# New Kernels For Density and Regression Estimation via Randomized Histograms

Ruhi Ruhi

A thesis
in
The Department
of
Mathematics & Statistics

Presented in Partial Fulfillment of the Requirements

for the degree of Master of Science (Mathematics) at
Concordia University
Montreal, Quebec, Canada

August 2018

# Concordia University
# School of Graduate Studies

This is to certify that the thesis prepared

By: Ruhi Ruhi

Entitled: New Kernels For Density and Regression Estimation via Randomized Histograms

and submitted in partial fulfillment of the requirements for the degree of

**Master of Science (Mathematics)**

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

______________________________ Chair
Dr.Cody Hyndman

______________________________ Examiner
Dr.Yogendra P. Chaubey

______________________________ Examiner
Dr.Frédéric Godin

______________________________ Thesis Supervisor
Dr. Arusharka Sen

Approved by ______________________________
Dr. Arusharka Sen, Graduate Program Director

August 28, 2018 ______________________________
Date Dr. André Roy, Dean, Faculty of Arts and Science

# ABSTRACT

New Kernels For Density and Regression Estimation via Randomized Histogram

Ruhi Ruhi

In early 20's, the first person to notice the link between Random Forests (RF) and Kernel Methods, *Leo Breiman* (Breiman, 2000), pointed out that Random Forests grown using independent and identically distributed random variables in the tree construction is equivalent to kernels acting on true distribution. Later, Scornet (Scornet, 2016b) defined Kernel based Random Forest (KeRF) estimates and gave explicit expression for the kernels based on Centered RF and Uniform RF. In this paper, we will study the general expression for the connection function (kernel function) of an RF when splits/cuts are performed according to uniform distribution and also according to any general distribution. We also establish the consistency of KeRF estimates in both cases and their asymptotic normality.

*Keywords:* Random Forest, Kernel Methods, Consistency.

# *Acknowledgements*

I would like to acknowledge my indebtedness and render my warmest thanks to my supervisor, Dr. Arusharka Sen for his inspiration, direction, and the consistent help of my master's studies. His friendly guidance and expert advice have been significant through all phases of the work.

I would also wish to express my gratitude to thesis committee Dr. Yogendra P. Chaubey and Dr. Frédéric Godin for valuable suggestions which have contributed greatly to the improvement of the thesis.

I humbly extend my thanks to faculty members of Department of Mathematics and Statistics, for providing excellent working conditions and graduate students who helped me through my studies at Concordia University.

And finally, I would like to thank my friends and family for their consistent support and inspiration throughout my studies.

# Contents

# 1 Introduction

The Random Forest is one of the most popular and most powerful machine learning algorithms. It is basically a trademark term used for an ensemble learning method that consists of pooling together the estimates from many randomly generated decision trees. A decision tree (for classification, regression etc.) is a sequentially constructed partition of the space of input variable (co-variate) $X$. A tree estimate is the average (for regression) or mode (for classification) of all $Y$-values (output) that fall in the partition-cell containing a given input $x$.

Breiman's work on CART (Classification and Regression Trees), ensemble estimator and Random Forest (RF) helped to bridge gap between statistics and computer science, particularly in the field of Machine Learning. Random Forests are exceptionally viable and progressively utilized measurable machine learning techniques. They give remarkable performances in many applied situations for classification and regression problems as they run productively on extensive databases and also they can deal with huge number of input variable without variable deletion and have ability to deal with small sample sizes and high dimensional feature spaces. The corresponding R package *randomForest* can be freely downloaded on the CRAN website (http://www.cran.r-project.org/).

On the hypothetical side, the account of Random Forests are less indisputable and, in spite of their widespread use, little is known about their mathematical properties. However, recent studies have been done towards narrowing the gap between theory and practice, which includes that of Denil et al.(2013) (Denil, Matheson, & Freitas, 2013), who proved the consistency of a particular online forest, Wager(2014) (Wager, 2014) and Mentch and Hooker(2015) (Mentch & Hooker, 2016), who studied the asymptotic normality and Scornet et al.(2015) (Scornet, 2016a) who proved its consistency under appropriate assumptions.

## 1.1 Framework

Most of the following material is based on the papers *Random Forest guided tour* (Biau & Scornet, 2016) and *Random Forest and Kernel Methods* (Scornet, 2016b) by Erwan Scornet and Gérard Biau.

### 1.1.1 Notations and definitions

As explained by Scornet and Biau (Biau & Scornet, 2016), the general structure of Random Forest is non-parametric regression estimation. Assume a training sample of size $n$, $\mathcal{D} = \{(\mathbf{X}_1, Y_1), ..., (\mathbf{X}_n, Y_n)\}$ of independent random variables distributed as independent prototype pair $(\mathbf{X}, Y)$ where $\mathbb{E}[Y^2] < \infty$. An input random vector $\mathbf{X} \in \mathcal{X} \subset \mathbb{R}^p$ is observed and the goal is to predict the square integrable random response Y $\in \mathbb{R}$ by estimating the regression function,

$$r(\mathbf{x}) = \mathbb{E}[Y \mid \mathbf{X} = \mathbf{x}].$$

The aim to use the data set $\mathcal{D}$ to construct an estimate $r_n : \mathcal{X} \to \mathbb{R}$ of the function $r$. The regression function estimate $r_n$ is (mean squared error) consistent if

$$\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2 \to 0 \text{ as } n \to \infty.$$

A random forest is a collection of $M$ randomized regression trees (trees where target variable can take continuous values). Let $r_n(\mathbf{x}, \Theta_j)$ be the predicted value at point $\mathbf{x}$ for the $j$-th tree in the family, where $\Theta_1, \Theta_2, .., \Theta_M$ are independent random variables, distributed as generic random variable $\Theta$, independent of the sample $\mathcal{D}$ (of sample size $n$). These random variables represent the randomization procedure employed. The trees are combined to form the finite forest estimate

$$r_{M,n}(\mathbf{x}, \Theta_1, .., \Theta_M) = \frac{1}{M}\sum_{j=1}^{M} r_n(\mathbf{x}, \Theta_j). \tag{1.1}$$

By the law of large numbers, the finite forest estimate approaches to infinite forest estimate (defining, $r_{\infty,n}(\mathbf{x}) = lim_{M\to\infty} r_{M,n}(\mathbf{x}, \Theta)$):
$\forall\ \mathbf{x} \in [0,1]^d$, almost surely as M $\to \infty$,

$$r_{\infty,n}(\mathbf{x}) = \mathbb{E}[r_n(\mathbf{x}, \Theta)].$$

There are large variety of forests depending on how trees are grown and how the random variable $\Theta$ influences the tree construction. A basic framework to assess the theoretical properties of forests involves models in which partitions do not depend on the training set $\mathcal{D}$. An example is the *Centred Random Forests*(Biau & Scornet, 2016), for which $\mathcal{X} = [0,1]^d$ and has properties as follows:
(i) Samples are drawn without replacement;
(ii) at each node of each individual tree, a coordinate is uniformly chosen in {1,..., d} and,
(iii) a split is performed at the center of the node along the selected coordinate.
The operations (ii)(iii) are recursively repeated $k$ times, where $k \in \mathbb{N}$ is a parameter. The algorithm stops when a binary tree with $k$ levels is built, so that each tree has exactly $2^k$ leaves in the end. The final estimation at the point $\mathbf{x}$ is obtained by taking the average of $Y_i$ corresponding to the $X_i$ in the cell of $\mathbf{x}$. The parameter

$k$ acts as a smoothing parameter that controls the size of the terminal node.
Another example is *Uniform Forest*, which is similar to Centred Forest except that once a split direction is chosen, the splits are made uniformly on the side of the cell, along the preselected co-ordinate (Arlot & Genuer, 2014).

## 1.2 Basics of Kernel and Kernel based Random Forest(KeRF)

One way to break down the complexity of Random Forests is to express forest estimate as a kernel estimate, i.e., estimate $r_n$ which takes the form

$$r_n(\mathbf{x}) = \frac{\sum_{i=1}^{n} Y_i K_n(\mathbf{X}_i, \mathbf{x})}{\sum_{i=1}^{n} K_n(\mathbf{X}_i, \mathbf{x})},$$

where $\{(\mathbf{X}_i, Y_i); 1 \leq i \leq n\}$ is the training set, $(K_n)_k$ is the $k$-th kernel functions of the sequence of kernels ; $n \in \mathbb{N}$ is parameter to be determined.
Note that the $K_n$ doesn't necessarily belongs to Nadaraya-Watson kernels family (Nadaraya, 1964) (Watson, 1964), which satisfy a translation-invariant homogeneous property of the form $K_h(\mathbf{x}, \mathbf{z}) = \frac{1}{h} K((\mathbf{x} - \mathbf{z})/h)$ for some smoothing parameter $h > 0$. The analysis of kernel is more complex, depending on the type of forest under investigation.

