left[\hat{f}_n(x)\right]\right)^2, \\
&\leq \frac{1}{nh^2}\,\text{Var}\left[K\left(\frac{x - X_i}{h}\right)\right] + \left(\frac{h^2M\sigma_K^2}{2}\right)^2, \\
&\leq \frac{C_X R(K)}{nh} + \frac{h^4M^2\sigma_K^4}{4}.
\end{aligned}$$

□

We now discuss the assumptions imposed in proving Theorem 1. The boundedness of the true density function f_X and its second derivative restricts the function's behaviour, which prevents pathological cases. For example, it prevents the true density function f_X from exhibiting extreme behaviour such as sharp spikes or rapid oscillations, which can cause instability in the estimation process since they could lead the KDE to be overly sensitive to small variations in the data as shown in Figure 6 in the subsequent page. Therefore, this assumption ensures that the KDE yields a smooth and stable estimate that reflects the underlying distribution and is robust to small changes in the sample, and to generalize better the underlying population from which the data are drawn.

The remaining two assumptions are concerned with the choice of kernel function itself. The reason for assuming a finite variance is clear as it affects how broadly the influence of each data point is spread, and hence a finite variance leads to a better variance-bias tradeoff. Similarly, the assumption of the kernel being square integrable ensures good mathematical stability, especially in the analysis of asymptotic behaviour in section 2.2.3 which is of crucial importance when we discuss bandwidth selection in section 2.3.

To conclude, as mentioned in remark 2, the key to minimizing the MSE is to find the right balance between bias and variance. Theorem 1 tells us that the variance component is inversely proportional to h, while the bias squared term is proportional to h^4, indicating that a larger h increases bias which dominates and increases the MSE overall. This is expected as a large value of bandwidth oversmoothes the KDE, resulting in increased bias as we will discuss further in section 2.3.

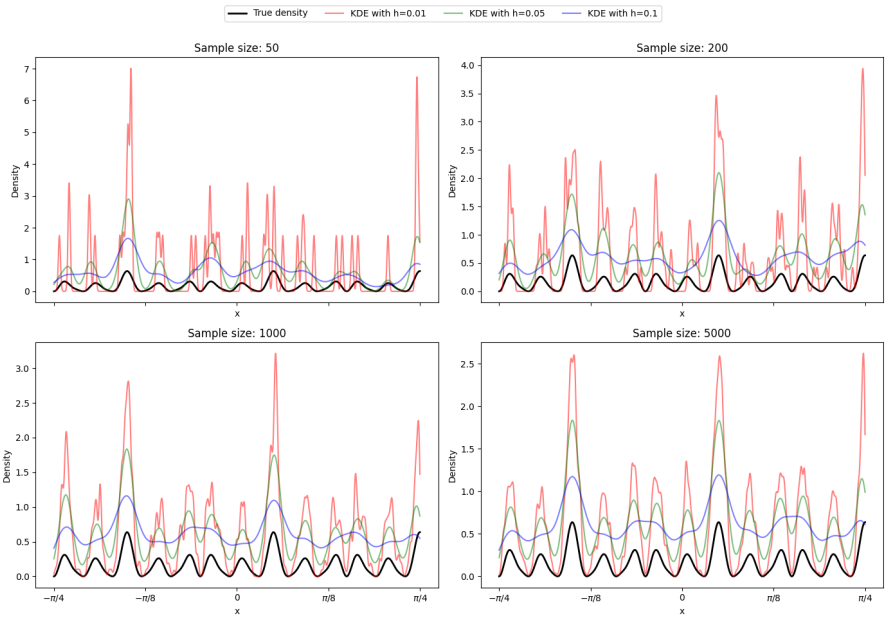


Figure 6: Illustration of a pathological density $f(x) = \frac{0.5(\sin^2(10x)+\cos^2(20x))(1+\sin(50x))}{\int 0.5(\sin^2(10x)+\cos^2(20x))(1+\sin(50x))\,dx}$ that exhibits sharp spikes and oscillations applying KDE with different bandwidths and sample sizes. The estimation becomes more accurate as we increase the sample size (reduces variance component in MSE), which is often not practical. However, even with large sample sizes, there is no single bandwidth that gives, at least visually, a *good enough* fit to the true density. When interpreting this plot and comparing the effect with different sample sizes, it's important to note the difference in the y-axis scale.

2.2.2 Mean Integrated Squared Error

One drawback of using the MSE error criterion is that it provides a local error. Hence, we can instead consider a *global error criteria* called the mean integrated squared error (MISE), which measures the expected squared difference over the entire domain.

Definition 6. *Let $\hat{f}$ be the kernel density estimator and given bandwidth $h > 0$, the mean integrated squared error (MISE) is defined as*

$$MISE\left[\hat{f}(\cdot;h)\right] = \mathbb{E}_{f_X} \int_{-\infty}^{\infty} \left(\hat{f}(x;h) - f_X(x)\right)^2 dx$$

One reason for adopting the expected value is because the data is often considered as random variables, and hence it is more appropriate to consider its expectation rather than just evaluate the performance for a set of observations realizations.

Similar to the MSE, the MISE can also be decomposed into terms involving the variance and bias squared.

Lemma 2. *The MISE can be decomposed into variance and squared bias terms*

Proof. Since the integrand for the MISE is non-negative, by Tonelli's Theorem, presented as Theorem 18.3 in Billingsley [2012], we can change the order of integral and expectation in the

expression of MISE and then the result follows:

$$\begin{aligned}\mathrm{MISE}\left[\hat{f}(\cdot;h)\right] &= \mathbb{E}_{f_X}\int_{-\infty}^{\infty}\left(\hat{f}(x;h)-f_X(x)\right)^2 dx,\\ &= \int_{-\infty}^{\infty}\mathrm{MSE}(\hat{f}(x;h))\,dx,\\ &= \int_{-\infty}^{\infty}\mathrm{Var}(\hat{f}(x;h))\,dx + \int_{-\infty}^{\infty}\left(\mathrm{bias}(\hat{f}(x;h))\right)^2 dx.\end{aligned}$$

□

Analogous to the MSE case, we analyze the variance and bias squared terms to obtain a bound on the MISE.

Proposition 1. *Assume conditions of Theorem 1. Then given bandwidth $h > 0$ we have*

$$\int_{-\infty}^{\infty} Var(\hat{f}(x;h))\,dx \le \frac{1}{nh}R(K)$$

Proof. Following the proof in Theorem 1 where we have shown that for all $x \in \mathbb{R}$

$$\mathrm{Var}(\hat{f}(x;h)) \le \frac{1}{nh^2}\mathbb{E}\left[K^2\left(\frac{x-z}{h}\right)\right].$$

It then follows that

$$\begin{aligned}\int \mathrm{Var}(\hat{f}(x;h))\,dx &\le \frac{1}{nh^2}\int\left[\int K^2\left(\frac{x-z}{h}\right)f_X(z)\,dz\right]dx,\\ &= \frac{1}{nh^2}\int f_X(z)\left[\int K^2\left(\frac{x-z}{h}\right)dx\right]dz,\end{aligned}$$

Applying substitution for $(z-x)/h$ for the second integral and using the fact that $\int f_X = 1$ we get the result

$$\int \mathrm{Var}(\hat{f}(x;h))\,dx \le \frac{1}{nh}R(K)$$

□

The analysis of the bias squared term is more involved. We invite readers to refer to page 13 of Tsybakov [2009] for some discussions and assumptions imposed on the true density f_X, and below we outline the main result.

Proposition 2. *Let K be a second order kernel and $\sigma_K^2 = \int u^2K(u)\,du < \infty$. Under some conditions on a restricted subset of the density f_X*[1]*, for any given bandwidth $h > 0$ and $n \ge 1$ we have*

$$\int \left(bias(\hat{f}(x;h))\right)^2 dx \le \frac{L^2\sigma_K^4h^4}{4}$$

where L is a constant such that

$$\left[\int\left(f_X''(x+t)-f_X''(x)\right)^2 dx\right]^{1/2} \le L, \quad \text{for all } t \in \mathbb{R} \tag{1}$$

[1] see page 13 of Tsybakov [2009]

Remark 4. *The proof essentially starts by applying the Taylor expansion of the density f_X and then applying twice the generalized Minkowski inequality. For details, refer to proposition 1.5 of Tsybakov [2009].*
Remark 5. *Essentially, we are assuming that the true density f_X is smooth enough with respect to the L^2 norm and that f_X belongs to the Nikol'ski class $\mathcal{H}(2, L)$, which means f_X satisfies equation 1. The choice of the constant L determines how fast the (second) derivative can change. Hence, the smaller the L, the smoother the density.*
Theorem 2. *Under conditions of Proposition 1 and Proposition 2, we have an upper bound on the MISE:*

$$MISE\left[\hat{f}(\cdot; h)\right] \leq \frac{R(K)}{nh} + \frac{L^2 \sigma_K^4 h^4}{4}$$

By comparing this bound with the MSE bound from Theorem 1, we see that the upper bound on the variance component is smaller in the MISE by a factor of C_X, which is an upper bound of the true density f_X. On the other hand, the difference between their squared bias term is the restriction imposed on the second derivative of the true density. To get an equality in the two upper bounds, we might just set $L = M^2 \geq f_X''(x)$ for all x, which would definitely make $f_X \in \mathcal{H}(2, L)$.

2.2.3 Asymptotic MISE

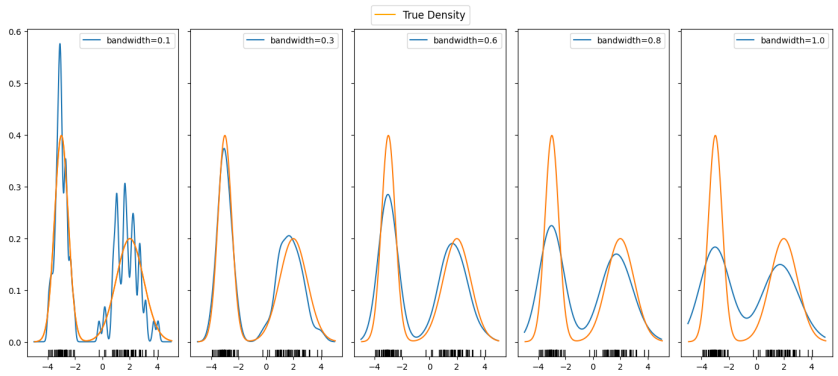
While the above analysis provides an upper bound for the chosen error criteria, we might as well consider an approximation to the exact form of the error, and this can be achieved by analyzing the asymptotic behaviour of the MISE.
Definition 7. *Assume K is a kernel satisfying $R(K) = \int K^2(u)\, du < \infty$, $\sigma_K^2 := \int u^2 K(u) > 0$ is finite and non-zero. Further assume the true density f_X is differentiable on $\mathbb{R}$ and the second derivative is L^2 integrable $R(f_X'') = \int (f_X''(x))^2\, dx < \infty$. Then the large sample approximation, or the **asymptotic MISE** of the kernel density estimator $\hat{f}(\cdot; h)$ is*

$$AMISE\left[\hat{f}(\cdot; h)\right] = \frac{1}{nh} R(K) + \frac{h^4}{4} \sigma_K^4 R(f_X'')$$

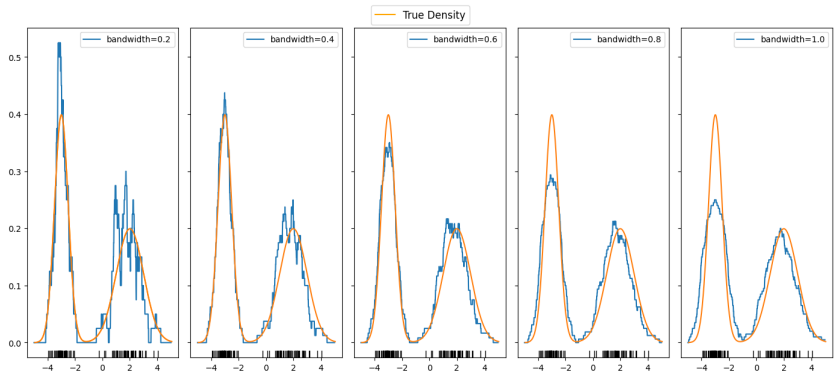
The first term is related to the variance component and the second term is related to the bias squared term. The involved proof is presented in Appendix (Proposition A.1) in Tsybakov [2009].
We postpone the analysis of the AMISE in sections 2.3 and 2.4 where we talk about the bandwidth selection methods and the existence of a theoretically optimum kernel based on the AMISE error criterion.

2.3 Univariate Bandwidth Selection

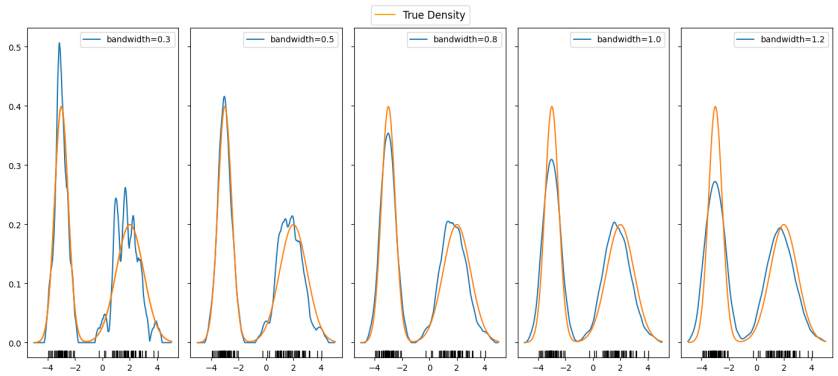
Bandwidth selection involves navigating a tradeoff between bias and variance. The objective is to identify an optimal bandwidth that achieves a *good* balance between bias and variance, thereby minimizing the discrepancy between the actual density and the KDE, according to certain loss functions. While determining such an optimal bandwidth is a complex endeavor, it has been thoroughly studied, for example, see Cao et al. [1994], and our goal is to introduce and explore some popular approaches in the subsequent sections. We also conducted a simulation study at the end of this section.
To motivate the importance of bandwidth selection, consider the following plots:
From Figure 7 We see that a good bandwidth that approximates the true density *close enough* is different for each kernel. But how does one determine the *optimum* bandwidth that gives



(a) Gaussian Kernel



(b) Uniform Kernel



(c) Epanechnikov Kernel

Figure 7: Illustration of the effect of bandwidth on KDE for the (a) Gaussian Kernel; (b) uniform kernel and (c) Epanechnikov kernel, with different bandwidth values. The KDE is plotted in blue while the true density, which is a mixture of Gaussian, is plotted in orange.

the most *accurate* estimate for the true density, without having to plot and visualize the effect of bandwidth for a range of different values, which is sometimes not possible? We explore approaches in the subsequent sections.

2.3.1 Rule-of-Thumb Selectors

Here we describe some quick and simple ways to selecting an optimum, if not, good enough bandwidth. These are known as rule-of-thumb selectors, which are usually applied to get a quick idea of what bandwidth to choose. Before doing so we first introduce the following straightforward result:

Corollary 1. *The bandwidth h that minimizes the asymptotic MISE in definition 7 is*

$$h^*_{AMISE} = \left(\frac{R(K)}{n\sigma_K^4 R(f_X'')} \right)^{1/5}$$

We can derive this result by differentiating the expression for the asymptotic MISE with respect to h and rearranging it.

2.3.1.1 Normal Scale Selector

One common choice is to assume the true density f_X is normal. This motivates the so-called normal scale selector, which replaces the unknown true density with the normal pdf, and calculates the theoretically optimum bandwidth, for example, under the asymptotic MISE.