### 1.2.1 Kernel based Random Forests (KeRF)

For all $\mathbf{x} \in [0, 1]^d$, we know (finite) random forest estimates satisfy

$$r_{M,n}(\mathbf{x}, \Theta_1, ..., \Theta_M) = \frac{1}{M} \sum_{j=1}^{M} \left( \sum_{i=1}^{n} \frac{Y_i \mathbb{1}_{X_i \in A_n(\mathbf{x}, \Theta_j)}}{N_n(\mathbf{x}, \Theta_j)} \right) \tag{1.2}$$

where $A_n(\mathbf{x}, \Theta_j)$ is the cell containing $\mathbf{x}$, determined by the random variable $\Theta_j$ and data set $\mathcal{D}$ and,

$$N_n(\mathbf{x}, \Theta_j) = \sum_{i=1}^{n} \mathbb{1}_{X_i \in A_n(\mathbf{x}, \Theta_j)}$$

is the number of data points in $A_n(\mathbf{x}, \Theta_j)$ (Scornet, 2016b).

This equation (1.2) is true in particular for non-adaptive forests (i.e., forests built independently of data) as the quantity of observations in each cell cannot be controlled. For example, given two cases with $N_n^1(\mathbf{x}, \Theta_j) > N_n^2(\mathbf{x}, \Theta_j)$ then by equation (1.2), $r_{M,n}^1(\mathbf{x}, \Theta_1, ..., \Theta_M) < r_{M,n}^2(\mathbf{x}, \Theta_1, ..., \Theta_M)$. Thus cells containing smaller number of data points tends to have greater estimate than those with larger number of data points.

To solve this problem, Kernel based random forest(KeRF) estimates were defined, which takes the form:

$$\widetilde{r}_{M,n}(\mathbf{x}, \Theta_1, ..., \Theta_M) = \frac{\sum_{j=1}^{M} \sum_{i=1}^{n} Y_i \mathbb{1}_{X_i \in A_n(\mathbf{x}, \Theta_j)}}{\sum_{j=1}^{M} N_n(\mathbf{x}, \Theta_j)} \tag{1.3}$$

To study more about KeRF, Scornet (Scornet, 2016b) proved a proposition for another form of KeRF estimates.

*Proposition*: Almost surely, for all $\mathbf{x} \in [0, 1]^d$,

$$\widetilde{r}_{M,n}(\mathbf{x}, \Theta_1, ..., \Theta_M) = \frac{\sum_{i=1}^{n} Y_i K_{M,n}(\mathbf{x}, \mathbf{X}_i)}{\sum_{j=1}^{n} K_{M,n}(\mathbf{x}, \mathbf{X}_j)}, \tag{1.4}$$

where

$$K_{M,n}(\mathbf{x}, \mathbf{z}) = \frac{1}{M} \sum_{j=1}^{M} \mathbb{1}_{z \in A_n(\mathbf{x}, \Theta_j)}. \tag{1.5}$$

$K_{M,n}$ is connection function of the M finite forest.

($K_{M,n}$ is a relative frequency of the set $A_n$ containing $x$.)

**Next**,

Defining infinite KeRF estimates $\widetilde{r}_{\infty,n}$ as

$$\widetilde{r}_{\infty,n} = lim_{M\to\infty}\widetilde{r}_{M,n}(\mathbf{x}, \Theta_1, ..., \Theta_M).$$

A proposition for infinite forest estimate (i.e., when number of trees M tends to infinity) was proved by Scornet (Scornet, 2016b).
Note that the infinite random forests are said to be discrete (or continuous), if its corresponding connection function $K_n$ is piece-wise constant (or continuous).

*Proposition*: Consider an infinite discrete or continuous forest. Then, almost surely, for all $\mathbf{x}$,$\mathbf{z} \in [0,1]^d$,

$$lim_{M\to\infty}K_{M,n}(\mathbf{x}, \mathbf{z}) = K_n(\mathbf{x}, \mathbf{z}),$$

where

$$K_n(\mathbf{x}, \mathbf{z}) = \mathbb{P}_\Theta[\mathbf{z} \in A_n(\mathbf{x}, \Theta)]$$

$K_n$ is called the connection function of infinite random forest. Thus, for all $\mathbf{x}\in [0,1]^d$,

$$\widetilde{r}_{\infty,n}(\mathbf{x}) = \frac{\sum_{i=1}^n Y_i K_n(\mathbf{x}, \mathbf{X}_i)}{\sum_{j=1}^n K_n(\mathbf{x}, \mathbf{X}_j)}, \tag{1.6}$$

Thus, infinite KeRF estimates are kernel estimates with kernel function equal to $K_n$.

The denominator of KeRF estimate can be adjusted to obtain a density estimator as we shall see in Chapter 2 and Chapter 3.

### 1.2.2 Uniform KeRF

Uniform Random Forests were first studied by Biau et al.(Biau, Devroye, & Lugosi, 2008). In Uniform RF, the splits are drawn uniformly on the cell edges with no prior on split location.

Using notation $K_k^{uf}$ to denote the connection function of uniform random forest of level $k$.

Scornet (Scornet, 2016b) obtained an explicit expression $K_k^{uf}$ for connection function for infinite uniform random forest as follows,

*Proposition*: For $k \in \mathbb{N}$ and for all $\mathbf{x} \in [0,1]^d$,

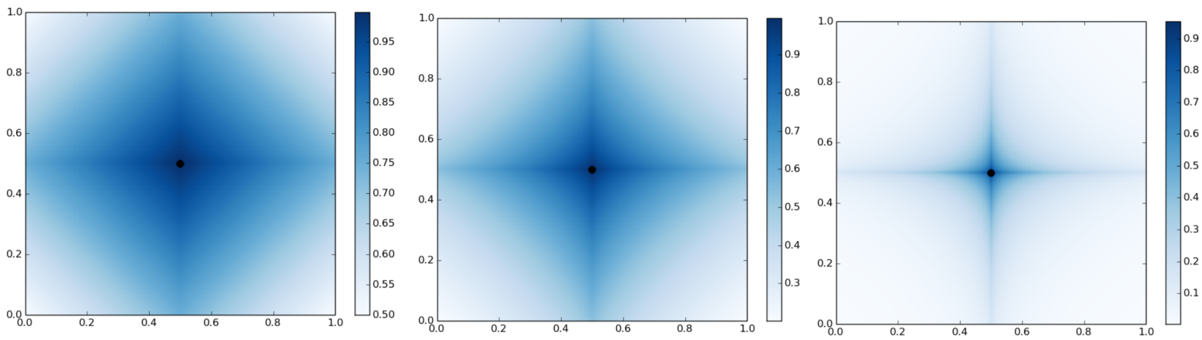
$$K_k^{uf}(\mathbf{0}, \mathbf{x}) = \sum_{k_1,..,k_d;\, \sum_{l=1}^d k_l = k} \frac{k!}{k_1!...k_d!} \left(\frac{1}{d}\right)^k \times \prod_{m=1}^{d} \left(1 - x_m \sum_{j=0}^{k_m - 1} \frac{(-ln\ x_m)^j}{j!}\right) \quad (1.7)$$

with convention $\sum_{j=0}^{-1} \frac{(-ln\ x_m)^j}{j!} = 0$.

The figure below represents the functions $f_1, f_2, \text{and} f_5$ in two dimensions defined by:

$$f_k : [0,1] \times [0,1] \to [0,1]$$
$$\mathbf{z} = (z_1, z_2) \mapsto K_k^{uf}\left(\mathbf{0}, \mid \mathbf{z} - (\frac{1}{2}, \frac{1}{2}) \mid\right),$$

where $\mid \mathbf{z} - \mathbf{x} \mid = (\mid z_1 - x_1 \mid, ..., \mid z_d - x_d \mid)$.



Scornet(Scornet, 2016b) also proved the expression of connection function $K_k^{uf}(\mathbf{x}, \mathbf{z})$ for level $k = 1, 2$ that has the form:

$$K_1^{uf}(x, z) = 1- \mid z - x \mid$$

$$K_2^{uf}(x, z) = 1- \mid z - x \mid + \mid z - x \mid ln\left(\frac{z}{1-x}\right), \tag{1.8}$$

for $x, z \in$ [0,1].

But for levels $k > 2$, the general expression of $K_k^{uf}$ couldn't be derived. So, to overcome this difficulty, he replaced $(\mathbf{x}, \mathbf{z}) \rightarrow K_k^{uf}(\mathbf{0}, \mid \mathbf{z} - \mathbf{x} \mid)$ which is simpler way to build an invariant-by-translation version of uniform kernel $K_k^{uf}$ and the infinite uniform KeRF estimate, denoted by $\widetilde{r}_{\infty,n}^{uf}$ takes the form-

$$\widetilde{r}_{\infty,n}^{uf}(\mathbf{x}) = \frac{\sum_{i=1}^{n} Y_i K_k^{uf}(\mathbf{0}, \mid \mathbf{X}_i - \mathbf{x} \mid)}{\sum_{j=1}^{n} K_k^{uf}(\mathbf{0}, \mid \mathbf{X}_j - \mathbf{x} \mid)}.$$

Later, the theorem below (Scornet, 2016b) was proved to find consistency of infinite uniform KeRF estimates.

**Theorem 1.1.** *Assuming*

$$Y = r(\boldsymbol{X}) + \epsilon$$

*where* $\epsilon$ *is a centred Gaussian noise, independent of* $\boldsymbol{X}$*, with finite variance* $\sigma^2 < \infty$*. Moreover,* $\boldsymbol{X}$ *is uniformly distributed on* $[0, 1]^d$ *and* $r$ *is Lipschitz of order 1. Then, providing* $k \rightarrow \infty$ *and* $n/2^k \rightarrow \infty$*, there exists a constant* $C_1 > 0$ *such that for all* $n > 1$ *and for all* $\boldsymbol{x} \in [0, 1]^d$*,*

$$\mathbb{E}\left[\widetilde{r}_{\infty,n}^{uf}(\boldsymbol{x}) - r(\boldsymbol{x})\right]^2 \leq C_1 n^{-2/(6+3d\, log2)}(logn)^2.$$

#### 1.2.2.1 Drawbacks of Scornet's general expression of connection function $K_k^{uf}$ of Uniform Random Forest

According to Scornet in his paper *Random Forests and Kernel Methods*(Scornet, 2016b), the general expression for connection function $K_k^{uf}(\mathbf{x}, \mathbf{z}); k > 2$ of uniform random forest is difficult to obtain. However, by using the method of *order statistics*, the general expression can be obtained by alternate partitioning scheme, which is shown in Chapter 2 of this thesis.

Also, the expression obtained in equation (1.8) by Scornet (Scornet, 2016b) has a corrected form,

$$K_2^{uf}(x, z) = 1- \mid z - x \mid + \mid z - x \mid ln\big(z(1-x)\big), \quad \text{for } x < z \tag{1.9}$$

which is also proved in Chapter 2.

## 1.3 Outline of thesis

The main objective of this paper is to find the explicit expression of kernel estimate for simplified randomization models, often called *purely random forests*. In this model, the domain of the explanatory variable $x$ is partitioned $m$ times using a random sample of size $n$, independent of the data $\mathcal{D}$, from a distribution supported on the domain of $x$. Note that $m = m_n$ depends on $n$, the size of the data. We consider both uniform as well as non-uniform partitioning distributions. We establish consistency, expansions for bias and variance and asymptotic normality of the resulting estimators.

We denote the expression for the kernel for uniform partitioning as $K_m^U(x, z)$ and for non-uniform partitioning as $K_m^G(x, z)$, where $m$ is the number of splits.

We show in Chapter 2 that the expression for the kernel for uniform partitioning is

for $x, z \in$ [0,1]

$$K_m^U(x,z) = (1- \mid z - x \mid)^m.$$

A similar expression for the kernel can be obtained in the multivariate case; the expression is of the form:

for **x**,**z** $\in [0,1]^d$

$$K_m^U(\mathbf{x},\mathbf{z}) = \prod_{i=1}^{d}(1- \mid z_i - x_i \mid)^m.$$

The proof for multivariate case is straightforward and is not discussed in this thesis.

The rest of the thesis is organized as follows:

Chapter 2 is devoted to the results on kernel estimates for uniform partitioning and their asymptotic behavior. In Chapter 3, kernel estimates for general partitioning are presented along with their asymptotic properties and simulations are presented in Chapter 4.

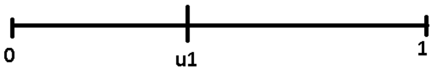
# 2 Uniform Random Forest

In this chapter, we first obtain (Sec. 2.1) the correct expression for the uniform random forest kernel considered by Scornet (Scornet, 2016b) when the partition size is 4. As explained in Chapter 1, this kernel is not very convenient to work with, hence we do not pursue this approach. Instead, we derive the kernel for the alternative partitioning scheme proposed in Chapter 1 (Sec. 1.3: *purely random forest*). The rest of the chapter is devoted to studying properties of the resulting estimators.

## 2.1 Connection function for Scornet's random forest

Consider an interval [0,1] and let $x, z \in$ [0,1].

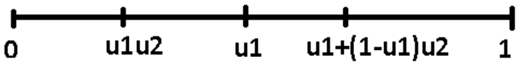
Let a point (say) $u_1$ be drawn uniformly on (0,1), which partitions the interval [0,1] in 2 intervals : (0,$u_1$) and $(u_1, 1)$.




The expression of kernel $K_1^{uf}$, already proved by Scornet (Scornet, 2016b), has form

$$K_1^{uf}(x, z) = 1- \mid z - x \mid .$$

Now, another point is drawn uniformly from one of the two sub-intervals and split is made at that point. This 2nd uniform split will be either in $(0,u_1)$ interval, if $x, z \in (0,\, u_1)$,creating 2 new intervals $(0,u_1u_2)$ and $(u_1u_2, u_1)$ or in the interval $(u_1,$ $1)$, if $x, z \in [u_1,\, 1]$, creating intervals $(u_1, u_1 + (1-u_1)u_2)$ and $(u_1 + (1-u_1)u_2, 1)$.




For any two points $x$ and $z$ in (0,1) (assuming w.l.o.g., x < z ),
Probability that $x$ and $z$ are in same cell after 2 splits is $K_2^{uf}(x,z)$,

$$
\begin{aligned}
K_2^{uf}(x,z) = {} & P(max(x,z) < u_1u_2) + P(min(x,z) > u_1u_2, max(x,z) < u_1) \\
& + P(min(x,z) > u_1, max(x,z) < u_1 + (1-u_1)u_2) \\
& + P(min(x,z) > u_1 + (1-u_1)u_2) \\
= {} & P(z < u_1u_2) + P(x > u_1u_2, z < u_1) + P(x > u_1, z < u_1 + (1-u_1)u_2) \\
& + P(x > u_1 + (1-u_1)u_2) \\
= {} & P(ln(z) < lnu_1 + lnu_2) + P(lnx > lnu_1 + lnu_2, lnz < lnu_1) \\
& + P(ln(1-x) < ln(1-u_1), ln(1-z) > ln(1-u_1) + ln(1-u_2)) \\
& + P(ln(1-x) < ln(1-u_1) + ln(1-u_2))
\end{aligned}
$$

[We know $u \sim unif(0,1) \implies -ln(u) \sim exp(1)$

(putting $-lnu_i = T_i$ and $-ln(1-u_i) = T_i^*; i = 1,2)$]

$$
\begin{aligned}
K_2^{uf}(x,z) = {} & P(T_1 + T_2 < -lnz) + P(T_1 + T_2 > -lnx, T_1 < -lnz) \\
& + P(T_1^* < -ln(1-x), T_1^* + T_2^* > -ln(1-z)) + P(T_1^* + T_2^* < -ln(1-x)) \\
= {} & P(S_2 < -ln(z)) + P(S_2 > -lnx, S_1 < -lnz) \\
& + P(S_1^* < -ln(1-x), S_2^* > -ln(1-z)) + P(S_2^* < -ln(1-x))
\end{aligned}
$$

where $S_i = T_1 + .. + T_i$ and

$S_i^* = T_1^* + .. + T_i^*$

Also, we know that the random process $\{N(t);\ t \geq 0\}$ such that

$$N(t) = max\{n : S_n \leq t\} \text{ where } S_n = T_1 + .. + T_n \ ;\ T_i \sim exp(\lambda)$$

is a Poisson Process with rate $\lambda$.