Corollary 2. *Let f_X be Gaussian with variance σ^2 and let K be the Gaussian kernel. Then the theoretically optimum bandwidth under the asymptotic MISE is*

$$h_{NS} = 1.06 n^{1/5} \sigma$$

We show the derivation in Appendix A.1. There are various ways to estimate σ, for example, by using the sample standard deviation $\hat{\sigma} = s$. This leads to the normal scale bandwidth selector, which is usually also referred to as the rule-of-thumb selector

$$\hat{h}_{NS} = \hat{h}_{ROT} = 1.06 n^{1/5} s$$

One needs also to note that the estimate s is sensitive to the presence of outliers in the data, which can lead to a very poor bandwidth value using the normal scale selector. One alternative is to use the interquartile range IQR instead as a measure of spread. This leads to the more conservative estimate of standard deviation $\hat{\sigma} = \min(IQR, s)$.

2.3.1.2 Maximal Smoothing Principle

As the name suggests, this bandwidth selection method chooses the largest bandwidth, for example, the supremum of the set of possible bandwidths with respect to some error criteria. For example, theorem 1 in Terrell [1990] shows the following result holds

Theorem 3. *Let σ be the standard deviation of the true density f_X. Then an upper bound of the optimal bandwidth with respect to the asymptotic MISE is*

$$h^* \leq \left[\frac{243 R(K)}{35 n \sigma_K^4} \right] \sigma$$

In Appendix A.2 we show that $z(x) = \frac{35}{32}(1 - x^2)^3$ defined on $[-1, 1]$ attains the bound. The

result of Theorem 3 motivates the so-known over-smoothed bandwidth

$$\hat{h}_{OS} = \left[\frac{243R(K)}{35n\sigma_K^4}\right]\hat{\sigma},$$

where $\hat{\sigma}$ is an estimate of the standard deviation. As the name suggests, this is a conservative choice of bandwidth as this choice of bandwidth oversmoothes the KDE, as the bandwidth is chosen to be the largest amongst the "best" ones. However, this gives a good starting point, where subsequent analysis can be performed to choose a better h, for example, by visually inspecting bandwidths that are fractions of $\hat{h}_{OS}$, as it narrowed down the range of bandwidth to choose from already. Below we give an example of when the underlying distribution is actually normal:

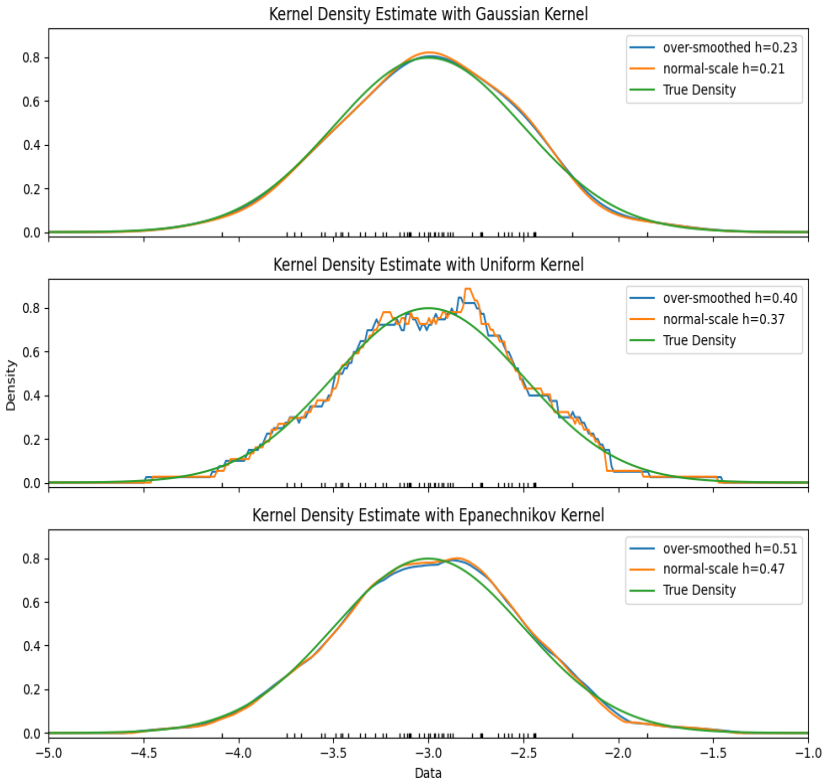


Figure 8: Illustration of the kernel density estimate using the normal-scale and over-smoothed bandwidth selectors with $n = 50$ synthetic data generated from Gaussian with $\mu = -3$ and $\sigma = 0.5$, using three different kernels.

Since the synthetic data generated came from a Gaussian distribution, the bandwidth given by the normal-scale selector seems really good as shown in Figure 8. Notice that the bandwidth values are very similar for both selectors, which is possibly due to the asymptotic nature of the maximal smoothing principle which leads to a result very close to the normal scale rule.

2.3.2 Plug-in Selectors

There are many ways to choose a bandwidth, and one option is to simply find the optimum bandwidth with respect to some error criteria, and simply plug in the values to obtain h^*, which turns out to not be very simple actually. For example, using the upper bound obtained in Theorem 1 for the MSE, we have the following straightforward result.

Corollary 3. *The bandwidth h that minimizes the upper bound of the MSE in Theorem 1 is*

$$h^* = \left(\frac{C_X R(K)}{nM^2\sigma_K^4}\right)^{1/5}$$

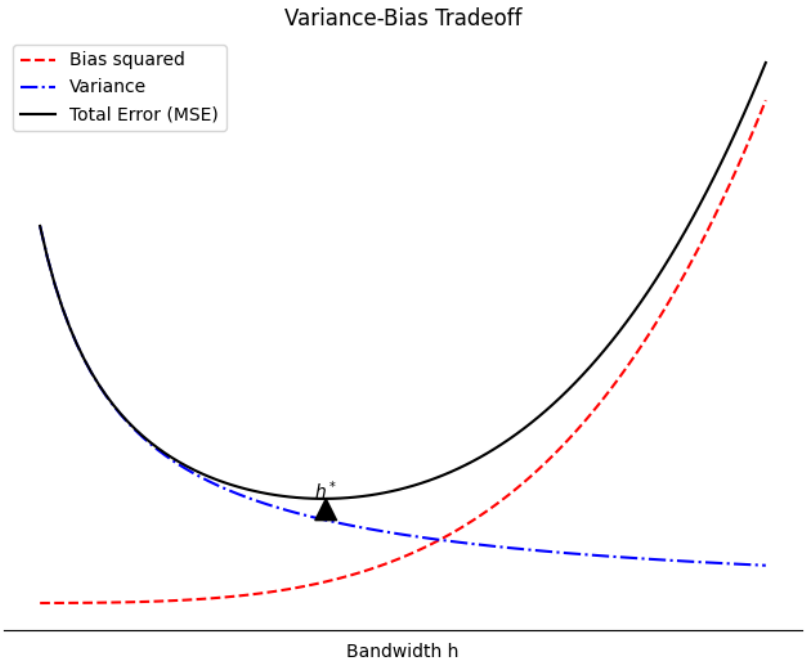


Figure 9: Illustration of the variance-bias tradeoff by plotting the upper bound, obtained from Theorem 1, of the MSE, variance, and bias squared as functions of the bandwidth h.

Alternatively, we can work with the asymptotic error criteria, which should give a better approximation of the error. For example, there also exists a similar version for the asymptotic MSE criterion, proved in section 2.3 of García-Portugués [Accessed 2024] under some further assumptions:

Theorem 4. *Let K be a kernel of order two with r derivatives. Under further assumptions, the theoretically optimum bandwidth under the asymptotic MSE criterion is*

$$h_{AMSE} = \left[\frac{2!K^{(r)}(0)}{-\sigma_K^2 R([f_X'']^{(r+2)/2})}\right]^{1/(r+3)}$$

However, this is still a local error criterion. Hence we instead consider the asymptotic MISE:

Corollary 4. *The bandwidth h that minimizes the asymptotic MISE in Theorem 4 is*

$$h^*_{AMISE} = \left(\frac{R(K)}{n\sigma_K^4 R(f_X'')}\right)^{1/5}$$

The only unknown term is $R(f_X'') = \int (f_X''(x))^2\,dx$. We will make use of the following result which gives an equivalent expression to $R(f_X'')$ and use this instead to estimate it, and in particular, also for the squared integral of higher derivatives of f_X which will also be used later to estimate $R(f_X'')$ in section 2.3.2.1 and 2.3.2.2.

Lemma 3. *Let f be a function such that f and its derivatives up to order $2s$ are continuous and integrable on $\mathbb{R}$. Further assume $f^{(i)}(x) \to 0$ as $x \to \pm\infty$ for $i = 0, 1, 2, \ldots, 2s$, where $f^{(0)} = f$. Then we have*

$$R(f^{(s)}) := \int (f^{(s)}(x))^2\,dx = (-1)^s \int f^{(2s)}(x) f(x) dx,$$

The proof is shown in Appendix A.3.

Applying Lemma 3 to f_X''in corollary 2, we obtain an equivalent form for the optimum bandwidth under asymptotic MISE:

Theorem 5. *Given a kernel K or order two. The theoretically optimum bandwidth under the asymptotic MISE is*

$$h^* = \left(\frac{R(K)}{n\sigma_K^4 \Psi_4(f_X)}\right)^{1/5}, \quad \text{where } \Psi_4(f_X) = \int f_X^{(4)}(x) f_X(x)\, dx.$$

Hall et al. [1991] introduced the following estimator for $\Psi_4(f_X)$:

$$\hat{\Psi}_4(g_4) = n^{-1} \sum_{i=1}^{n} \hat{f}^{(4)}(X_i; g_4) = n^{-2} \sum_{i=1}^{n} \sum_{j=1}^{n} L_g^{(4)}(X_i - X_j),$$

where L is a kernel and g_4 is a bandwidth which are not necessarily equal to K and h^*. This proposes other challenges. For example, if we take $L = K$ (the same kernel), then how do we choose the bandwidth g_4? If we want the best one as in the sense of Theorem 5, then the calculation of g_4^* will depend again on another unknown density function. We invite readers to refer Hall and Marron [1987] for detailed treatments in the estimation of the integrated squared density derivatives and introduce briefly how we could employ the idea in the following sections. Using the estimator $\hat{\Psi}_4(g_4)$, we have the two following bandwidth estimators: the direct plug-in(DPI) and the solve-the equation (STE) estimators.

2.3.2.1 Direct Plug-in

The direct plug-in (DPI) method would use theorem 5 with the estimator $\hat{\Psi}_4(g_4)$:

$$h_{DPI}^* = \left(\frac{R(K)}{n\sigma_K^4 \hat{\Psi}_4(g_4)}\right)^{1/5}.$$

If we use again the same kernel K in the calculation of $\hat{\Psi}_4$ and we choose the bandwidth g_4 according to the asymptotic MSE criterion in Theorem 5 we have

$$g_{AMSE}^* = \left[\frac{2K^{(4)}(0)}{-n\sigma_K^2 \Psi_6(f_X)}\right]^{1/7},$$

where the assumption required to calculate $\hat{\Psi}_4(g_4)$ is $r = 4$ in theorem 5. The problem now is clear: To find a good enough bandwidth g_4 for $\hat{\Psi}_4(g_4)$ we need to compute $\Psi_6(f_X)$, where we are going to use the estimate $\hat{\Psi}_6(g_6)$ which depends on $\Psi_8(f_X)$, which we estimate by $\hat{\Psi}_8(g_8)$whose estimate depends on $\Psi_{10}(f_X)$ so on and so forth for some other bandwidths $g_6, g_8, \ldots$. In this case, it is natural to consider using a ROT selector, such as the normal scale selector introduced earlier and thereby computing the subsequent estimates and to get $\hat{\Psi}_4(g_4^*)$ and hence the ultimate goal that we started with, h_{AMISE}^*. Using properties of a normal distribution with mean zero and standard deviation σ, Wand and Jones [1994] showed that we have

$$\Psi_r^{NS} = \frac{(-1)^{r/2} r!}{(2\sigma)^{r+1} (r/2)! \pi^{1/2}}$$

This provides a convenient starting point for calculating $\hat{\Psi}_4(g^*)$, and the following example from section 3.6.1 of Wand and Jones [1994] illustrates this

Example 2. *Illustration of the DPI selector using the same kernel $L = K$ of order two and the number of stages is 2 i.e., we need to find $\hat{\Psi}_4$ so we go two stages back, find $\hat{\Psi}_6$ and $\hat{\Psi}_8$:*

1. *Estimate* Ψ_8 *using the normal scale estimate and the formula for* Ψ_r^{NS} *with* $r = 8$*:*

$$\hat{\Psi}_8^{NS} = 105/(32\pi^{1/2}s^9),$$

   where s *is the sample standard deviation.*
2. *Estimate* Ψ_6 *using* $\hat{\Psi}_6(g_6)$ *where*

$$g_6 = \left[\frac{-2K^{(6)}(0)}{n\sigma_K^2\hat{\Psi}_8^{NS}}\right]^{1/9}$$

3. *Estimate* Ψ_4 *using* $\hat{\Psi}_4(g_4)$ *where*

$$g_4 = \left[\frac{-2K^{(4)}(0)}{n\sigma_K^2\hat{\Psi}_6(g_6)}\right]^{1/7}$$

.

Then the DPI bandwidth is given by

$$\hat{h}_{DPI,2} = \left[\frac{R(K)}{\sigma_K^4\hat{\Psi}_4(g_4)n}\right]^{1/5},$$

where the subscript of 2 indicates that we did a two-stage DPI.
This example also highlights the importance of the choice of kernel. In order to apply this two-stage DPI, we need the kernel to be at least six times differentiable, which is only attained by the Gaussian kernel amongst the ones we showed in Figure 3. Of course, there are many other kernels that satisfy this level of differentiability. A straightforward example is a sixth-order polynomial kernel $K(x) = (1 - x^2)^3$ defined for $x \in [-1, 1]$ which satisfies definition 1.
Another problem with the use of a DPI selector is the number of stages that we should take. While the two-stage DPI (and the solve-the Equation) selector, first proposed by Sheather and Jones [1991], have shown good practical results e.g., see Cao et al. [1994], there is still a lot of discussion on this topic, see for example Chacón and Tenreiro [2013], which proposed an automatic (data-based) method for choosing the number of stages to be employed, which is more reliable and outperforms the two-stage case. Such discussions are beyond the scope of this text and we encourage readers to explore these topics further on their own.

2.3.2.2 Solve-the Equation

Following the initial discussion at the beginning of this section, the solve-the equation (STE) method, also developed by Hall et al. [1991], involves solving an equation for the optimal bandwidth h_{AMISE}. Without loss of generality, assume $K = L$ for simplicity. Note that rearranging h_{AMISE} in Corollary 3 for n we obtain

$$n = \left[\frac{R(K)}{\sigma_K^4\Psi_4}\right]h_{AMISE}^{-5}.$$

Substituting this result in the expression of h_{AMSE} in Theorem 5 we obtain the following equivalent form:

$$g_{AMSE} = -\left[\frac{2K^{(4)}(0)\sigma_K^2}{R(K)}\right]^{1/7}\frac{\Psi_4}{\Psi_6}h_{AMISE}^{5/7}.$$

Thus we can write the bandwidth g as a function of h_{AMISE} by defining

$$g(h_{AMISE}) = -\left[\frac{2K^{(4)}(0)\sigma_K^2}{R(K)}\right]^{1/7} \frac{\hat{\Psi}_4(g_4)}{\hat{\Psi}_6(g_6)} h_{AMISE}^{5/7},$$

where we have again used the estimators $\hat{\Psi}_4(g_4), \hat{\Psi}_6(g_6)$ for Ψ_4 and Ψ_6 respectively and for some other bandwidths g_4, g_6, which can be chosen using a similar approach as in example 2. Thus the STE selector involves finding a (numerical) solution to the following equation for h:

$$h = \left[\frac{R(K)}{n\sigma_K^4 \hat{\Psi}_4(g(h))}\right]^{1/5},$$

which gives another "optimum" bandwidth under the asymptotic MISE criterion.

2.3.3 Cross-Validation Methods

2.3.3.1 Least Squares Validation

Least squares cross-validation is an automated, data-centric approach for determining the optimal smoothing parameter h. The core idea behind this method is to choose a bandwidth that minimizes the integrated squared error of the estimated function i.e., reducing the error in estimating the desired density function $\hat{f}$ [Li and Racine, 2011]. We consider the Integrated Squared Error (ISE) defined by

$$\text{ISE}[\hat{f}(\cdot;h)] := \int \left(\hat{f}(x;h) - f_X(x)\right)^2 dx = \int \left(\hat{f}(x;h)\right)^2 dx - 2\int \hat{f}(x;h) f_X(x)\, dx + \int (f_X(x))^2\, dx.$$

In the above equation, the third term has no dependence on h so it vanishes, hence we will minimize

$$\int \left(\hat{f}(x;h)\right)^2 dx - 2\int \hat{f}(x;h) f_X(x)\, dx$$

w.r.t h. Also note that, The cross-validation technique involves optimizing h, where the second integral, $\int \hat{f}(x;h) f_X(x)\, dx$, is equivalent to the $\mathbb{E}_{f_X}[\hat{f}(X)]$. Hence we can approximate $\mathbb{E}_{f_X}[\hat{f}(X)]$ by $n^{-1}\sum_{i=1}^n \hat{f}_{-i}(X_i;h)$, where

$$\hat{f}_{-i}(X_i;h) = \frac{1}{(n-1)h} \sum_{\substack{j=1 \\ j\neq i}}^{n} k\left(\frac{X_j - X_i}{h}\right)$$

represents the kernel estimator excluding the i-th observation, i.e, the leave-one-out kernel of $f(X_i)$. For the first integral, $\int \hat{f}(x;h)^2\, dx$, the expression simplifies to

$$\begin{aligned}\int \hat{f}(x;h)^2\, dx &= \frac{1}{n^2h^2} \sum_{i=1}^n \sum_{j=1}^n \left[\int k\left(\frac{X_i - x}{h}\right) k\left(\frac{X_j - x}{h}\right) dx\right] \\ &= \frac{1}{n^2h^2} \sum_{i=1}^n \sum_{j=1}^n \hat{k}\left(\frac{X_i - X_j}{h}\right),\end{aligned}$$

where $\hat{k}(u) = \int k(u)k(v-u)\, dv$ is the twofold convolution kernel derived from $k(\cdot)$. If $k(u) = \exp\left(-\frac{u^2}{2}\right)/\sqrt{2\pi}$, a standard normal kernel, then $\hat{k}(u) = \exp\left(-\frac{u^2}{4}\right)/\sqrt{4\pi}$, a normal kernel (i.e., normal PDF) with mean zero and variance two, which follows since two independent $N(0,1)$ random variables sum to a $N(0,2)$ random variable.

2.3.3.2 Likelihood Cross-Validation

The likelihood-based cross-validation method is a data-centric automated process used for determining the optimal bandwidth parameter, denoted as h. This technique generates a density estimate that can be interpreted through entropy theory, implying that the estimate is close to the true density in terms of Kullback-Leibler divergence. This method employs cross-validation for bandwidth selection by aiming to maximize the leave-one-out log-likelihood function, there isn't any pattern to choosing which observation to omit, hence the score function is taken as the log-likelihood average:

$$CV(h) = n^{-1} \sum_{i=1}^{n} \ln \hat{f}_{h,-i}(X_i),$$

where $\hat{f}_{h,-i}(X_i)$ signifies the density estimate computed excluding the i-th data point, refer to A.4. The primary limitation of likelihood cross-validation stems from its acute sensitivity to the tail behavior of the distribution function $f(x)$. This can engender inaccuracies, particularly with distributions that exhibit heavy tails, when conventional kernel functions are utilized.

2.3.4 Simulation Study for Bandwidth Selectors

Here we present a short simulation study to compare the performance of the univariate bandwidth selectors introduced.

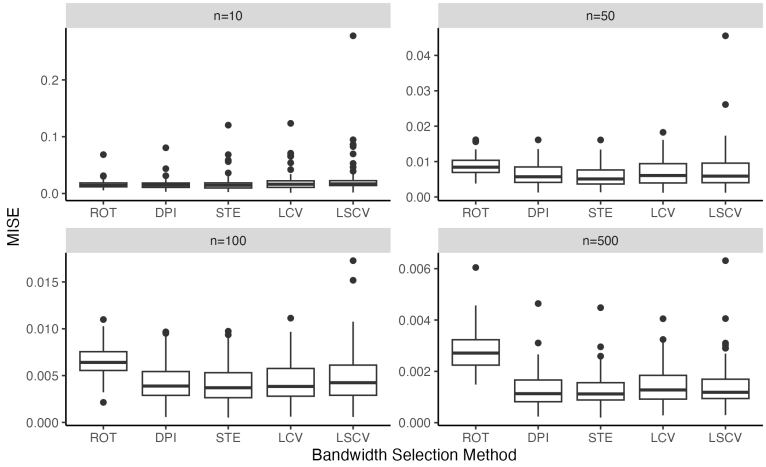


Figure 10: The figure presents a series of box plots comparing the mean integrated squared error (MISE) of kernel density estimates (KDE) across different bandwidth selection methods: Rule of Thumb (ROT), Direct Plug-In (DPI), Solve the Equation (STE), Least Squares Cross-Validation (LSCV), and Likelihood Cross Validation (LCV). The comparisons are made for varying sample sizes of $n = 10, 50, 100$, and 500 from a mixture of Gaussian distributions $N(0, 1)$ and $N(5, 2)$. Each box plot summarizes the distribution of the MISE over 100 simulations.

From Figure 10 we see that as the sample size n increases, the MISE decreases, which is in accordance with our result in Theorem 2 as increasing n reduces the component involving the variance in the upper bound of MISE. We also note that when n is small, we get similar results in terms of MISE regardless of the choice of bandwidth selector. This is because KDE is data-driven and when the sample size is small, the rule-of-thumb selectors can do equally good due to insufficient information from data. However, as n increases, we also observe that the bandwidth selectors other than the ROT all behave similarly. This is also expected since the assumption of the underlying true density being normal is often not true, hence leading to greater MISE compared with other bandwidth selectors, and the other bandwidth selectors can take advantage of more data to produce a more accurate estimate of the true density.

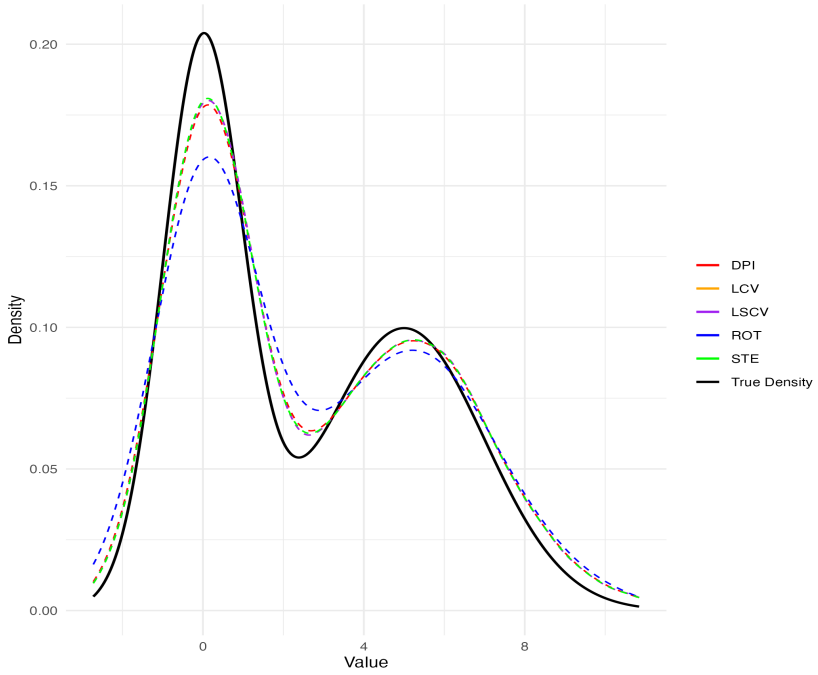


Figure 11: Comparison of kernel density estimates using different bandwidth selectors with $n = 500$. Each KDE curve represents an average of 100 simulations to provide a smoothed estimate of the underlying probability density function.
Figure 11 also confirms our analysis from Figure 10 as by visual inspection, we see that other bandwidth selectors outperform the ROT selector when sample size $n = 500$ is large, and have similar performance.

2.4 Choice of Univariate Kernels

The role of a kernel is a function that weighs the contributions of data points within its neighbourhood to the density estimate. In mathematics, there are many different definitions for a kernel, but there is actually a reason for adopting the definition we used (definition 1) – those that fail to satisfy the symmetric and unimodal properties are inadmissible as shown by Cline [1988], using the MISE as the loss function.
In this section, we present some well-known univariate kernels and compare some key characteristics between these univariate kernels. It is however important to note that the role of the kernel is much less important than the choice of bandwidth in KDE as we will discuss further below.
Recall figure 3 from section 2.1 that showed some well-known univariate kernels whose definitions will be introduced below. We observe that these kernels are very similar in shape, which is one of the reasons why the choice of the kernel is not as important in KDE because they can be made very similar to each other by adjusting the bandwidth. Nonetheless, the choice of the kernel can still have some subtle impacts on the KDE, as for example shown in Figure 7, where the estimated kernel density curve is smoother with a Gaussian and Epanchnikov kernel than with a uniform kernel, even with a *good enough* choice of bandwidth.
The **Gaussian kernel** is described by the well-known formula of a Gaussian pdf with mean zero and $\sigma > 0$, though it's common practice to set $\sigma = 1$:

$$K(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2} x^2 \right).$$

In general, we set $\sigma = 1$ and let the bandwidth h control the smoothness of the KDE.

The **uniform kernel**, a.k.a. the rectangular kernel is defined as:

$$K(x) = \begin{cases} \frac{1}{2} & \text{if } |x| \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

When used in KDE, the uniform kernel gives equal weight to all points within a bandwidth distance from the point of estimation. This results in a piecewise constant density estimate. However, the uniform kernel is not smooth, and it is not differentiable at the boundaries $x \pm 1$. This leads to a density estimate with sharp edges and flat sections, as shown in Figure 3.
The **Epanechnikov kernel** [Epanechnikov, 1969] is the theoretically optimum kernel that minimises the asymptotic MISE [Serfling, 1981], which is defined as

$$k(x) = \begin{cases} \frac{3}{4\sqrt{5}}(1 - \frac{1}{5}u^2) & \text{if } u^2 < 5.0 \\ 0 & \text{otherwise,} \end{cases}$$

The **Biweight** kernel

$$K(x) = \begin{cases} \frac{15}{16}(1 - x^2)^2 & \text{if } |x| \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

2.4.1 Smoothness

We have defined four types of kernels, each taking a slightly different shape. One might naturally ask, which shape is optimum? How do we measure this optimally? There are many different characteristics of a kernel that we can compare, and below we compare their smoothness.
Definition 8. *A kernel K is said to be* $\mathbf{C^k}$ ***smooth*** *if it belongs to the class C^k i.e., if for all $j \leq k$, the j-th derivative $K^{(j)}(x)$ exists and is continuous.*

Kernel	C^k smoothness
Gaussian	C^∞
Biweight	C^4
Epanechnikov	C^2
Uniform	C^0

Table 1: C^k smoothness of kernels

The smoothness of a kernel affects both the visual representation and the mathematical continuity of the estimated density. However, we have seen that for example in Figure 7, it suffices to have kernels with smoothness greater than C^0 for producing a smooth enough curve for the KDE, as the overall smoothness can just be controlled by the bandwidth.
What is really crucial about the smoothness of the kernel is related to the bandwidth selection, in particular, the plug-in bandwidth selectors we introduced in section 2.3.2.1 and 2.3.2.2, where we needed the kernel to be at least of C^6 smooth to conduct a two-stage plug-in, and even higher smoothness for more complex stages. Hence amongst the kernels we introduced so far, only the Gaussian kernel satisfies this level of smoothness, which is also the reason why it's usually preferred in practical implementations and applications of KDE.

2.4.2 Efficiency

Since the Epanechnikov kernel is theoretically optimal under the asymptotic MISE, even if it might not always be the best practical choice, a natural criterion for selecting kernels could be their relative efficiency compared to the Epanechnikov kernel.

Definition 9. *The efficiency metric for a kernel K, denoted as* ***Eff***(K)*, is computed relative to the Epanechnikov kernel K_{EP}, and is given by*

$$\textit{Eff}(K) = \left(\frac{\textit{MISE}_{opt}(\hat{f}) \text{ using } K_{EP}}{\textit{MISE}_{opt}(\hat{f}) \text{ using } K} \right)^{5/4}.$$

This efficiency is quantified as the ratio of the integrals involving squared kernels and their derivatives raised to the power 5/4. Despite the influence of kernel choice on the MISE, this selection is often of limited consequence to the overall efficiency.

Kernel	Relative Efficiency
Epanechnikov	1.000
Biweight	0.994
Gaussian	0.951
Uniform	0.930

Table 2: Efficiencies of several kernels relative to the Epanechnikov kernel

Table 2 from Wand and Jones [1994] indicates that the single-peaked, univariate kernels introduced yield similar results in terms of relative efficiency. Therefore, the decision among different kernels can be based on factors like computational simplicity and computational requirements, such as the smoothness of the kernel used in bandwidth selection methods. Once again, this discussion reinforces the prevalent use of the Gaussian kernel in practical applications, attributed to its C^∞ smoothness.

2.5 Selected Advanced Topics

This section offers a concise exploration of several advanced topics, designed to act as a guide for further exploration in future projects without delving into exhaustive detail.

2.5.1 Higher Order Kernels

In section 2.2.1 we defined the m-th order of a kernel in definition 5, and based our analysis of the performance of KDE using second-order kernels. Here we discuss higher-order kernels and their implications.
While the Epanechnikov kernel is of second order, there also exists a version of it but as a fourth-order kernel, defined by

$$k(u) = \begin{cases} \frac{3}{4\sqrt{5}} \left(\frac{15}{8} - \frac{7}{8}u^2 \right) \left(1 - \frac{1}{5}u^2 \right) & \text{if } u^2 < 5.0 \\ 0 & \text{otherwise,} \end{cases}$$

and the sixth-order univariate Epanechnikov kernel is given by

$$k(u) = \begin{cases} \frac{3}{4\sqrt{5}} \left(\frac{175}{64} - \frac{105}{32}u^2 + \frac{231}{320}u^4 \right) \left(1 - \frac{1}{5}u^2 \right) & \text{if } u^2 < 5.0 \\ 0 & \text{otherwise.} \end{cases}$$

The main advantage of using higher-order kernels, at least from a theoretical point of view, is that it has much better convergence properties compared with second-order kernels, which can be made as close as the parametric rate. For more discussion see section 2.8 of Wand and Jones [1994].