$$\begin{aligned}
K_2^{uf} &= P(N(-ln(z)) \geq 2) + P(N(-ln(z)) = 1, N(-lnx) - N(-lnz) = 0) \\
&\quad + P(N(-ln(1-x)) = 1, N(-ln(1-z)) - N(-ln(1-x)) = 0) + P(N(-ln(1-x)) \geq 2) \\
&\quad [\text{ where } N(t) \sim Pois(t) \text{ and } N(t_1) - N(t_2) \sim Pois(t_1 - t_2)] \\
&\quad = \left[\sum_{j=2}^{\infty}(-lnz)^j \frac{e^{lnz}}{j!}\right] + \left[\left\{e^{lnz}(-lnz)^1\right\}\left\{e^{lnx-lnz}(-lnx+lnz)^0\right\}\right] \\
&\quad + \left[\left\{(-ln(1-x))^1 e^{ln(1-x)}\right\}\left\{e^{ln(1-z)-ln(1-x)}(-ln(1-z)+ln(1-x))^0\right\}\right] \\
&\quad + \left[\sum_{j=2}^{\infty} e^{ln(1-x)} \frac{(-ln(1-x))^j}{j!}\right] \\
&\quad = \left[z\left(\sum_{j=0}^{\infty}\frac{(-lnz)^j}{j!} - \frac{(-lnz)^0}{0!} - \frac{(-lnz)^1}{1!}\right)\right] + \left[(-zlnz)\left(\frac{x}{z}\right)\right] \\
&\quad + \left[-(1-x).ln(1-x)\left(\frac{1-z}{1-x}\right)\right] \\
&\quad + \left[(1-x)\left(\sum_{j=0}^{\infty}\frac{(-ln(1-x))^j}{j!} - (-ln(1-x))^0 - (-ln(1-x))^1\right)\right]
\end{aligned}$$

$$\begin{aligned}
&= z(e^{-lnz} - 1 + lnz) - xlnz - (1-z).ln(1-x) + (1-x)\big(e^{-ln(1-x)} - 1 + ln(1-x)\big) \\
&= 1 - z + zlnz - xlnz - (1-z)ln(1-x) + x + (1-x)ln(1-x) \\
&= 1 - (z-x) + (z-x)lnz + (z-x)ln(1-x)
\end{aligned}$$

Hence,

$$K_2^{uf}(x,z) = 1 - \mid z - x \mid + \mid z - x \mid ln(z(1-x)) \quad ;\text{ for } x < z$$

This expression for $k = 2$ is the corrected form of the expression proved by *Erwan Scornet* (Scornet, 2016b).

For $k = 3$ or more, the general expression of the connection function $K_k^{uf}(\mathbf{x},\mathbf{z})$ is difficult to obtain. So this method is not convenient for higher splits.

In the next section, we consider the partitioning scheme of a *purely random forest*. Recall that in this scheme a sample of size $n$ of uniformly distributed points is used to create a partition of the interval $[0, 1]$. We denote the resulting kernel by $K_m^U(x, z)$ and corresponding estimate by $r_n(x)$.

## 2.2 Expression for Kernel by method of order statistics

Observing $K_m^U(x, z)$ is the probability that $x$ and $z$ are connected in (infinite) random forest after $m$ splits, the function $K_m^U$ characterizes the shape of the cells in the infinite forest.

**Theorem 2.1.** *Let $m \in \mathbb{N}$ and consider a uniform random forest of level m. Then, for all x, z $\in$ [0,1],*

$$K_m^U(x, z) = (1 - |x - z|)^m.$$

*Proof.* w.l.o.g., assume $x < z$.

Let $m$ points be drawn independently from uniform distribution on (0,1) and let splits be made at those points for partitioning.

Assuming their order statistics to be

$u_{(1)} < u_{(2)} < u_{(3)} < .. < u_{(m)}$.

Let $u_{(j)} < x < z < u_{(j+1)}$ for some $j \in$ {1,2,..,m}

We know that joint pdf of two order statistics, such that $1 \leqslant i < j \leqslant m$ is

$$\begin{aligned} f_{(i)(j)}(x_{(i)}, x_{(j)}) =& \frac{m!}{(i-1)!(j-i-1)!(m-j)!}[F(x_{(i)})]^{i-1} \\ & \times [F(x_{(j)}) - F(x_{(i)})]^{j-i-1}[1-F(x_{(j)})]^{m-j} f(x_{(i)}) f(x_{(j)}), \end{aligned}$$

$-\infty < x_{(i)} < x_{(j)} < \infty$

Now, Considering

$$\begin{aligned} P(x \,\&\, z \in j\text{-th cell}) &= P(u_{(j)} \leq x, z \leq u_{(j+1)}) \\ &= \int_0^x \int_z^1 f_{u_{(j)},u_{(j+1)}}(u_{(j)}, u_{(j+1)}) du_{(j)} du_{(j+1)} \\ &= \int_0^x \int_z^1 \frac{m!}{(j-1)!(j+1-j-1)!(m-j-1)!}[F(u_{(j)})]^{j-1} \\ &\quad \times [F(u_{(j+1)}) - F(u_{(j)})]^{j+1-j-1}[1-F(u_{(j+1)})]^{m-j-1} f(u_{(j)}) f(u_{(j+1)}) \\ &= \int_0^x \int_z^1 \frac{m!}{(j-1)!(m-j-1)!} u_{(j)}^{j-1}(1-u_{(j+1)})^{m-j-1} du_{(j)} du_{(j+1)} \\ &= \frac{m!}{(j-1)!(m-j-1)!} \int_0^x u_{(j)}^{j-1} du_{(j)} \int_z^1 (1-u_{(j+1)})^{m-j-1} du_{(j+1)} \\ &= \frac{m!}{(j-1)!(m-j-1)!} \left( \frac{(u_{(j)})^j}{j}\bigg|_0^x . \frac{(1-u_{(j+1)})^{m-j}}{m-j}\bigg|_1^z \right) \\ &= \frac{m!}{j!(m-j)!} \, x^j \, (1-z)^{m-j} \end{aligned}$$

We know, $K_m^U$(x,z) = P( $x, z \in$ same cell).

$$\begin{aligned} P(x, z \; lies \; in \; same \; cell) &= \sum_{j=1}^{m} \frac{m!}{(j-1)!(m-j)!} \, x^j \, (1-z)^{m-j} \\ &= (x+1-z)^m \\ K_m^U(x,z) &= (1 - |z-x|)^m \end{aligned}$$

□

**Alternative proof**:

w.l.o.g., assume $x < z$.

Let $m$ points be drawn independently from uniform distribution on (0,1) and let splits be made at those points for partitioning. Let split point be denoted by $u$. Then,

$$\begin{aligned} P(x, z \text{ lies in same cell after } m \text{ splits}) &= (P[u \in (0, x) \cup (z, 1)])^m \\ &= (1 - |z - x|)^m \end{aligned}$$

## 2.3 Asymptotic behavior of $r_n(x)$

We know that the estimate takes the form,

$$r_n(x) = \frac{\sum_{i=1}^n Y_i K_m^U(x, X_i)}{\sum_{i=1}^n K_m^U(x, X_i)}.$$

As proved in Theorem 2.1,

$$K_m^U(x, X_i) = (1 - |x - X_i|)^m.$$

So,

$$r_n(x) := \frac{N_n(x)}{D_n(x)} = \frac{\frac{m}{n} \sum_{i=1}^n Y_i (1 - |x - X_i|)^m}{\frac{m}{n} \sum_{i=1}^n (1 - |x - X_i|)^m}. \tag{2.1}$$

where $N_n(x) = \frac{m}{n} \sum\limits_{i=1}^{n} Y_i (1 - |x - X_i|)^m$ and $D_n(x) = \frac{m}{n} \sum\limits_{i=1}^{n} (1 - |x - X_i|)^m$.

As we shall see below , the denominator $D_n(x)$ converges to $2f(x)$. Therefore, the $D_n(x)$ can be modified to make it a density estimator (say) $F_n(x)$ which is $\frac{D_n(x)}{2}$.

### 2.3.1 Consistency and Asymptotic Normality of $r_n(x)$

In this section, we explore large-sample properties of our estimator. To accomplish this objective, we require some assumptions mentioned below:

**Assumptions (A1):**

- $f(x)$ & $r(x)$ are at least twice differentiable.
- $m = m_n \to \infty$ as $n \to \infty$ and

  $\frac{m}{n} \to 0$.