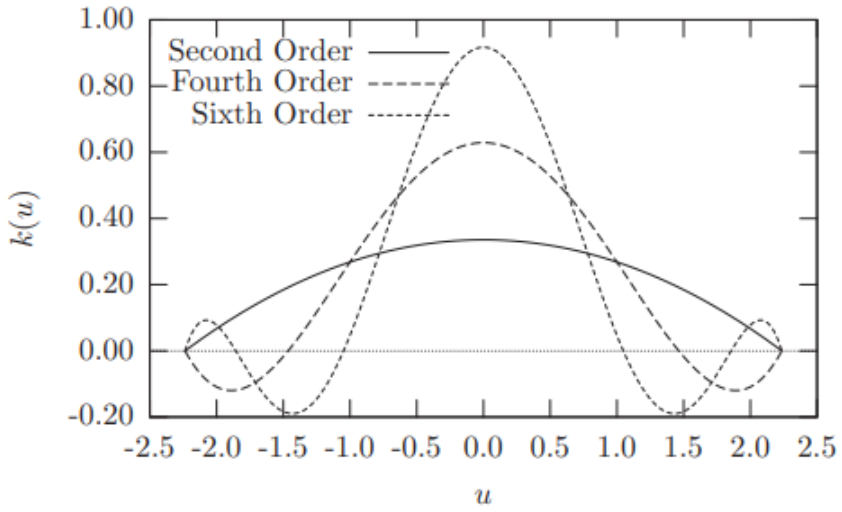


Figure 12: Epanechnikov kernel of different orders [Li and Racine, 2011]

However, Figure 12 illustrates that kernels of orders higher than two assign negative weights, leading to the possibility of negative density estimates. This outcome is not preferable because it undermines interpretability; having negative density does not logically align with the concept of density. As a result, higher-order kernels are generally not employed in practice.

2.5.2 Bandwidth Selection Based on the Rate of Convergence

Theorem 6. *If f_X is m times continuously differentiable such that $\int |f_X^{(m)}(x)|^2 dx < \infty$, then there exists a constant C_f such that for small $h > 0$*

$$\int E_f(\hat{f}(x) - f_X(x))^2 dx \leq C_f \left(\frac{1}{nh} + h^{2m} \right).$$

Furthermore, when $h_n \sim n^{-1/(2m+1)}$, we have $MISE(\hat{f}_n) = O(n^{-2m/(2m+1)})$.

Proposition 3. *From the above, we can note that $n^{-1/(2m+1)}$ is the best possible optimal rate for Theorem 3.*

The results above are adapted from Vaart [1998], which demonstrate that selecting a bandwidth that shrinks at a rate proportional to $n^{-1/(2m+1)}$ as $n \to \infty$ is one method to minimize the MISE, which ensures that the rate of convergence of the MISE is optimized.

2.5.3 Bootstrap Techniques in Bandwidth Determination

Recent advancements in the selection of bandwidth for kernel density estimation have underscored the bootstrap method's emergence as a dominant approach. This method diverges from the conventional Mean Squared Error (MSE) by adopting a bootstrapped variant, denoted as MSE*, which simplifies the optimization process [Zambom and Dias, 2012]. The exploration of different techniques includes the suggestion to resample from a reduced portion of the original dataset $X_1, \ldots, X_n$, and the proposition to compute a preliminary density using an auxiliary kernel L and an initial bandwidth b_n. This foundational step is critical for bandwidth determination, as it aids in the calculation of the scale parameter s, adjusting the bandwidth h to $h = n^{-1/5}s$.

A highlighted method proposes the expression

$$\hat{f}^*_{n,s}(x) = \frac{1}{n^{4/5}s} \sum_{i=1}^{n} K\left(\frac{x - X^*_i}{n^{-1/5}s}\right),$$

which introduces a novel bootstrapped MSE definition:

$$MSE^*_{n,s}(x) = E^*\left((\hat{f}^*_{n,s}(x) - \hat{f}_{b_n}(x))^2\right).$$

The subsequent optimal bandwidth determination is given by

$$h_n = n^{-1/5} \arg\min_s MSE^*_{n,s}.$$

The adoption of bootstrap methods has expanded widely, being applied in various fields of estimation and analysis, see for example Delaigle and Gijbels [2004].

2.5.4 Multivariate KDE

A d-dimensional multivariate kernel density estimator usually has the following form,

$$\begin{aligned}\hat{f}(\mathbf{x}, \mathbf{H}) &= n^{-1} \sum_{i=1}^{n} |\mathbf{H}|^{-1/2} K\left(\mathbf{H}^{-1/2}(\mathbf{x} - \mathbf{X}_i)\right) \\ &= n^{-1} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{X}_i)\end{aligned}$$

where,

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

The $d \times d$ bandwidth or smoothing matrix, a fixed, symmetric, positive definite matrix, is represented by $\mathbf{H}$, where d represents the problem's dimension. For each $i = 1, 2, \ldots, n$, define $\mathbf{x} = (x_1, x_2, \ldots, x_d)^{\mathrm{T}}$ and $\mathbf{X}_i = (X_{i1}, X_{i2}, \ldots, X_{id})^{\mathrm{T}}$ for $i = 1, 2, \ldots, n$. These variables represent a sequence of independently distributed and identically distributed (iid) d-dimensional random variables sampled from a typically unknown density f. The kernel functions before and after scaling are represented in this context by K and $K_{\mathbf{H}}$.

2.5.4.1 Example: Unemployment Rates and Urban Population Analysis

In this analysis, we examine data from the United States concerning urban populations, represented as the natural logarithm of city sizes, and their corresponding unemployment rates. The dataset comprises a sample size of $n = 295$ cities. Gan and Zhang [2006] hypothesized that larger cities typically have lower average unemployment rates. We illustrate the estimated joint probability density function (PDF) in Figure 13, utilizing a least squares cross-validated approach for bandwidth selection and employing a second-order Gaussian kernel. The bandwidths obtained from cross-validation were approximately 0.665 for unemployment rates and 0.351 for city sizes.

The density estimate depicted in Figure 13 from Li and Racine [2011] corroborates the assumption that metropolises are likely to experience reduced unemployment rates. Specifically, Figure 13 exhibits a distribution with a pronounced "right-angled" shape, indicating a concentration of probability mass towards lower unemployment rates for larger cities. Conversely, as urban populations decrease, there is an initial shift of the probability mass towards the origin, followed by a movement towards elevated unemployment rates as the city sizes continue to diminish.

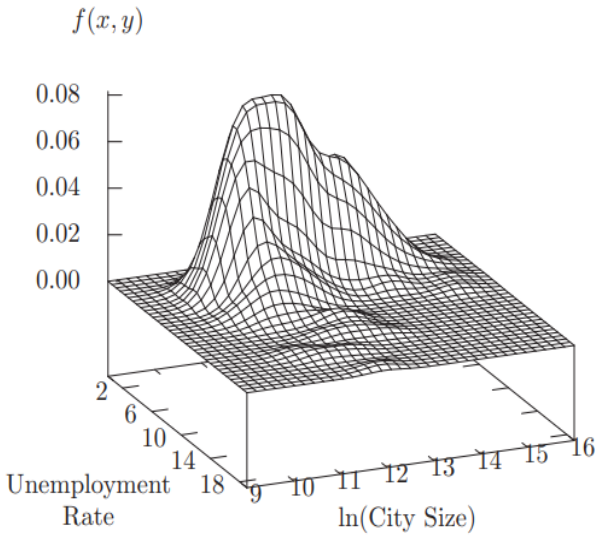


Figure 13: Unemployment rate and ln(city size) joint density estimate

2.5.5 Histogram

We note that a histogram is a fundamentally well-known non-parametric density estimate. The density estimate is computed by partitioning the probability sample space Ω into disjoint subsets or *bins* and counting the number of data points that fall into each bin. Note that the number of bins $n \in \mathbb{N}$. The $\#\{X_i \in B_l\}$ indicates the total number of data points from $X_1, X_2, \ldots, X_n$ that fit into the corresponding bin B_l of width h. Let B_l be the l-th bin. The widths of the bins are all equal. Furthermore, we present the estimate for the probability density function (PDF),

$$f(x) = \frac{\#\{X_i \in B_l\}}{nh} = \frac{k}{nh}$$

where:

- $\#\{X_i \in B_l\}$ is the number of data points X_i that fall into the bin B_l,
- n is the number of bins,
- h is the width of each bin,
- k is the total count of data points in B_l.

In this section, we will introduce the simplest and the most prevalent non-parametric estimator which is the histogram. A histogram is a basic graphical method to represent and visualize the distribution for the given dataset briefly and we don't need to make some underlying assumption before using the histogram. The construction for the histogram is also easy, assume a function f on interval $[a, b]$, then let m be an integer and we can define these bins

$$B_1 = [a, a + \frac{b-a}{m}), B_2 = [a + \frac{b-a}{m}, a + \frac{2(b-a)}{m}), ...B_m = [b - \frac{b-a}{m}, b]$$

by dividing the interval into m equal parts with binwidth $h = \frac{1}{m}$.
The histogram estimator is defined by

$$\hat{f}_n(x) = \sum_{j=1}^{m} \frac{\hat{Y}_j}{nh} I(x \in B_j)$$

and the bias of this estimator is

$$E(\hat{f}_n(x)) = E(\sum_{j=1}^{m} \frac{\hat{Y}_j}{nh} I(x \in B_j))$$
$$= \frac{1}{h} \sum_{j=1}^{m} E(\hat{p_j}(x))$$

where Y_j is the number of observations in the bin B_j and p_j is the probability that the observation in the jth bin.
And the expectation and variance of the histogram estimator $\hat{f}_n(x)$ is $E(\hat{f}_n(x)) = \frac{p_j}{h}$ and $Var(\hat{f}_n(x)) = \frac{p_j(1-p_j)}{nh^2}$.

2.5.6 Confidence Intervals and Confidence Bands

2.5.6.1 Confidence Intervals

As a statistical methodology, the kernel density estimation has its own confidence intervals. Recall the definition of the kernel density estimator is

$$\hat{f}_n(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h} K\left(\frac{X_i - x}{h}\right)$$

At the given point x, the KDE is the sample mean of Y_i where $Y_i = \frac{1}{h} K(\frac{X_i - x}{h})$.
By combining with the central limit theorem, we also can know

$$\sqrt{n}(\frac{\hat{f}_n(x) - E(\hat{f}_n(x))}{Var(Y_i)}) \to N(0, 1)$$

As h tends to zero and substitute the $Var(Y_i) = \frac{\hat{f}_n(x){\sigma_k}^2}{h}$, we can obtain the asymptotic distribution for $\hat{f}_n(x)$ is

$$\sqrt{nh}(\hat{f}_n(x) - E(\hat{f}_n(x)) \to N(0, \hat{f}_n(x){\sigma_k}^2)$$

Therefore the $1 - \alpha$ confidence interval for the KDE at the given point x is the

$$\hat{f}_n(x) \pm z_{1-\frac{\alpha}{2}} \sqrt{\hat{f}_n(x){\sigma_k}^2}$$

2.5.6.2 Confidence Bands

As we mentioned in the previous subsection, the confidence interval is suitable for one given point, x_0, and now we would like to indicate the confidence bands to provide a plausible range that is more comprehensive and global to describe the confidence intervals for all points on the estimated curve, the kernel density estimation. In many cases, these confidence intervals construct a 95 % confidence band.
A pointwise confidence band is $\hat{f}(x) \pm h(x)$ and for each point x we have

$$Pr(\hat{f}(x) - h(x) \leq \hat{f}(x) \leq \hat{f}(x) + h(x)) = 0.95$$

2.5.7 Kernel Density Estimation with Boundary Correction

2.5.7.1 Data Binning

Data binning is a type of pre-processing and evaluating the given data by grouping these observations which have similar features to reduce the errors. By identifying the type of data and choosing a suitable binning method, we assign the data points that have a similar boundary to the same bins. There are a few popular binning methods and rules, such as linear binning, simple binning, and histogram. See Scott and Sheather [1985] for an example study using data binning for KDE.

2.5.7.2 Implementing FGPAs for Bandwidth Selection

In the field of Kernel Density Estimation (KDE), handling edge effects remains a notable challenge when dealing with data having inherent boundaries. Standard KDE methods tend to be computationally rigorous and intensive, especially when computing a multitude of necessary operations and also selecting the appropriate bandwidth parameter, i.e., (scaling with time complexity $O(n^2)$), which is one barrier for KDE. Hence, we now explore the potential of leveraging **Field-Programmable Gate Arrays (FPGAs)** to expedite the process of determining this optimal bandwidth. We have adopted a commonly referenced technique known as PLUGIN for our discussion. Section 2.3.2.1 details the PLUGIN algorithm's translation into an FPGA-friendly format. We aim to present the PLUGIN method in a format that is not only optimized for FPGA applications but also maintains the use of the standard normal kernel function in its computation. All the necessary notation and details for the result of this algorithm can be found here Gramacki et al. [2015]

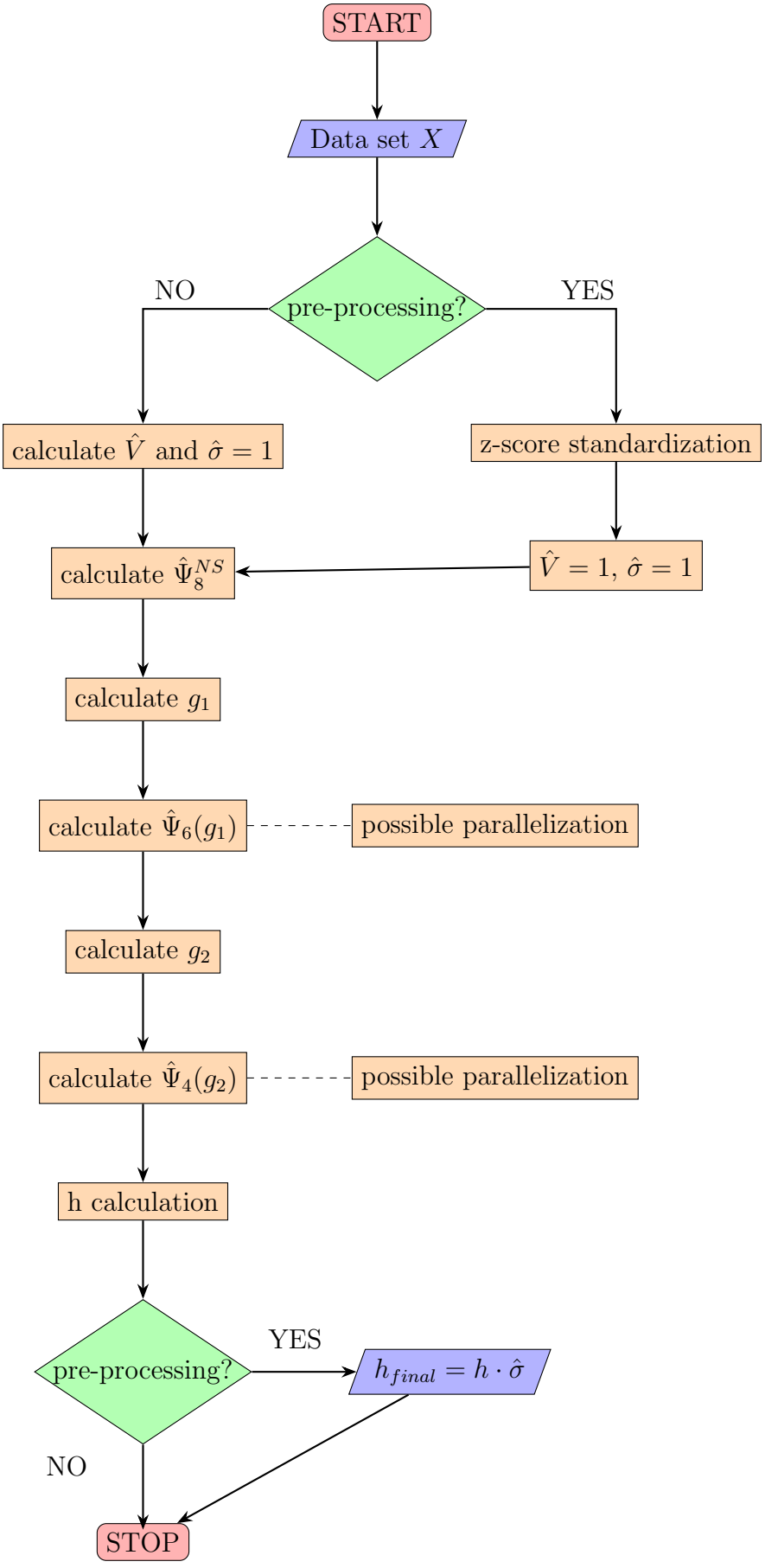


Figure 14: PLUGIN Algorithm Flowchart for FPGA implementation.

Chapter 3

Non-Parametric Regression

Regression is a method for studying the relationship between a **response variable Y** and a **covariate X**. The covariate is also called a **predictor variable** or a **feature**. One way to summarize the relationship between **X** and **Y** is through the **regression function**

$$r(x) = \mathbb{E}(Y|X = x) = \int yf(y|x)dy$$

Our goal is to estimate the regression function $r(x)$ from data of the form

$$(Y_1, X_1), \ldots, (Y_n, X_n) \sim F_{X,Y}$$

from Wasserman [2010].
In this Chapter, we take both the Parametric approach in section 3.1 and the Non-parametric approach in section 3.2 to conduct regression. Subsequently, we shall engage in an analysis of the resemblances and disparities between the two types of methodologies.

3.1 Parametric regression

In this section, we shall emphasize a couple of the most popular and familiar methods of parametric regression including simple linear regression and multiple regression, the two most commonly seen parametric regression.

3.1.1 Linear Regression

We postulate a linear function $r(x)$, signifying linearity in parameters, not variables. In the context of simple linear regression, the model presumes:

$$r(x) = \beta_0 + \beta_1 x.$$

With homoscedasticity, $\text{Var}(\epsilon_i|X = x) = \sigma^2$, the model becomes:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i,$$

assuming ϵ_i also satisfies $\mathbb{E}(\epsilon_i|X_i) = 0$.
Parameters β_0 (intercept) and β_1 (slope) are estimated by $\hat{\beta}_0$ and $\hat{\beta}_1$, forming the estimated regression line:

$$\hat{r}(x) = \hat{\beta}_0 + \hat{\beta}_1 x.$$

Optimal parameter estimation entails minimizing the RSS:

$$RSS = \sum_{i=1}^{n} (Y_i - \hat{r}(x_i))^2.$$

The least squares method yields the estimators:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2},$$
$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}.$$

Assuming normal error distribution, these estimators are also maximum likelihood estimators. For details of derivation see for example Wasserman [2010].

3.2 Non-Parametric Regression

From the previous discussion of parametric methods of conducting regression, we notice that the regression analysis is only valid and trustworthy under certain assumptions such as linearity in parameters, normality of residuals, etc. In the real-world context, these kinds of assumptions are too strict, and most of the time, the raw data collected are not able to meet these requirements. In this section, we will see several ways of conducting Regression, but **non-parametric**, which is a big class of methods that don't have to meet as demanding constraints as parametric methods do.

3.2.1 Nadaraya–Watson Regression

We begin with the kernel regression estimator. Kernel regression is a type of non-parametric estimator that is particularly useful when the form of the regression function is unknown. It does not assume a parametric model for the relationship between variables, instead estimating the regression function locally at each point of interest.
For a set of data points $Y_1, \dots Y_n$, we don't assume a model on it, but we wish to estimate the best line of fit to describe the data. Thus we might want to choose an estimator $a \equiv \hat{r}_n(x)$ to fit the line and minimize the error. We use sums of squares to measure the error; thus, we wish to minimize the equation $\sum_{i=1}^{n}(Y_i - a)^2$. Intuitively we see the solution is a constant function that $\hat{r}_n(x) = \bar{Y}$, which seems not a good estimator.
Therefore, we define the weight function $w_i = K(\frac{x_i - x}{h})$, and minimize the weighted sums of squares error $\sum_{i=1}^{n} w_i(x)(Y_i - a)^2$.
Then we minimize the equation by taking the first derivative w.r.t a, and set it to zero. Thus we have:

$$\begin{aligned} WSS &= \sum_{i=1}^{n} w_i(x)(Y_i - a)^2 \\ \frac{\partial}{\partial a} WSS &= -2 \sum_{i=1}^{n} w_i(x)(Y_i - a) \\ &\implies \sum_{i=1}^{n} w_i(x) Y_i = a \sum_{i=1}^{n} w_i(x) \\ &\implies \hat{r}_n(x) \equiv a = \frac{\sum_{i=1}^{n} w_i(x) Y_i}{\sum_{i=1}^{n} w_i(x)} \end{aligned}$$

Definition 10 (Nadaraya-Watson kernel estimator)**.** *Let $h > 0$ be a positive number, called the bandwidth. The Nadaraya–Watson kernel estimator is defined by*

$$\hat{r}_n(x) = \sum_{i=1}^{n} w_i(x) Y_i$$

where K is a kernel and the weights $w_i(x)$ are given by

$$w_i(x) = \frac{K\left(\frac{x-x_i}{h}\right)}{\sum_{j=1}^{n} K\left(\frac{x-x_j}{h}\right)}.$$

where K is the kernel function, and we have introduced some popular ones in previous sections. Note that in most cases, the choice of kernel K is not too important. Results using different kernels are usually numerically similar. What matters is the choice of bandwidth h. Larger bandwidths will generate smoother estimates while smaller ones give rough estimates. In practice, the bandwidth should depend on sample size, so we sometimes denote it as h_n.
The following theorem illustrates the impact of the bandwidth parameter on the accuracy of the estimator. To articulate these results, it is essential to posit certain assumptions regarding the behavior of $x_1, x_2, \ldots, x_n$ as n grows. For the theorem to hold, we shall assume that these observations are independently and identically distributed, drawn from a probability density function f.

Theorem 7. *The risk (w.r.t integrated squared error loss) of the Nadaraya–Watson kernel estimator is*

$$\begin{aligned} R(\hat{r}_n, r) =& \frac{h_n^4}{4} \left(\int x^2 K(x)\, dx \right)^2 \int \left(r''(x) + 2r'(x) \frac{f'(x)}{f(x)} \right)^2 dx \\ &+ \frac{\sigma^2 \int K^2(x)\, dx}{n h_n} \int \frac{1}{f(x)}\, dx + o(n h_n^{-1}) + o(h_n^4) \end{aligned}$$

as $h_n \to 0$ and $n h_n \to \infty$.

The decomposition of the risk for the Nadaraya-Watson kernel estimator reveals two principal components: the first being the squared bias, and the second, being the variance. See the discussion of this trade-off property in chapter 3.2 of Takezawa [2006].
Note a term of particular significance within the bias is

$$2r'(x) \frac{f'(x)}{f(x)},$$

This is what we call the design bias, reflecting the dependency on the sample distribution. This indicates the bias' sensitivity to the sample points' distribution. Additionally, kernel estimators are prone to increased bias at data boundaries, known as boundary bias. It is suggested that local polynomial regression can mitigate such biases.
Upon optimizing the bias-variance trade-off by differentiating and nullifying the result, the ideal bandwidth is determined to be

$$h_* = \left(\frac{1}{n}\right)^{1/5} \left(\frac{\sigma^2 \int K^2(x) dx \int dx}{f(x) (\int x^2 K^2(x) dx)^2 \int \left(r''(x) + 2r'(x) \frac{f'(x)}{f(x)} \right)^2 dx} \right)^{1/5}$$

Nadaraya-Watson estimator is a flexible and intuitive method of non-parametric regression. It

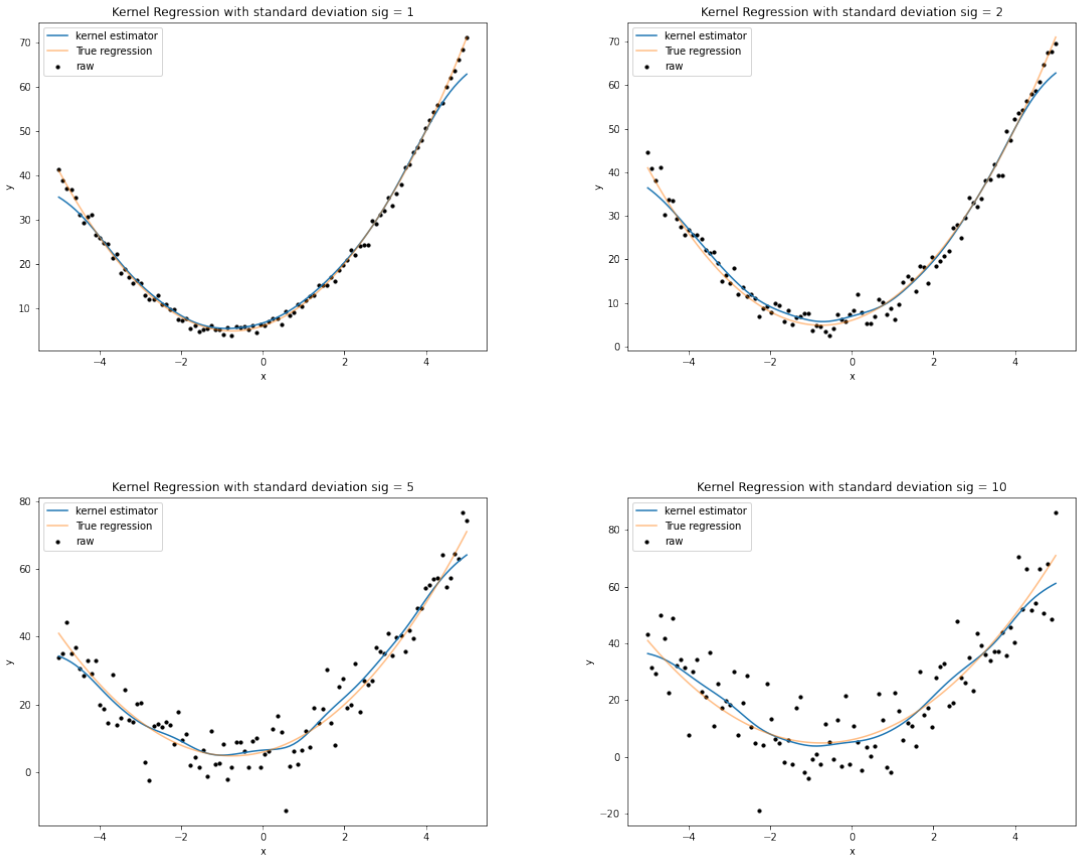


Figure 15: Nadaraya-Watson estimator using Gaussian Kernel of the simulated function with sample size $n = 100$ and bandwidth $h = 0.5$ illustrating the effects of variance term on the overall bias. $\sigma^2 = 1$ (top left), $\sigma^2 = 4$ (top right), $\sigma^2 = 25$ (bottom left), $\sigma^2 = 100$ (bottom right). The plots show an obvious worse fit as variance increases.

preserves the simplicity compared to other methods such as splines which will be discussed later in this chapter. It's also very easy to understand. However, the estimator is very sensitive to the choice of bandwidth. An inappropriate bandwidth might be destructive.

3.2.2 Local Polynomial Regression

Kernel regression can be interpreted as a local constant estimator as it's actually a locally weighted mean. It's easy to see that Nadaraya-Watson Regression is not quite flexible because it's only weighted mean. Nadaraya-Watson regression is also not a great choice when the relationship becomes complex. To capture these properties we introduce a local polynomial estimator, where we use polynomials of degree p instead of locally weighted constants.
For a fixed value x, at which we wish to estimate $r(x)$, and for values u close to x, we use the polynomial of the form:

$$P_x(u; a) = a_0 + a_1(u - x) + \frac{a_2}{2!}(u - x)^2 + \ldots + \frac{a_p}{p!}(u - x)^p$$

To obtain the local polynomial estimator, we need to estimate the coefficients $a = (a_0, \ldots, a_p)^T$ of the polynomial. This is again achieved by choosing $\hat{a} = (\hat{a}_0, \ldots, \hat{a}_p)^T$ that minimizes the

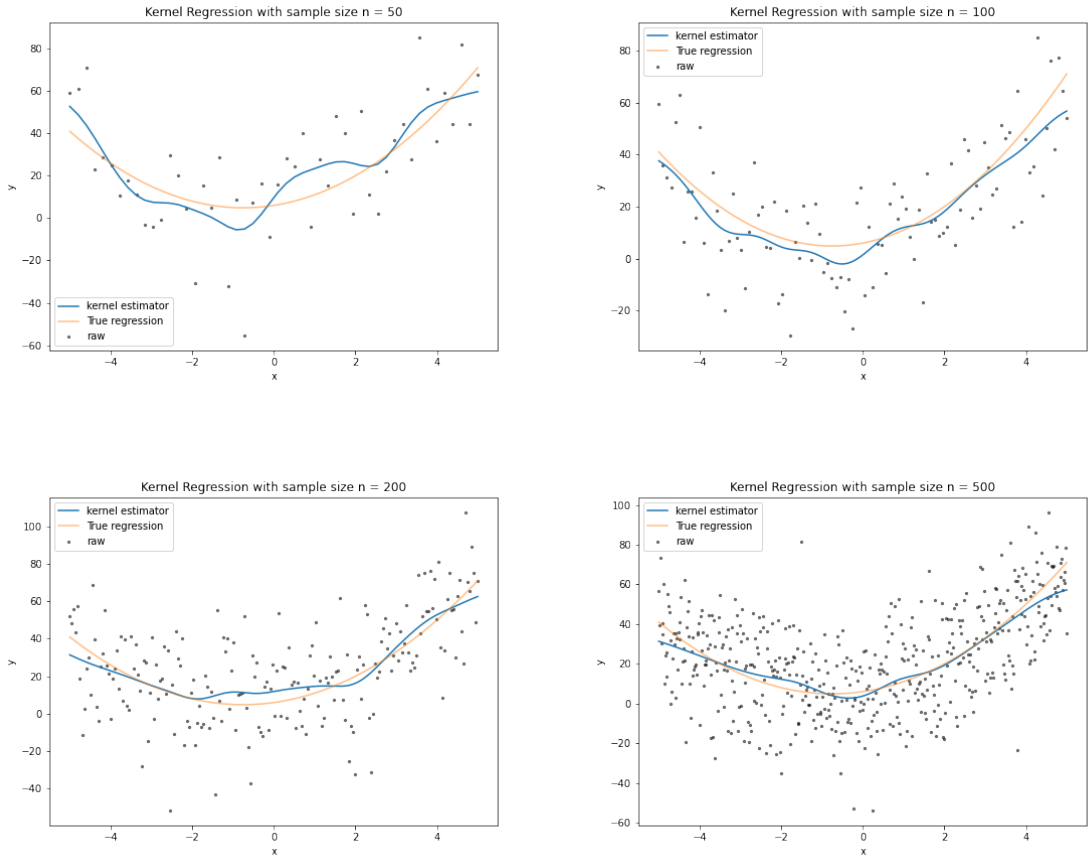


Figure 16: Nadaraya-Watson estimator using Gaussian Kernel of the simulated function with high variance $\sigma^2 = 400$ and bandwidth $h = 0.5$ illustrating the effects of sample size on the overall bias. $n = 50$ (top left), $n = 100$ (top right), $n = 200$ (bottom left), $n = 500$ (bottom right). The plots clearly show a better fit as the sample size increases.