  (In other words, $m \to 0$ but at slower rate than $n^{-1}$)

**Theorem 2.2.** *Assume (A1) holds. For $x \in [0,1]$,*
*Let $N_n(x) = \frac{m}{n}\sum_{i=1}^{n} Y_i(1-|x-X_i|)^m$ be a uniform kernel regression estimator and $D_n(x) = \frac{m}{n}\sum_{i=1}^{n}(1-|x-X_i|)^m$ be a uniform kernel density estimator.*
*Then,the following convergence in probability holds:*

$$r_n(x) := \frac{N_n(x)}{D_n(x)} = \frac{\left(\frac{m}{n}\sum_{i=1}^{n} Y_i(1-|x-X_i|)^m\right)}{\left(\frac{m}{n}\sum_{i=1}^{n}(1-|x-X_i|)^m\right)} \to \frac{r(x)f(x)}{f(x)} = r(x) \tag{2.2}$$

*Proof.* : We will first consider $\mathbb{E}[D_n(x)]$ of equation (2.2),

$$\begin{aligned}
\mathbb{E}[D_n(x)] &= \mathbb{E}\left(\frac{m}{n}\sum_{i=1}^{n}(1-|x-X_i|)^m\right) \\
&= m\mathbb{E}(1-\mid x-X_1\mid)^m \\
&= m\int_0^1 (1-\mid x-u\mid)^m f(u)du \\
&= m\int_{-(1-x)}^{x} (1-\mid s\mid)^m f(x-s)ds \qquad \text{; putting } x-u=s \\
&= m\int_{-(1-x)}^{x} (1-\mid s\mid)^m \left[f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\right]ds \\
&\text{(using Taylor series expansions for } f(x-s)) \\
&= m\int_{-(1-x)}^{0} (1+s)^m \left[f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\right]ds \\
&\quad + m\int_0^x (1-s)^m \left[f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\right]ds
\end{aligned}$$

$$
\begin{aligned}
&= \frac{m}{m+1} f(x)\Big(2 - x^{m+1} - (1-x)^{m+1}\Big) \\
&+ \frac{m}{m+1} f'(x)\Big(x(1-x)^{m+1} - (1-x)x^{m+1}\Big) \\
&+ \frac{m}{(m+1)(m+2)} f'(x)\Big((1-x)^{m+2} - x^{m+2}\Big) \\
&- \frac{m}{m+1}\frac{f''(x)}{2}.\Big((1-x)^2 x^{m+1} + x^2(1-x)^{m+1}\Big) \\
&- \frac{m}{(m+1)(m+2)} f''(x)\Big(x^{m+2}(1-x) + x(1-x)^{m+2}\Big) \\
&+ \frac{m}{(m+1)(m+2)(m+3)} f''(x)\Big(2 - x^{m+3} - (1-x)^{m+3}\Big).
\end{aligned} \tag{2.3}
$$

Taking $m \to \infty$ , as $0 < x < 1 \quad \& \quad 0 < (1-x) < 1$

$\implies x^m \to 0$ and $(1-x)^m \to 0$

Hence,

$$\mathbb{E}[D_n(x)] \to 2f(x)$$

Next,

Taking $\mathbb{E}[N_n(x)]$ of equation (2.2),

$$
\begin{aligned}
\mathbb{E}[N_n(x)] &= \mathbb{E}\Bigg(\frac{m}{n}\sum_{i=1}^{n} Y_i(1 - |x - X_i|)^m\Bigg) \\
&= m\mathbb{E}(Y_1(1- \mid x - X_1 \mid)^m) \\
&= m \int\int v(1- \mid x - u \mid)^m f(u,v)dudv \\
&= m \int_0^1 (1- \mid x - u \mid)^m \Big(\underbrace{\int v f(v \mid u)dv}_{=r(u)}\Big) f(u)du \quad ; \text{putting } x - u = s \\
&= m \int_{-(1-x)}^{x} (1- \mid s \mid)^m f(x-s) r(x-s) ds \\
&= m \int_{-(1-x)}^{0} (1+s)^m f(x-s)r(x-s)ds + m\int_0^x (1-s)^m f(x-s)r(x-s)ds \\
&= m \int_{-(1-x)}^{0} (1+s)^m \Big[f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\Big] \\
&\qquad \Big[r(x) - sr'(x) + \frac{s^2}{2}r''(x) + \mathcal{O}(s^3)\Big]ds \\
&+m\int_0^x (1-s)^m \Big[f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\Big]\Big[r(x) - sr'(x) + \frac{s^2}{2}r''(x) + \mathcal{O}(s^3)\Big]ds
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{m}{m+1} f(x)r(x)\Big(2 - x^{m+1} - (1-x)^{m+1}\Big) \\
&+ \frac{m}{m+1}(f(x)r(x))'\Bigg(x(1-x)^{m+1} - (1-x)x^{m+1}\Bigg) \\
&+ \frac{m}{(m+1)(m+2)}(f(x)r(x))'\Bigg((1-x)^{m+2} - x^{m+2}\Bigg) \\
&- \frac{m}{2(m+1)}(f(x)r(x))''\Bigg(x^2(1-x)^{m+1} + (1-x)^2x^{m+1}\Bigg) \\
&- \frac{m}{(m+1)(m+2)}(r(x)f(x))''\Bigg(x(1-x)^{m+2} - (1-x)x^{m+2}\Bigg) \\
&+ \frac{m}{(m+1)(m+2)(m+3)}(f(x)r(x))''\Bigg(2 - x^{m+3} - (1-x)^{m+3}\Bigg)
\end{aligned} \tag{2.4}
$$

Taking $m \to \infty$,

as $0 < x < 1$ & $0 < (1-x) < 1$

$\implies x^m \to 0$ and $(1-x)^m \to 0$

$$\mathbb{E}[N_n(x)] \to 2f(x)r(x)$$

Hence,

$$\frac{\mathbb{E}[N_n(x)]}{\mathbb{E}[D_n(x)]} \to \frac{2r(x)f(x)}{2f(x)} = r(x)$$

**Further,** we will show the consistency of our estimate.

For that we will prove $Var[N_n(x)] \to 0$ and $Var[D_n(x)] \to 0$.

$$
\begin{aligned}
Var[D_n(x)] &= Var\Big(\frac{m}{n}\sum_{i=1}^{n}(1 - |x - X_i|)^m\Big) \\
&= \frac{m^2}{n} Var(1- \mid x - X_1 \mid)^m \\
&= \frac{m}{n}\Big[m\mathbb{E}(1- \mid x - X_1 \mid)^{2m} - \frac{1}{m}(m\mathbb{E}(1- \mid x - X_1 \mid)^m)^2\Big] \\
&= \frac{m}{n}\Big[m\int_0^1 (1- \mid x - u \mid)^{2m} f(u)du - \frac{1}{m}(m\mathbb{E}(1- \mid x - X_1 \mid)^m)^2\Big] \\
&= \frac{m}{n}\Big[m\int_{-(1-x)}^{x} (1- \mid s \mid)^{2m} f(x-s)ds - \frac{1}{m}(m\mathbb{E}(1- \mid x - X_1 \mid)^m)^2\Big]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{m}{n}\Big[m\int_{-(1-x)}^{0}(1+s)^{2m}\Big(f(x)-sf'(x)+\frac{s^2}{2}f''(x)+\mathcal{O}(s^3)\Big)ds \\
&\quad + m\int_{0}^{x}(1-s)^{2m}\Big(f(x)-sf'(x)+\frac{s^2}{2}f''(x)+\mathcal{O}(s^3)\Big)ds \\
&\quad - \frac{1}{m}(m\mathbb{E}(1-\mid x-X_1\mid)^m)^2\Big] \\
&= \frac{m}{n}\Bigg[\frac{m}{2m+1}f(x)(2-x^{2m+1}-(1-x)^{2m+1}) \\
&\quad + \frac{m}{2m+1}f'(x)(x(1-x)^{2m+1}-(1-x)x^{2m+1}) \\
&\quad + \frac{m}{(2m+1)(2m+2)}f'(x)((1-x)^{2m+2}-x^{2m+2}) \\
&\quad - \frac{m}{2(2m+1)}f''(x)(x^2(1-x)^{2m+1}+(1-x)^2x^{2m+1}) \\
&\quad - \frac{m}{(2m+1)(2m+2)}f''(x)((1-x)x^{2m+2}+x(1-x)^{2m+2}) \\
&\quad + \frac{m}{(2m+1)(2m+2)(2m+3)}f''(x)(2-x^{2m+3}-(1-x)^{2m+3}) \\
&- \frac{1}{m}(m\mathbb{E}(1-\mid x-X_1\mid)^m)^2\Bigg]
\end{aligned}
$$