locally weighted sums of squares:

$$\sum_{i=1}^{n} w_i(x) \left(Y_i - P_x(X_i; a)\right)^2 \tag{2}$$

Note that if we set $p = 0$, then it comes back to a local constant estimator. We now start from $p = 1$, in which case it's also called local linear regression. When we raise $p = 2, 3, \ldots$, it will become local quadratic/cubic ... regression, accordingly.
To reduce the notation, we introduce the problem in matrix form:

$$X_x = \begin{pmatrix} 1 & x_1 - x & \dots & \frac{(x_1-x)^{p-1}}{(p-1)!} & \frac{(x_1-x)^p}{p!} \\ 1 & x_2 - x & \dots & \frac{(x_2-x)^{p-1}}{(p-1)!} & \frac{(x_2-x)^p}{p!} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_{n-1} - x & \dots & \frac{(x_{n-1}-x)^{p-1}}{(p-1)!} & \frac{(x_{n-1}-x)^p}{p!} \\ 1 & x_n - x & \dots & \frac{(x_n-x)^{p-1}}{(p-1)!} & \frac{(x_n-x)^p}{p!} \end{pmatrix}, a = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{p-1} \\ a_p \end{pmatrix}$$

We also rewrite weights in matrix form as an $n \times n$ diagonal matrix, with each entry

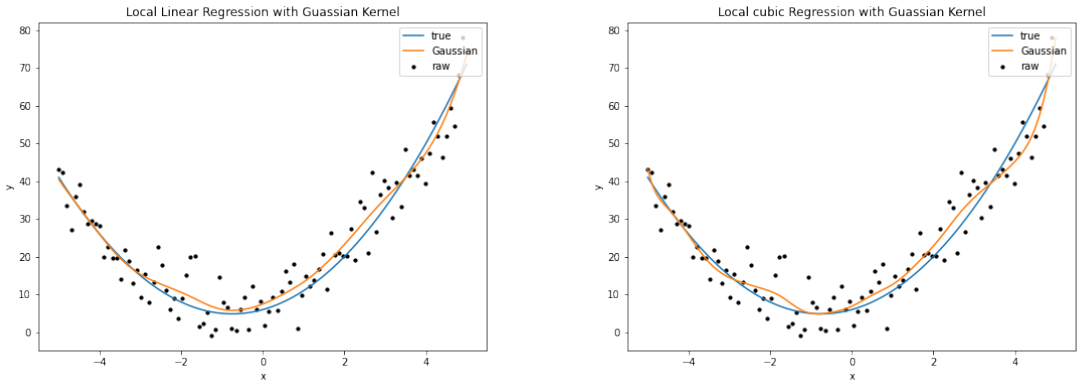


Figure 17: Local polynomial estimator using Gaussian Kernel of the simulated function with variance $\sigma^2 = 25$ and bandwidth $h = 0.5$ illustrating the effects of varying degrees of polynomial. Local linear regression (left), local cubic regression (right).

representing the weight of nth point

$$W_x = \begin{pmatrix} w(x_1, x) & 0 & \dots & 0 & 0 \\ 0 & w(x_2, x) \dots & 0 & 0 & \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & w(x_{n-1}, x) & 0 \\ 0 & 0 & \dots & 0 & w(x_n, x) \end{pmatrix}$$

Thus, we can rewrite equation 2 as:

$$(Y - X_x a)^T W_x (Y - X_x a) \tag{3}$$

Similarly, we solve the least square problem by first taking the derivative of equation 3 w.r.t a. We have the following result:

$$\frac{\partial}{\partial a}(Y - X_x a)^T W_x (Y - X_x a) = -2X_x^T W_x Y + 2X_x^T W_x X a$$

Nullifying the equation we get:

$$\begin{aligned} -2X_x^T W_x Y + 2X_x^T W_x X a &= 0 \\ \implies X_x^T W_x Y &= X_x^T W_x X a \end{aligned}$$

we further assume that $X_x^T W_x Y$ is non-singular, we have:

$$a = (X_x^T W_x Y)^{-1} X_x^T W_x X$$

Note that when $u = x$, we have $r_n(x) = a_0$. And from our calculation, this actually corresponds to the first component of $(X_x^T W_x Y)^{-1} X_x^T W_x X$, we denote it as $e_1^T (X_x^T W_x Y)^{-1} X_x^T W_x X$.

Theorem 8. *The local polynomial regression estimate is*

$$\hat{r}_n(x) = \sum_{i=1}^{n} w_i(x) Y_i$$

where $w(x)^T = (w_1(x), \ldots, w_n(x))$,

$$w(x)^T = e_1^T (X_x^T W_x X_x)^{-1} X_x^T W_x,$$

$e_1 = (1, 0, \ldots, 0)^T$ *and* X_x *and* W_x *are defined in 3.2.2.*
Notice that our estimate is also susceptible to the choice of h. We can again select the bandwidth by minimizing the cross-validation error.
Local polynomial offers more flexibility in modelling complex relationships but is also susceptible to bandwidths selection. Since local polynomials can capture local variations of data compared to the Nadaraya-Watson estimator, it may perform better at extrapolation. It's easy to see one of the limitations of local polynomials is that it's computationally ineffective at the cost of higher flexibility. Additionally, the two estimators we have seen so far focus more on the local properties of the data, they are both ignorant of global patterns.

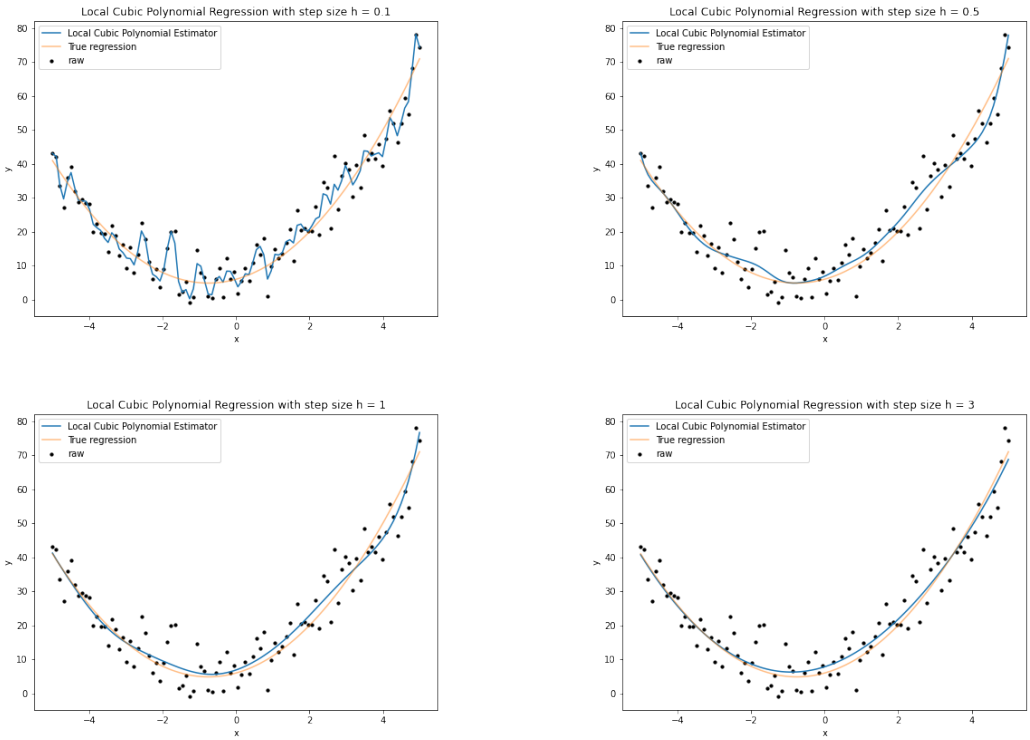


Figure 18: Local cubic estimator using Gaussian Kernel of the simulated function with fixed variance $\sigma^2 = 25$ and sample size $n = 100$ illustrating how the choice of bandwidth affects the performance. $h = 0.1$ (top left), $h = 0.5$ (top right), $h = 1$ (bottom left), $h = 3$ (bottom right). The plots clearly show smaller bandwidth leads to a wiggly fit and as bandwidth increases, the regression line clearly becomes smoother.

3.2.3 Splines

As previously mentioned, Nadaraya-Watson regression and local polynomial regression only capture local properties of the data, it suffers from ignoring global patterns. Thus it gives rise to spline regression, which can not only capture local patterns separated by different so-called knots and provide even more flexibility but also take care of the global pattern of data.
Definition 11. *Let* $\xi_1 < \xi_2 < \ldots < \xi_n$ *be a set of ordered points — called* ***knots*** *— contained in some interval* (a, b). *An* M*th-order* ***spline*** *is a piecewise* $M-1$ *degree polynomial with* $M-2$ *continuous derivatives at the knots.*

Piecewise cubic splines are most commonly used, and it's of the form:

$$r(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \sum_{j=1}^{n} b_j \left((x - \xi_j)_+\right)^3 \tag{4}$$

where $a_0, a_1, a_2, a_3, b_1, \ldots, b_n$ are coefficients to be estimated; $\{\xi_i\}_{i=1}^n$ are knots; $(x)_+$ is a function defined as:

$$(x)_+ = \begin{cases} x & \text{if} \quad x \geq 0 \\ 0 & \text{if} \quad x < 0 \end{cases}$$

It's clear that we have $n + 4$ coefficients to estimate, and to reduce the notation, we rewrite equation 4 as:

$$r(x) = \sum_{j=1}^{n+4} \beta_j h_j(x)$$

where we define $h_i(x) = x^{i-1}$ for $i = 1, 2, 3, 4$. Next, we wish to estimate the coefficients that minimize the sums of squares. Thus we express our problem as:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}, \quad \text{where } \hat{\beta} \text{ minimizes } \|\mathbf{y} - \mathbf{X}\beta\|^2$$

To avoid a wiggly fit, we then impose a constraint on the parameters:

$$\sum \beta_{3k}^2 \leq C$$

where C is a constant, and $\beta = \begin{pmatrix} \beta_0 & \beta_1 & \beta_2 & \beta_3 & \beta_{31} & \ldots & \beta_{3n} \end{pmatrix}^T$. Then we define a $(n+4) \times (n+4)$ matrix $\mathbf{D}$, such that $\mathbf{D} = \begin{pmatrix} \mathbf{0}_{4\times4} & \mathbf{0}_{4\times n} \\ \mathbf{0}_{n\times4} & \mathbf{I}_{n\times n} \end{pmatrix}$. Thus we can transfer our minimization problem to:

$$\|\mathbf{y} - \mathbf{X}\beta\|^2 \quad \text{s.t.} \quad \beta^T \mathbf{D} \beta \leq C$$

minimization problem of this kind is commonly solved by minimizing the equivalent Lagrangian function:

$$\|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda^6 \beta^T \mathbf{D} \beta$$

for some $\lambda > 0$. The reason why we choose the power of λ to be 6 can be found in chapter 3 of Ruppert et al. [2003]. It can be solved:

$$\begin{aligned} \frac{\partial}{\partial \beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda^6 \beta^T \mathbf{D} \beta &= -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) + 2\lambda^6 \mathbf{D}\beta \\ &= -2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\beta + 2\lambda^6 \mathbf{D}\beta \end{aligned}$$

Set it to zero and solve β, we get:

$$\beta = (\mathbf{X}^T\mathbf{X} + \lambda^6 \mathbf{D})^{-1} \mathbf{X}^T \mathbf{y}$$

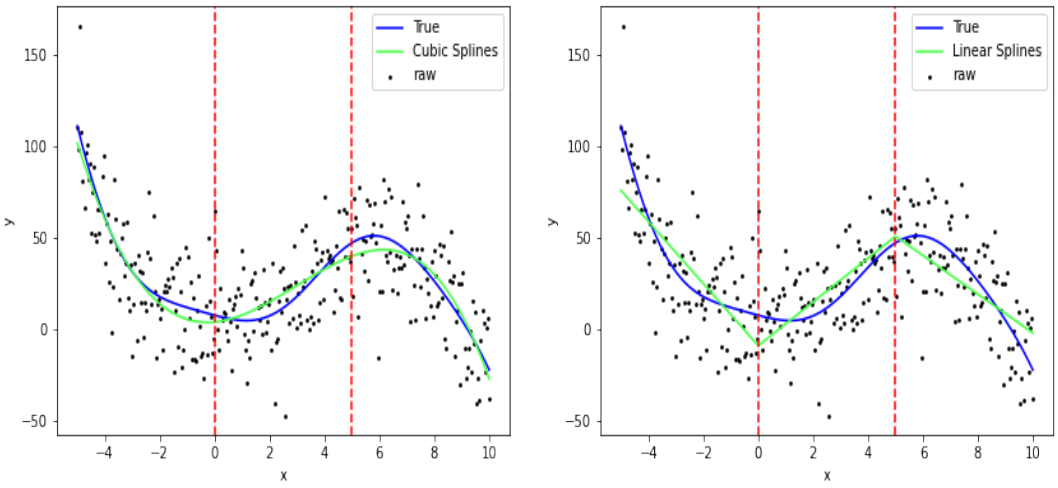


Figure 19: Spline regression of simulated data with fixed knots $\xi_1 = 0$, $\xi_2 = 5$ and sample size $n = 300$ but different choices of degrees. Cubic splines (left), linear splines (right). The plots indicate cubic splines fit a smooth function over the data, but linear splines give the linear regression in each area. Both might be useful with different requirements and backgrounds.

We see that the splines can certainly provide a more flexible model with knots without ignoring global patterns, but it also causes new problems. It would be difficult to interpret the coefficients of the splines due to the presence of knots. Just as the choice of bandwidths is essential for the Nadaraya-Watson estimator and local polynomial estimator, the choice of knots can be challenging for splines. This might be resolved by consulting the experts to gain some certain domain knowledge or using data-driven methods like cross-validation.

3.3 Performance

In the previous sections, we illustrate many essentials and further details about some vital and practical non-parametric regression models, such as Nadaraya-Watson, local polynomials, and splines. Moreover, based on some data, we try to draw and compare the effects of our regression with the true regressions in different step sizes h. However, we just describe the observations and comparisons from these regression models in some qualitative ways.
Indeed, sometimes we can get the better regression model by visualizing since some well-performed regression models hardly have differences with the true regression and we can also ignore some models that are overfitting or underfitting obviously with some extreme step sizes. But, when we cannot distinguish which models are the best only based on our eyes, then it is necessary to use and develop some quantitative methods or estimators to assess the performance of our regression models previously.
To measure the performance of non-parametric regression models, many indexes and tests can be applied, such as mean absolute error (MAE) and robustness to outliers. Therefore, we tend to highlight two aspects of analyzing residuals, goodness-of-fit and R-squared, in assessing the performance and appropriateness of non-parametric regression models.
Recall the general form of non-parametric regression model, the variable Y_i has the relationship with the co-variate X_i:

$$Y_i = f(X_i) + \xi_i,\ i = 1, 2, ...n$$
$$E(\xi_i) = 0,\ Var(\xi_i) = \sigma^2 < \infty$$

where f is the regression curve and the residual ξ_i is the difference between the true value and fitted value.