From assumption (A1),

$m \to \infty$ as $n \to \infty$ and $\frac{m}{n} \to 0$

as $0 < x < 1$ & $0 < (1-x) < 1$

$\implies x^m \to 0$ and $(1-x)^m \to 0$

Thus,

$$Var[D_n(x)] \to 0$$

Now,

$$
\begin{aligned}
Var[N_n(x)] &= Var\Big(\frac{m}{n}\sum_{i=1}^{n}Y_i(1-|x-X_i|)^m\Big) \\
&= \frac{m^2}{n}V\Big(Y_1(1-\mid x-X_1\mid)^m\Big) \\
&= \frac{m}{n}\Big[m\mathbb{E}(Y_1^2(1-\mid x-X_1\mid)^{2m}) - \frac{1}{m}\{m\mathbb{E}(Y_1(1-\mid x-X_1\mid)^m)\}^2\Big]
\end{aligned}
$$

$$= \frac{m}{n}\left[ m \int\int v^2 (1- \mid x-u \mid)^{2m} f(u,v) du dv - \underbrace{\frac{1}{m}\{m\mathbb{E}(Y(1- \mid x - X_1 \mid)^m)\}^2}_{A} \right]$$

$$= \frac{m}{n}\left[ m \int\int v^2 (1- \mid x-u \mid)^{2m} f(v|u) f(u) du dv - A \right]$$

$$= \frac{m}{n}\left[ m \int_{-(1-x)}^{x} (1- \mid s \mid)^{2m} f(x-s) \Bigg( \underbrace{\int v^2 f(v|x-s) dv}_{=\zeta(x-s)} \Bigg) ds - A \right]$$

$$\left[ \text{define: } \int y^2 f(y|x) dy = \zeta(x) \right]$$

$$= \frac{m}{n}\left[ m \int_{-(1-x)}^{x} (1- \mid s \mid)^{2m} f(x-s) \zeta(x-s) ds - A \right]$$

$$= \frac{m}{n}\Bigg[ m \int_{-(1-x)}^{x} (1- \mid s \mid)^{2m} \Bigg( f(x) - s f'(x) + \frac{s^2}{2} f''(x) + \mathcal{O}(s^3) \Bigg)$$
$$\times \Bigg( \zeta(x) - s\zeta'(x) + \frac{s^2}{2} \zeta''(x) + \mathcal{O}(s^3) \Bigg) ds - A \Bigg]$$

$$= \frac{m}{n}\Bigg[ m \int_{-(1-x)}^{0} (1+s)^{2m} \Bigg( f(x) - s f'(x) + \frac{s^2}{2} f''(x) + \mathcal{O}(s^3) \Bigg)$$
$$\times \Bigg( \zeta(x) - s\zeta'(x) + \frac{s^2}{2} \zeta''(x) + \mathcal{O}(s^3) \Bigg) ds$$
$$+ m \int_{0}^{x} (1-s)^{m} \Bigg( f(x) - s f'(x) + \frac{s^2}{2} f''(x) + \mathcal{O}(s^3) \Bigg)$$
$$\times \Bigg( \zeta(x) - s\zeta'(x) + \frac{s^2}{2} \zeta''(x) + \mathcal{O}(s^3) \Bigg) ds - A \Bigg]$$

$$= \frac{m}{n}\Bigg[ \frac{m}{2m+1} (f(x)\zeta(x))(2 - x^{2m+1} - (1-x)^{2m+1})$$
$$+ \frac{m}{2m+1} (f(x)\zeta(x))'(x(1-x)^{2m+1} - (1-x)x^{2m+1})$$
$$+ \frac{m}{(2m+1)(2m+2)} (f(x)\zeta(x))'((1-x)^{2m+2} - x^{2m+2})$$
$$+ \frac{m}{4m+2} (f(x)\zeta(x))''((1-x)^2 x^{2m+1} - x^2(1-x)^{2m+1})$$
$$- \frac{m}{(2m+1)(2m+2)} (f(x)\zeta(x))''((1-x)x^{2m+2} + x(1-x)^{2m+2})$$
$$+ \frac{m}{(2m+1)(2m+2)(2m+3)} (f(x)\zeta(x))''(2 - x^{2m+3} - (1-x)^{2m+3}) - A \Bigg]$$

From assumption (A1),

$m \to \infty$ as $n \to \infty$ and $\frac{m}{n} \to 0$

as $0 < x < 1$ & $0 < (1 - x) < 1$

$\implies x^m \to 0$ and $(1 - x)^m \to 0$

Thus,

$$Var[N_n(x)] \to 0$$

which implies that $N_n(x) - \mathbb{E}[N_n(x)] \to 0$ in probability and

$D_n(x) - \mathbb{E}[D_n(x)] \to 0$ in probability.

Thus,

$$r_n(x) \to r(x)$$

in probability hence, is consistent. $\square$

**Next**, we study the asymptotic normality of our regression estimator $N_n(x)$ and density estimator $D_n(x)$.

**Theorem 2.3.** *Let $m \in \mathbb{N}$ be number of splits and $n$ be the sample size. Assuming (A1) holds, we have*

$$\sqrt{\frac{n}{m}} \begin{pmatrix} N_n(x) - \mathbb{E}(N_n(x)) \\ D_n(x) - \mathbb{E}(D_n(x)) \end{pmatrix} \longrightarrow \mathcal{N}_2 \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} f(x)\zeta(x) & f(x)r(x) \\ f(x)r(x) & f(x) \end{pmatrix} \right)$$

*where $\zeta(x) = \mathbb{E}[Y^2 \mid X = x]$.*

*Proof.* Consider

$\sqrt{\frac{n}{m}}\Big(\mathbb{E}[N_n(x)] - 2f(x)r(x)\Big)$ and $\sqrt{\frac{n}{m}}\Big(\mathbb{E}[D_n(x)] - 2f(x)\Big)$

Recall equation (2.4)

$$
\begin{aligned}
&\sqrt{\frac{n}{m}}\Big(\mathbb{E}[N_n(x)]-2f(x)r(x)\Big)\\
&=\sqrt{\frac{n}{m}}\Bigg[\frac{m}{m+1}f(x)r(x)\Big(2-x^{m+1}-(1-x)^{m+1}\Big)\\
&+\frac{m}{m+1}(f(x)r(x))'\Big(x(1-x)^{m+1}-(1-x)x^{m+1}\Big)\\
&+\frac{m}{(m+1)(m+2)}(f(x)r(x))'\Big((1-x)^{m+2}-x^{m+2}\Big)\\
&-\frac{m}{2(m+1)}(f(x)r(x))''\Big(x^2(1-x)^{m+1}+(1-x)^2x^{m+1}\Big)\\
&-\frac{m}{(m+1)(m+2)}(r(x)f(x))''\Big(x(1-x)^{m+2}-(1-x)x^{m+2}\Big)\\
&+\frac{m}{(m+1)(m+2)(m+3)}(f(x)r(x))''\Big(2-x^{m+3}-(1-x)^{m+3}\Big)-2f(x)r(x)\Bigg]
\end{aligned}
$$

Taking $m\to\infty$,

as $0<x<1$ & $0<(1-x)<1$

$\implies x^m\to 0$ and $(1-x)^m\to 0$

This implies,

$$\sqrt{\frac{n}{m}}\Big(\mathbb{E}[N_n(x)]-2f(x)r(x)\Big)\to 0$$

Similarly, Recall equation (2.3)

$$
\begin{aligned}
\sqrt{\frac{n}{m}}\Big(\mathbb{E}[D_n(x)]-2f(x)\Big)&=\sqrt{\frac{n}{m}}\Bigg[\frac{m}{m+1}f(x)\Big(2-x^{m+1}-(1-x)^{m+1}\Big)\\
&+\frac{m}{m+1}f'(x)\Big(x(1-x)^{m+1}-(1-x)x^{m+1}\Big)\\
&+\frac{m}{(m+1)(m+2)}f'(x)\Big((1-x)^{m+2}-x^{m+2}\Big)\\
&-\frac{m}{m+1}\frac{f''(x)}{2}.\Big((1-x)^2x^{m+1}+x^2(1-x)^{m+1}\Big)\\
&-\frac{m}{(m+1)(m+2)}f''(x)\Big(x^{m+2}(1-x)+x(1-x)^{m+2}\Big)\\
&+\frac{m}{(m+1)(m+2)(m+3)}f''(x)\Big(2-x^{m+3}-(1-x)^{m+3}\Big)-2f(x)\Bigg]
\end{aligned}
$$