3.3.1 Goodness-of-fit

Maydeu-Olivares and García-Forero [2010] indicated the goodness-of-fit describes how well the regression model fits the observations data and the necessity of assessing the performance of a model since the fitting regression models with bad performance may bring impacts to further inference. Benedetti [1977] emphasized the key role of using the Priestley–Chao estimator nonparametric regression model for assessing suitability.

Definition 12 (Priestley–Chao estimator)**.** *For the general non-parametric regression model* $Y_i = f(X_i) + \xi_i$*, the Priestley–Chao estimator is* $\hat{f}_n(x) = \sum_{i=1}^{n} Y_i(\frac{x_i - x_{i-1}}{h_n})K(\frac{x - x_i}{h_n})$

Then the variance of random error is

$$\widehat{\sigma^2_{\xi,n}} = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \hat{f}_n(x_i))^2$$

We combine the $\widehat{\sigma^2_{\xi,n}}$ with the Priestley–Chao estimator to get the goodness-of-fit measurement

$$R_n(x) = \frac{\hat{f}_n{}^2(x)}{\widehat{\sigma^2_{\xi,n}} + \hat{f}_n{}^2(x)}$$

Furthermore, we can also choose different appropriate ways to find the goodness of fit of a regression model. On the one hand, comparing some prevalent estimators values, such as the Nadaraya-Watson estimator and Priestley-Chao estimator, and choosing the one with the highest value to provide a better fit towards the model. On the other hand, using some residual-based hypothesis tests, like the chi-squared test, to determine which model is more suitable for our goal.

3.3.2 R-squared

Except for using residuals to assess the fitness between the fitted and true data, R-squared is another useful measurement for performance and it's easier for us to apply in some advanced non-parametric regression models and scenarios. However, it's hard for us to find a common R-squared expression like it is in a parametric model with coefficients, then we formulate it for the non-parametric regression model. See Doksum and Samarov [1995] to know further details and more application examples of using R-squared in various statistical models.

Based on the normal definition of R-squared, we have

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

where SSR is the explained variability by regression, SSE is the unexplained variability by regression (sum of square error) and SST is the total variability of the data.

In the mentioned general form of the non-parametric regression model, we obtain the $f(X) = E(Y|X)$, $Var(\xi|X) = \sigma^2$ and $E(\xi|X) = 0$. Then plug them into the R-squared definition we have:

$$R^2 = \frac{Var(f(X))}{Var(Y)} = 1 - \frac{E(Y - f(X))^2}{Var(Y)}$$

In conclusion, we use the residuals to deliver inferences towards two key components in measuring the performance of the non-parametric regression models. The expressions and applications of R-squared and chi-squared tests are quite similar to them in the parametric regression. Moreover, the Priestley-Chao and Nadaraya-Watson estimators instead of AIC(Akaike information criterion) and some estimators play similar roles in parametric regression models. It is worth emphasizing that the use of them is flexible and depends on our needs.

3.4 Smoothing Parameter Selection

After exploring a few well-known parametric and non-parametric regression models, we would like to see how to estimate the regression curves and functions in some more precious ways. Furthermore, we try to analyze our estimation specifically by using smoothing parameters with different selection methods in this part.

3.4.1 Smoothing Parameter

The smoothing parameter is the controlling parameter that can only affect the smoothness of our fitted curve. For example, we mentioned bandwidth in the histogram part and the width h is the smoothing parameter for this non-parametric statistics method. Moreover, smaller bandwidth triggers undersmoothing with small bias and large variance. Conversely, larger bandwidth leads to oversmoothing with higher bias and lower variance. Therefore, we can use the tradeoff method to balance the bias and variance in a more optimal way.

3.4.2 Selection Methods

There are plenty of smoothing parameter methods, such as Bayesian information criterion, Cross validation, and k nearest neighbors. In this part, we will illustrate two more selection methods for smoothing parameters, leave-one-out cross-validation(LOOCV) and projection estimation methods. Compared with other selection methods, LOOCV is good at estimating and optimizing risk by dropping each observation to construct our model exhaustively. The implementation and theory of minimizing the risk and observation errors is similar to the generalized cross-validation which used smoothing parameters methods indicated in Silverman [1984] and combined with the bias-variance tradeoff method. On the other hand, the projection estimation tends to smooth the parameter and regression by minimizing the (weighted) MISE of the regression function and using the corresponding suitable N to balance the bias and variance.

3.4.2.1 Leave-One-Out Cross Validation

The leave-one-out cross-validation method, a special configuration of k-fold cross-validation, usually be used to evaluate the performance of our machine learning algorithm. It trains the model by leaving one sample from the training set and using the rest of the samples to train the model to predict the observation $\hat{y}_i$ and then repeats this process for each sample.
After training the dataset, the estimation test error is:

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

where $I(y_i \neq \hat{y}_i)$ is the one-zero error, $I(y_i) = \begin{cases} 1, \text{if } y_i \neq \hat{y}_i \\ 0, \text{if } y_i = \hat{y}_i \end{cases}$

Therefore, to show the leave-one-out cross-validation has effects in smoothing the parameter, Wasserman [2010] illustrates the Leave-one-out Cross-Validation score in Chapter 5.3 and the evaluation procedure for prediction error:
The leave-one-out cross-validation (LOOCV) error can be expressed as:

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \hat{Y}_i}{1 - H_{ii}}\right)^2,$$

where H_{ii} is the i-th diagonal entry in the hat matrix $H = X(X^TX)^{-1}X^T$.
We can approximate the LOOCV by representing each H_{ii} as the average leverage, $\bar{H} = \frac{1}{n}\sum_{i=1}^{n} H_{ii}$. Using the Taylor series expansion, we obtain $\frac{1}{(1-\bar{H})^2} \approx 1 + 2\bar{H}$.
Hence, the average residual sum of squares, which is a crucial component in LOOCV, is given by:

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{r}_n(x_i))^2.$$

Therefore, we can approximate the LOOCV as:

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i^{(-i)})^2,$$

where $\hat{r}_{(-i)}(x_i)$ is the estimator, and $\hat{Y}_i^{(-i)}$ is the predicted value when observation (x_i, Y_i) is omitted.
Applying the shortcut that incorporates the average leverage, we find:

$$CV_{(n)} = (1 + 2\bar{H})\hat{\sigma}^2,$$

where $\hat{\sigma}^2$ is the average of the squared residuals.
In addition, compared with other k-fold Cross Validation methods, LOOCV lowers the bias and keeps the estimated error consistent by fitting the whole dataset again and again. Furthermore, the comprehensive testing and estimation of the complex model avoid the underfitting in the regression and performance. Additionally, relevant properties about leave-one-out kernels can be found here A.4

3.4.2.2 Projection Estimation

Another smoothing parameter method is given by the projection estimation of the regression function f. This estimation method tries to approximate the projection of the function f, $\sum_{j=1}^{N}\theta_j\varphi_j$ and changes the smoothness by adjusting the magnitude of the parameter N, where $\{\varphi_j\}$ is the orthonormal basis and $\theta_j = \int_0^1 f(x)\varphi_j(x)dx$.
The general form of the non-parametric regression model, as we mentioned previously:

$$Y_i = f(X_i) + \xi_i, i = 1, 2, ..., n$$

The section 1.7 in Tsybakov [2009] introduced the case of uniform distribution for X_i over $[0, 1]$, then $X_i = \frac{i}{n}$ and based on the definition of θ_j, we can get the well-estimated θ_j is:

$$\frac{1}{n}\sum_{i=1}^{n} f(\frac{i}{n})\varphi_j(\frac{i}{n})$$

Combining with the regression model, we can get the estimator for θ_j is:

$$\hat{\theta}_j = \frac{1}{n}\sum_{i=1}^{n} Y_i \varphi_j(\frac{i}{n})$$

Definition 13 (Projection Estimator)**.** *Let $N \geq 1$ and $N \in Z$, the projection estimator of the function f on $L_2[0,1]$ is :*

$$\hat{f}_{nN}(x) = \sum_{j=1}^{N} \hat{\theta}_j \varphi_j(\frac{i}{n})$$

Note that the order of the projection estimation, N, plays an analogous role with the bandwidth h in the histogram method. In addition, as the small magnitude of N will lead to oversmoothing and the large one will trigger the undersmoothing. Similarly, the principle of applying and finding the most suitable N in smoothing is also based on the tradeoff rule for minimizing and balancing the bias and variance.
Referring to Definition 6 for MISE (mean integrated squared error) and Definition 13 for projection estimator, Tsybakov [2009] also illustrated and added the weighted coefficients, $\lambda_j = I(j \leq N)$, into the estimator to generate the weighted projection estimator and describe its n finite approximation of the regression function f in a general way on $L_2[0,1]$:

$$f_{n,\lambda}(x) = \sum_{j=1}^{n} \lambda_j \varphi_j(x) \frac{1}{n} \sum_{i=1}^{n} Y_i \varphi_j(X_i)$$

And the MISE of the weighted projection estimator $f_{n,\lambda}(x)$ will be:

$$\begin{aligned} WMISE &= E_f ||f_{n,\lambda}(x) - f(x)||_2^2 \\ &= E_f \int_0^1 (f_{n,\lambda}(x) - f(x))^2 dx \end{aligned}$$

Plugging the infinite expansion of the regression function $f = \sum_{j=1}^{\infty} \theta_j \varphi_j$, the weigthed projection estimator, and the estimator $\hat{\theta}_j$ into the WMISE, then we will derive:

$$\begin{aligned} WMISE &= E_f \int_0^1 (\sum_{j=1}^{n} \lambda_j \varphi_j(x) \frac{1}{n} \sum_{i=1}^{n} Y_i \varphi_j(X_i) - \sum_{j=1}^{\infty} \theta_j \varphi_j)^2 dx \\ &= E_f \int_0^1 (\sum_{j=1}^{n} \lambda_j \hat{\theta}_j \varphi_j(x) - \sum_{j=1}^{\infty} \theta_j \varphi_j(x))^2 dx \\ &= E_f (\sum_{j=1}^{n} (\lambda_j \hat{\theta}_j)^2) + \sum_{j=n+1}^{\infty} (\theta_j)^2 \end{aligned}$$

Therefore, by observing this simplification of the weighted mean integrated squared error, the main part (i.e., leading term) of MISE is the expectation and we can adjust the different weighted coefficients to minimize some squared terms of the sum.
Even though both Leave-One-Out Cross Validation and projection estimation are smoothing parameter selection methods, there are some advantages in their principles between them respectively. The LOOCV method tends to show a more comprehensive view of data and fitted values. Except for balancing and reducing the bias and variance to minimize the MSE, LOOCV

can prevent extreme overfitting and underfitting in the machine learning model. Since there is one iteration for leaving each observation, then we can avoid overfitting by testing our data exhaustively to get a better estimation of our model performance rather than using the methods that train and test the data closely and locally. Being different from it, the projection estimation uses the combination of the orthonormal basis and the given data to smooth the regression function with finite N approximation by lowering the mean integrated squared error.

Chapter 4

Conclusion

In-depth research on non-parametric statistical techniques has been done for our project, with a special emphasis on non-parametric regression and Kernel Density Estimation (KDE). We started with the fundamental ideas of KDE and explored performance metrics such as Mean Squared Error and its expansion into integrated forms, emphasising the important work of bandwidth optimisation. We evaluated a range of bandwidth selection strategies, from simple rules to sophisticated plug-in approaches, and we integrated cross-validation methods into our bandwidth determination procedure, resulting in an extensive simulation analysis.
We also looked at how kernel choice affects KDE performance and concluded that, although there are many other types of kernels, in most cases the Gaussian kernel offers a reasonable compromise between computational and accuracy issues. The discussion also included advanced applications that are pertinent to current issues, like multivariate and boundary-corrected KDE.
Transitioning to non-parametric regression, we revisited foundational theories and contrasted these models with traditional parametric regression. Our discussion spanned Nadaraya-Watson and local polynomial regression, along with spline methods, furnishing a diverse toolkit for robust data analysis. This was accompanied by an assessment of performance metrics and a critique on smoothing parameter selection, underscoring the utility of leave-one-out cross-validation. This extensive analysis serves to underline the applicability of non-parametric approaches in empirical research.
Integrating the insights from KDE with non-parametric regression has yielded valuable perspectives on estimating density functions and modeling complex data relationships. Due to the constraints of scope and focus within this report, we regretfully acknowledge that further explorations on certain topics could not be pursued. However, we propose several promising avenues for future research that could continue the work initiated here.

1. Extending KDE applications to high-dimensional datasets to advance our comprehension of complex data structures.
2. Integrating KDE with contemporary machine learning frameworks to refine non-parametric regression approaches, with particular emphasis on large-scale data.
3. Assessing the resilience of KDE and non-parametric regression against datasets characterized by noise, anomalies, or atypical distribution patterns, aiming to develop more robust statistical methods.
4. Implementing the techniques discussed herein across a spectrum of real-world datasets to evaluate their empirical utility and to further refine these methodologies based on the findings.

However, in acknowledging the limitations inherent in any statistical methodology, we recognize that KDE and non-parametric approaches, while powerful, are not universal solutions. They may not adequately address all types of data, particularly when facing datasets with complex

structures or when computational efficiency is a critical factor. We encourage readers to maintain a comprehensive perspective and consider exploring alternative statistical methods that may offer more suitable or efficient solutions for specific types of data or analysis scenarios. It is through the continued pursuit of a diverse array of statistical tools and techniques that the field can advance and adapt to the ever-evolving landscape of data analysis.