Taking $m \to \infty$,

as $0 < x < 1$ & $0 < (1-x) < 1$

$\implies x^m \to 0$ and $(1-x)^m \to 0$

This implies,

$$\sqrt{\frac{n}{m}}\bigg(\mathbb{E}[D_n(x)] - 2f(x)\bigg) \to 0$$

Consider,

$$\begin{aligned}\sqrt{\frac{n}{m}}\bigg(r_n(x) - r(x)\bigg) &= \sqrt{\frac{n}{m}}\bigg(\frac{N_n(x)}{D_n(x)} - r(x)\bigg)\\ &= \sqrt{\frac{n}{m}}\Bigg[\frac{N_n(x) - \mathbb{E}(N_n(x))}{D_n(x)} - \frac{\mathbb{E}(N_n(x))}{\mathbb{E}(D_n(x))}\frac{[D_n(x) - \mathbb{E}(D_n(x))]}{D_n(x)} + \frac{\mathbb{E}(N_n(x))}{\mathbb{E}(D_n(x))} - r(x)\Bigg]\end{aligned} \tag{2.5}$$

Next, we define

$$\sigma_{11}(x) := \frac{n}{m}Var[N_n(x)] \to f(x)\zeta(x)$$

$$\sigma_{22}(x) := \frac{n}{m}Var[D_n(x)] \to f(x)$$

$$\sigma_{12}(x) = \sigma_{21}(x) := \frac{n}{m}Cov[N_n(x), D_n(x)]$$

$$Cov[N_n(x), D_n(x)] = Cov\Bigg(\frac{m}{n}\sum_{i=1}^{n} Y_i(1-|x-X_i|)^m, \frac{m}{n}\sum_{j=1}^{n}(1-\mid x - X_j \mid)^m\Bigg)$$

$$= \frac{m^2}{n^2}\sum_i \sum_j cov(Y_i(1-|x-X_i|)^m, (1-\mid x - X_j \mid)^m)$$

$$= \frac{m^2}{n^2}\sum_{i=j=1}^{n} cov(Y_i(1-|x-X_i|)^m, (1-|x-X_i|)^m)$$

$$+ \frac{m^2}{n^2}\underbrace{\sum_{i\neq j} cov(Y_i(1-|x-X_i|)^m, (1-\mid x - X_j \mid)^m)}_{=0} \ (as\ X_i's \text{ are i.i.d})$$

$$= \frac{m^2}{n}\Bigg[\mathbb{E}(Y_1(1-\mid x - X_1 \mid)^{2m}) - \mathbb{E}(Y_1(1-\mid x - X_1 \mid)^m)\mathbb{E}(1-\mid x - X_1 \mid)^m\Bigg]$$

$$= \frac{m^2}{n}\Big[\int\int v(1- \mid x-u \mid)^{2m} f(u,v)dudv - \mathbb{E}(Y_1(1- \mid x-X_1 \mid)^m)\mathbb{E}(1- \mid x-X_1 \mid)^m\Big]$$

$$= \frac{m^2}{n}\Big[\int (1- \mid x-u \mid)^{2m}\underbrace{\Big(\int v f(v \mid u)dv\Big)}_{=r(u)} f(u)du - \mathbb{E}(Y_1(1- \mid x-X_1 \mid)^m)\mathbb{E}(1- \mid x-X_1 \mid)^m\Big]$$

$$= \frac{m^2}{n}\Big[\int_0^1 (1- \mid x-u \mid)^{2m} f(u)r(u)du - \mathbb{E}(Y_1(1- \mid x-X_1 \mid)^m)\mathbb{E}(1- \mid x-X_1 \mid)^m\Big]$$

$$= \frac{m^2}{n}\Big[\int_{-(1-x)}^{x} (1- \mid s \mid)^{2m} f(x-s)r(x-s)ds - \mathbb{E}(Y_1(1- \mid x-X_1 \mid)^m)\mathbb{E}(1- \mid x-X_1 \mid)^m\Big]$$

(putting $x-u=s$)

$$= \frac{m^2}{n}\Big[\int_{-(1-x)}^{0} (1+s)^{2m}\Big(f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\Big)$$

$$\times \Big(r(x) - sr'(x) + \frac{s^2}{2}r''(x) + \mathcal{O}(s^3)\Big)ds - \int_0^x (1-s)^{2m}\Big(f(x) - sf'(x) + \frac{s^2}{2}f''(x) + \mathcal{O}(s^3)\Big)$$

$$\times \Big(r(x) - sr'(x) + \frac{s^2}{2}r''(x) + \mathcal{O}(s^3)\Big)ds - \mathbb{E}(Y_1(1- \mid x-X_1 \mid)^m)\mathbb{E}(1- \mid x-X_1 \mid)^m\Big]$$

$$= \frac{m}{n}\Big[\frac{m}{2m+1}f(x)r(x)(2 - x^{2m+1} - (1-x)^{2m+1})$$

$$+ \frac{m}{2m+1}(f(x)r(x))'[x(1-x)^{2m+1} - x^{2m+1}(1-x)]$$

$$+ \frac{m}{(2m+1)(2m+2)}(f(x)r(x))'[(1-x)^{2m+2} - x^{2m+2}]$$

$$- \frac{m}{2(2m+1)}(f(x)r(x))''((1-x)^2x^{2m+1} + x^2(1-x)^{2m+1}]$$

$$- \frac{m}{(2m+1)(2m+2)}(f(x)r(x))''[x(1-x)^{2m+2} + (1-x)x^{2m+2}]$$

$$+ \frac{m}{(2m+1)(2m+2)(2m+3)}(f(x)r(x))''[2 - x^{2m+3} - (1-x)^{2m+3}]$$

$$- \frac{1}{m}(m\mathbb{E}(Y_1(1- \mid x-X_1 \mid)^m))(m\mathbb{E}(1- \mid x-X_1 \mid)^m)\Big]$$

Letting $m \to \infty$

$$\implies \sigma_{12} = \frac{n}{m}Cov[N_n(x), D_n(x)] \to f(x)r(x)$$

Thus, by *Cramér-Wold device* the joint convergence stated in theorem 2.3 follows.

Let $\mathcal{Z}_1 \sim \mathcal{N}(0, \sigma_{11})$, the limiting distribution of regression estimate $N_n(x)$ and

$\mathcal{Z}_2 \sim \mathcal{N}(0, \sigma_{22})$, the limiting distribution of density estimate $D_n(x)$.

Then, by Central Limit Theorem,

$$\sqrt{\frac{n}{m}}\Big(N_n(x) - \mathbb{E}(N_n(x))\Big) \xrightarrow{d} \mathcal{Z}_1$$
$$\sqrt{\frac{n}{m}}\Big(D_n(x) - \mathbb{E}(D_n(x))\Big) \xrightarrow{d} \mathcal{Z}_2$$

Thus, applying Slutsky's Theorem to equation (2.5),

$$\sqrt{\frac{n}{m}}\Big(r_n(x) - r(x)\Big) \xrightarrow{d} \left(\frac{\mathcal{Z}_1}{2f(x)} - \frac{r(x)}{2f(x)}\mathcal{Z}_2\right)$$

since $D_n(x) \to 2f(x)$ , $N_n(x) \to 2f(x)r(x)$ and $\frac{\mathbb{E}(N_n(x))}{\mathbb{E}(D_n(x))} \to r(x)$.