Appendix A

Kernel Density Estimation

A.1 Proof of Corollary 2

Corollary 2. *Let f_X be Gaussian with variance σ^2 and let K be the Gaussian kernel. Then the theoretically optimum bandwidth under the asymptotic MISE is*

$$h_{NS} = 1.06 n^{1/5} \sigma$$

Proof. First recall the expression of optimum bandwidth under AMISE in corollary 1

$$h^*_{AMISE} = \left(\frac{R(K)}{n \sigma_K^4 R(f_X'')} \right)^{1/5}$$

We need to find $R(K)$ and $R(f_X'')$. Since we assumed K is the Gaussian kernel we have

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

where we have used the standard result of Gaussian integral $\int_{-\infty}^{\infty} e^{-ax^2} \, dx = \frac{\sqrt{\pi}}{a}$. Next we need to find $R(f_X'')$, which requires to integrate the second derivative of the Gaussian pdf with mean zero and standard deviation σ, which is very involved in terms of calculation. Luckily we can rely on technology:

```
import numpy as np
from sympy import symbols, diff, exp, sqrt, pi, integrate

x, sigma = symbols('x sigma', real=True)

# Gaussian pdf with mean zero
f_X = (1/(sigma*sqrt(2*pi))) * exp(-x**2 / (2*sigma**2))

# second derivative
f_X_2nd_deriv = diff(f_X, x, x)

# its square
squared_f_X_2nd_deriv = f_X_2nd_deriv**2

# integral
R_f_X_2nd_deriv = integrate(squared_f_X_2nd_deriv, (x, -np.inf, np.inf))

# Simplify the result
R_f_X_2nd_deriv.simplify()
```

✓ 0.9s

$$\begin{cases} \frac{3}{8\sqrt{\pi}\sigma^5} & \text{for } 2\left|\arg(\sigma)\right| < \frac{\pi}{2} \\ \frac{\int_{-\infty}^{\infty} \left(\sigma^2 - x^2\right)^2 e^{-\frac{x^2}{\sigma^2}} \, dx}{2\pi\sigma^{10}} & \text{otherwise} \end{cases}$$

where we can ignore the $2|arg(\sigma)| \leq \pi/2$ since $\sigma \in \mathbb{R}$. Then we can substitute $R(K)$ and $R(f_X'')$ in the expression of $h^*_{AMISE} = \left(\frac{R(K)}{n\sigma_K^4 R(f_X'')}\right)^{1/5}$ and realise $\sigma_K^4 = 1$ since the variance of standard normal (Gaussian kernel) is 1 to get:

$$\begin{aligned} h^*_{AMISE} &= \left(\frac{R(K)}{n\sigma_K^4 R(f_X'')}\right)^{1/5}, \\ &= \left(\frac{8\sqrt{\pi}\sigma^5}{2\sqrt{\pi}3n}\right)^{1/5}, \\ &= \frac{4}{3}n^{-1/5}\sigma, \\ &\approx 1.06n^{-1/5}\sigma = h_{NS} \end{aligned}$$

□

A.2 Attaining the bound in Theorem 3

The following result is adapted from Terrell [1990].
Claim: Let $z(x) = \frac{35}{32}(1-x^2)^3$ be a density function with mean zero and variance σ^2 defined on $[-1, 1]$. Let f be any other density functions with the same variance and mean zero. Then z attains the bound in Theorem 3.

Proof. The claim implies that $z(x)$ minimises $R(f'')$, which in turn maximises h_{AMISE} as theorem 3 relies on the result of asymptotic MISE, in particular the optimum bandwidth presented in corollary 1. Hence we need to show that $R(z'') \leq R(f'')$.
Consider the error $e(x) = f(x) - z(x)$. By linearity we have $e''(x) = f''(x) - z''(x)$ and $e''(x)^2 = (f''(x) - z''(x))^2 = f''(x)^2 + z''(x)^2 - 2h''(x)f''(x)$. It follows that

$$R(f'') = R(e'') - R(z'') + 2\int z''(x)f''(x)\,dx.$$

Using $f''(x) = e''(x) + z''(x)$ we have

$$\begin{aligned} R(f'') &= R(e'') - R(z'') + 2\int z''(x)e''(x)\,dx + 2R(z''), \\ &= R(z'') + R(e'') + 2\int z''(x)e''(x)\,dx. \end{aligned}$$

Since $R(e'')$ is non-negative, it suffices to show that $\int z''(x)e''(x)\,dx \geq 0$. We proceed with the calculation. First note that $z''(x) = \frac{105}{16}(-1 + 6x^2 - 5x^4)$. It follows that

$$\int_{-\infty}^{\infty} z''(x)e''(x)\,dx = \frac{105}{16}\int_{-\infty}^{\infty}[-1 + 6x^2 - 5x^4]e''(x)\,dx,$$

Applying integration by parts with $u = -1 + 6x^2 - 5x^4$ and $v = e'(x)$ and noting that z is defined for $|x| \leq 1$ (and hence so are its derivatives) we have

$$\begin{aligned}\int_{-\infty}^{\infty} z''(x)e''(x) &= \frac{105}{16}(-1 + 6x^2 - 5x^4)e'(x)\Big|_{-1}^{1} + \frac{105}{4}\int_{-\infty}^{\infty}(5x^3 - 3x)e'(x)\,dx, \\ &= 0 + \frac{105}{4}\int_{-\infty}^{\infty}(5x^3 - 3x)e'(x)\,dx,\end{aligned}$$

Applying integration by parts with $u = 5x^3 - 3x$ and $v = e(x)$ we have

$$\begin{aligned}\int_{-\infty}^{\infty} z''(x)e''(x) &= \frac{105}{4}(5x^3 - 3x)e(x)\Big|_{-1}^{1} + \frac{315}{4}\int_{-\infty}^{\infty}(1 - 5x^2)e(x)\,dx, \\ &= \frac{105}{2}(e(1) + e(-1)) + \frac{315}{4}\int_{-\infty}^{\infty}(1 - 5x^2)e(x)\,dx \\ &= \frac{105}{2}(e(1) + e(-1)) + \frac{315}{4}\int_{-1}^{1}(1 - 5x^2)e(x)\,dx \\ &+ \frac{315}{4}\int_{|x|>1}(5x^2 - 1)e(x)\,dx.\end{aligned}$$

Now since $e(x) = f(x) - z(x)$, we have $z(x) = 0$ for $|x| > 1$, $z(1) = \frac{35}{32}(1 - 1^2)^3 = 0$ by definition and $f(x)$ is also non-negative since it's a density. Hence it follows that $e(x) \geq 0$ for $|x| \geq 1$, so the first term is positive. The second term is zero because it can be written as $\frac{315}{4}(\int e(x)\,dx - 5\int x^2 e(x)\,dx)$, and since $e(x)$ is the difference between two densities with mean zero and equal variance, the two integrals are also zero i.e., $\int e(x)\,dx = \int x^2 e(x)\,dx = 0$. The third term is non-negative because the integrand is *non-negative* for $|x| > 1$ by definition. Hence it follows that $R(z'') \leq R(f'')$, which implies that z attains the.

□

A.3 Proof of Lemma 3

Lemma 3. *Let f be a function such that f and its derivatives up to order $2s$ are continuous and integrable on $\mathbb{R}$. Further assume $f^{(i)}(x) \to 0$ as $x \to \pm\infty$ for $i = 0, 1, 2, \ldots, 2s$, where $f^{(0)} = f$. Then we have*

$$R(f^{(s)}) := \int (f^{(s)}(x))^2\,dx = (-1)^s \int f^{(2s)}(x)f(x)dx,$$

Proof. We prove by induction on s.
Base case ($s = 1$): For the base case, we need to show that

$$R(f') = -\int_{-\infty}^{\infty} f''(x)f(x)\,dx.$$

This can be shown directly by integration by parts, taking $u = f'(x)$ and $dv = f'(x)dx$, yielding $du = f''(x)dx$ and $v = f(x)$, and noting that the boundary terms (where $x \to \pm\infty$) are zero by assumption.
Inductive step: Assume the statement holds for some integer $s \geq 1$, that is,

$$R(f^{(s)}) = (-1)^s \int_{-\infty}^{\infty} f^{(2s)}(x)f(x)\,dx.$$

We need to show that the statement holds for $s+1$. By definition,

$$R(f^{(s+1)}) = \int_{-\infty}^{\infty} (f^{(s+1)}(x))^2 \, dx.$$

Applying integration by parts, with $u = f^{(s+1)}(x)$ and $dv = f^{(s+1)}(x)dx$, we have $du = f^{(s+2)}(x)dx$ and $v = f^{(s)}(x)$. Since the boundary terms vanish,

$$R(f^{(s+1)}) = -\int_{-\infty}^{\infty} f^{(s)}(x) f^{(s+2)}(x) \, dx.$$

Using the induction hypothesis, we replace $f^{(s)}(x)$ with $(-1)^s f(x)$ and $f^{(2s)}(x)$, yielding

$$R(f^{(s+1)}) = -(-1)^s \int_{-\infty}^{\infty} f^{(2s)}(x) f^{(2)}(x) \, dx.$$

Since $f^{(2s)}(x)f^{(2)}(x) = f^{(2(s+1))}(x)$, we have

$$R(f^{(s+1)}) = (-1)^{s+1} \int_{-\infty}^{\infty} f^{(2(s+1))}(x) f(x) \, dx,$$

which completes the inductive step. Hence, by the principle of induction, the result holds for all integers $s \geq 1$. □

A.4 Leave-One-Out Kernel

To determine the kernel function between two points $\mathbf{x_i}$ and $\mathbf{x_j}$, considering the corresponding leave-one-out density estimates $\hat{p}^{(i)}$ and $\hat{p}^{(j)}$, which exclude $\mathbf{x_i}$ and $\mathbf{x_j}$ respectively. The LOO process is key in deriving Mercer Kernel .
Let $\mathcal{X}$ encompass all possible input values, such that $\mathcal{X} \in \mathbb{R}^d$. Furthermore,

$$\mathcal{P} = \{p \mid p(x) > 0, \forall \mathbf{x} \in \mathcal{X}; \int p(x) \, dx = 1\}.$$

Now, the Hellinger inner product for two probability distributions is introduced as:

$$\langle p, q \rangle = \int \left(\sqrt{p(\mathbf{x})} - \sqrt{p_0(\mathbf{x})}\right) \left(\sqrt{q(\mathbf{x})} - \sqrt{p_0(\mathbf{x})}\right) d\mathbf{x},$$

where $\mathcal{P}$ is the set of probability distributions over $\mathcal{X}$ and p_0 is an arbitrary element in $\mathcal{P}$.
Define the training sample set as $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathcal{X}^n$, with a corresponding estimator modeled as a function $\mathcal{X}^n \to \mathcal{P}$.
Consider $\hat{p} \in \mathcal{P}$ as the density estimate from the training set, and $\hat{p}^{(k)}$ as the density estimate with the k-th sample omitted. The Leave-One-Out (LOO) kernel, is the inner product of $\hat{p}^{(i)}$ and $\hat{p}^{(j)}$ given p_0 at $\hat{p}$:

$$K_l(\mathbf{x_i}, \mathbf{x_j}) = 4(n-1)^2 \int \left(\sqrt{\hat{p}^{(i)}(\mathbf{x})} - \sqrt{\hat{p}(\mathbf{x})}\right) \left(\sqrt{\hat{p}^{(j)}(\mathbf{x})} - \sqrt{\hat{p}(\mathbf{x})}\right) d\mathbf{x}.$$

Remark 6. *The magnitude of a Leave-One-Out (LOO) kernel correlates with the effect on the estimated distribution $\hat{p}$ when a sample is omitted. The kernel value increases if the LOO distributions $\hat{p}^{(i)}$ and $\hat{p}^{(j)}$ significantly shift in the same direction. If the removal of $\mathbf{x_i}$ or $\mathbf{x_j}$ does not yield any influence, the LOO kernel equates to zero, making the samples orthogonal to each other.*

Bibliography

Jacqueline K. Benedetti. On the nonparametric estimation of regression functions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(2):248–253, 3 1977.

Patrick Billingsley. *Probability and Measure*. John Wiley and Sons, Inc., 3rd edition, 2012.

Ricardo Cao, Antonio Cuevas, and Wensceslao González Manteiga. A comparative study of several smoothing methods in density estimation. *Computational Statistics and Data Analysis*, 17(2):153–176, 1994. ISSN 0167-9473. doi: https://doi.org/10.1016/0167-9473(92)00066-Z.

J. E. Chacón and C. Tenreiro. Data-based choice of the number of pilot stages for plug-in bandwidth selection. *Communications in Statistics - Theory and Methods*, 42(12):2200–2214, 2013. doi: 10.1080/03610926.2011.606486.

Daren B. H. Cline. Admissibile kernel estimators of a multivariate density. *The Annals of Statistics*, 16(4):1421–1427, 1988. ISSN 00905364.

Aurore Delaigle and I. Gijbels. Bootstrap bandwidth selection in kernel density estimation from a contaminated sample. *Annals of the Institute of Statistical Mathematics*, 56:19–47, 02 2004. doi: 10.1007/BF02530523.

Kjell Doksum and Alexander Samarov. Nonparametric estimation of global functionals and a measure of the explanatory power of covariates in regression. *The Annals of Statistics*, 23 (5):1443–1473, 10 1995.

V. A. Epanechnikov. Non-parametric estimation of a multivariate probability density. *Theory of Probability and Its Applications*, 14(1):153–158, 1969. doi: 10.1137/1114019.

Ronald A. Fisher. *Statistical Methods for Research Workers*. Oliver and Boyd, 1925.

Li Gan and Qinghua Zhang. The thick market effect on local unemployment rate fluctuations. *Journal of Econometrics*, 133(1):127–152, 2006. ISSN 0304-4076. doi: https://doi.org/10.1016/j.jeconom.2005.03.011.

Eduardo García-Portugués. Nonparametric estimation, Accessed 2024. Online course notes for Nonparametric Statistics at UC3M.

Carl Friedrich Gauss. *Theoria Motus Corporum Coelestium in Sectionibus Conicis Solem Ambientium*. Perthes et Besser, 1809.

Artur Gramacki, Marek Sawerwain, and Jaroslaw Gramacki. Fpga-based bandwidth selection for kernel density estimation using high level synthesis approach. *Bulletin of the Polish Academy of Sciences Technical Sciences*, 64, 05 2015. doi: 10.1515/bpasts-2016-0091.

Peter Hall and J.S. Marron. Estimation of integrated squared density derivatives. *Statistics and Probability Letters*, 6(2):109–115, 1987. ISSN 0167-7152. doi: https://doi.org/10.1016/0167-7152(87)90083-6.

Peter Hall, Simon J. Sheather, M. C. Jones, and J. S. Marron. On optimal data-based bandwidth selection in kernel density estimation. *Biometrika*, 78(2):263–269, 1991. ISSN 00063444.

Pierre-Simon Laplace. *Théorie Analytique des Probabilités*. Veuve Courcier, 1812.

Qi Li and Jeffrey Scott Racine. *Nonparametric Econometrics*. Princeton University Press, 2011.

Henry B. Mann and Donald R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, 18(1):50–60, 1947.

A. Maydeu-Olivares and C. García-Forero. Goodness-of-fit testing. *International Encyclopedia of Education*, pages 190–196, 2010. doi: https://doi.org/10.1016/B978-0-08-044894-7.01333-6.

David Ruppert, M. P. Wand, and R. J. Carroll. *Semiparametric Regression*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2003.

David Scott and Simon Sheather. Kernel density estimation with binned data. *Communications in Statistics-theory and Methods - COMMUN STATIST-THEOR METHOD*, 14:1353–1359, 01 1985. doi: 10.1080/03610928508828980.

R.J. Serfling. Approximation theorems of mathematical statistics. *Biometrics*, 37(4):869, 1981. doi: 10.2307/2530199.

S. J. Sheather and M. C. Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53 (3):683–690, 1991. ISSN 00359246.

B. W. Silverman. A fast and efficient cross-validation method for smoothing parameter choice in spline regression. *Journal of the American Statistical Association*, 79(387):584–589, 1984. doi: 10.1080/01621459.1984.10478084.

Kunio Takezawa. *Introduction to nonparametric regression / Kunio Takezawa.* Wiley series in probability and statistics. Wiley-Interscience, Hoboken, N.J, 2006. ISBN 0471771457.

George R. Terrell. The maximal smoothing principle in density estimation. *Journal of the American Statistical Association*, 85(410):470–477, 1990. ISSN 01621459.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, New York, 2009. ISBN 978-0-387-79051-0.

A. W. van der Vaart. *Nonparametric Density Estimation*, page 341–357. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1998.

M.P. Wand and M.C. Jones. *Kernel Smoothing*. Chapman and Hall/CRC, 1st edition, 1994. doi: 10.1201/b14876.

Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer Publishing Company, Incorporated, 2010. ISBN 1441923225.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

A. Zambom and R. Dias. A review of kernel density estimation with applications to econometrics. https://arxiv.org/pdf/1212.2812.pdf, 2012.