Thus,

$$\begin{aligned}
\sqrt{\frac{n}{m}}\Big(r_n(x) - r(x)\Big) &\sim \mathcal{N}\left(0, \frac{\sigma_{11}}{4f^2(x)} - 2\frac{r(x)}{4f^2(x)}\sigma_{12} + \frac{r^2(x)}{4f^2(x)}\sigma_{22}\right) \\
&\sim \mathcal{N}\left(0, \frac{f(x)\zeta(x)}{4f^2(x)} - 2\frac{r^2(x)f(x)}{4f^2(x)} + \frac{r^2(x)f(x)}{4f^2(x)}\right) \\
&\sim \mathcal{N}\left(0, \frac{1}{4f(x)}(\zeta(x) - r^2(x))\right) \\
&= \mathcal{N}\left(0, \frac{Var[Y \mid X = x]}{4f(x)}\right)
\end{aligned}$$

and

$$\begin{aligned}
\sqrt{\frac{n}{m}}\begin{pmatrix} N_n(x) - \mathbb{E}(N_n(x)) \\ D_n(x) - \mathbb{E}(D_n(x)) \end{pmatrix} &\longrightarrow \mathcal{N}_2\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{11}(x) & \sigma_{12}(x) \\ \sigma_{21}(x) & \sigma_{22}(x) \end{pmatrix}\right) \\
&\longrightarrow \mathcal{N}_2\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} f(x)\zeta(x) & f(x)r(x) \\ f(x)r(x) & f(x) \end{pmatrix}\right)
\end{aligned}$$

$\square$

# 3 Kernel for General partitioning distribution

Recall that in Chapter 2, $m$ points are drawn uniformly on (0,1) and splits are made at those points. In this chapter, $m$ points are drawn from a general distribution $G(\cdot)$ and splits are performed at those points. We will study the expression for the connection function and show that the corresponding estimate $r_n(x)$ is consistent for $r(x)$. We denote the resulting kernel by $K_m^G(x,z)$.

## 3.1 Expression for the connection function $K_m^G(x,z)$

We know that $K_m^G(x,z)$ is the probability that $x$ and $z$ are connected in random forests after $m$ splits.

**Theorem 3.1.** *Let $m \in \mathbb{N}$. Then, for any two points $x$ and $z$, the connection function of a finite forest of level m takes the form:*

$$K_m^G(x,z) = (1- \mid G(z) - G(x) \mid)^m$$

*where $G(\cdot)$ is the distribution function of split points.*

*Proof.* Let $m$ points be chosen from any distribution with c.d.f. $G(x)$ and splits $(y_i; i = 1, 2, ., m)$ are made at these $m$ points.

Take order statistics of splits to be

$y_{(0)} \leqslant y_{(1)} \leqslant y_{(2)} .. \leqslant y_{(m)}$

where $y_{(i)} \sim$ general distribution function $G$ with density $g$.

w.l.o.g., Assuming $x < z$.

Let $x, z \in [y_{(i)}, y_{(i+1)}]$ for some $i$,

$$
\begin{aligned}
P(x, z \in i^{th}\ cell) &= P(y_{(i)} \leqslant x < z \leqslant y_{(i+1)}) = P(y_i \leqslant x,\ z \leqslant y_{(i+1)}) \\
&= \int_{-\infty}^{x} \int_{z}^{\infty} g_{Y_{(i)}, Y_{(i+1)}}(y_{(i)}, y_{(i+1)}) dy_{(i)} dy_{(i+1)} \\
&= \int_{-\infty}^{x} \int_{z}^{\infty} \frac{m!}{(i-1)!(i+1-i-1)!(m-i-1)!} [G(y_{(i)})]^{i-1} \\
&\times [G(y_{(i+1)}) - G(y_{(i)})]^{i+1-i-1} [1 - G(y_{(i+1)})]^{m-i-1} . g(y_{(i)}) . g(y_{(i+1)}) dy_{(i)} dy_{(i+1)} \\
&= \frac{m!}{(i-1)!(m-i-1)!} \underbrace{\int_{-\infty}^{x} (G(y_{(i)})^{i-1} g(y_{(i)}) dy_{(i)}}_{= \left[ \frac{(G(y_{(i)}))^i}{i} \right]_{-\infty}^{x}} \underbrace{\int_{z}^{\infty} (1 - G(y_{(i+1)}))^{m-i-1} g(y_{(i+1)}) dy_{(i+1)}}_{= \left[ \frac{(1 - G(y_{(i+1)}))^{m-i}}{m-i} \right]_{\infty}^{z}} \\
&= \frac{m!}{i!(m-i)!} . (G(x))^i (1 - G(z))^{m-i}
\end{aligned}
$$

$$
\begin{aligned}
\text{P}(x, z \text{ lies in same cell after } m \text{ splits}) &= \sum_{i=0}^{m} \frac{m!}{i!(m-i)!} . (G(x))^i (1 - G(z))^{m-i} \\
&= (1 - G(z) + G(x))^m \\
K_m^G(x, z) &= (1 - |G(z) - G(x)|)^m
\end{aligned}
$$

□

**Alternative proof**:

w.l.o.g., assume $x < z$.

Let $m$ points be chosen from any distribution with c.d.f. $G(x)$ and splits are made at these $m$ points. Denoting split point by $y$.

Then,

$$\begin{aligned} P(x, z \text{ lies in same cell after } m \text{ splits}) &= (P[y \in (-\infty, x) \cup (z, \infty)])^m \\ &= (1 - |G(z) - G(x)|)^m. \end{aligned}$$

## 3.2 Asymptotic behavior of $r_n(x)$

In the above section, we find that the connection function has the form:

$$K_m^G(x, z) = (1 - |G(x) - G(z)|)^m.$$

Therefore, the estimate becomes:

$$r_n(x) := \frac{N_n(x)}{D_n(x)} = \frac{\frac{m}{n}\sum_{i=1}^n Y_i(1 - |G(x) - G(X_i)|)^m}{\frac{m}{n}\sum_{i=1}^n (1 - |G(x) - G(X_i)|)^m}. \tag{3.1}$$

where $N_n(x) = \frac{m}{n}\sum\limits_{i=1}^n Y_i(1 - |G(x) - G(X_i)|)^m$ and
$D_n(x) = \frac{m}{n}\sum\limits_{i=1}^n (1 - |G(x) - G(X_i)|)^m$.
As we shall see below, the denominator $D_n(x)$ converges to $2\frac{f(x)}{g(x)}$. Therefore, the $D_n(x)$ can be modified to make it a density estimator (say) $F_n(x)$ which is $\frac{1}{2}g(x)D_n(x)$.

### 3.2.1 Consistency and Asymptotic Normality of $r_n(x)$

In this section, we investigate large-sample properties of our estimator. For this, we need the assumptions **(A1)**, mentioned in the Chapter 2.

**Theorem 3.2.** *Assume (A1) holds and $m \in \mathbb{N}$ is number of splits and $n$ be sample size.*

*Let* $N_n(x) = \frac{m}{n}\sum_{i=1}^{n} Y_i(1-|G(x)-G(X_i)|)^m$, *the kernel regression estimator and* $D_n(x) = \frac{m}{n}\sum_{i=1}^{n}(1-|G(x)-G(X_i)|)^m$, *the kernel density estimator where* $G(\cdot)$ *is the distribution function of split points with density* $g$.

*Then, the following convergence in probability holds:*

$$r_n(x) := \frac{N_n(x)}{D_n(x)} = \frac{\left(\frac{m}{n}\sum_{i=1}^{n} Y_i(1-|G(x)-G(X_i)|)^m\right)}{\left(\frac{m}{n}\sum_{i=1}^{n}(1-|G(x)-G(X_i)|)^m\right)} \to \frac{\frac{r(x)f(x)}{g(x)}}{\frac{f(x)}{g(x)}} = r(x) \tag{3.2}$$

*Proof.* We will first consider $\mathbb{E}[D_n(x)]$ of equation (3.2),

$$\begin{aligned}\mathbb{E}[D_n(x)] &= \mathbb{E}\left(\frac{m}{n}\sum_{i=1}^{n}(1-|G(x)-G(X_i)|)^m\right)\\ &= m\mathbb{E}(1-|G(x)-G(u)|)^m\end{aligned}$$

$$\begin{aligned}&= m\int_{-\infty}^{\infty}(1-|G(x)-G(u)|)^m f(u)du\\ &= m\int_{-(1-G(x))}^{G(x)}(1-|s|)^m \frac{f(G^{-1}(G(x)-s))}{g(G^{-1}(G(x)-s))}ds\end{aligned}$$

Putting $s = G(x) - G(u) \implies u = G^{-1}(G(x)-s)$

$du = \frac{-1}{g(G^{-1}(G(x)-s))}ds$

Let $ms = t \implies mds = dt$

$$= \int_{-m(1-G(x))}^{mG(x)}\left(1-\frac{|t|}{m}\right)^m \frac{f(G^{-1}(G(x)-\frac{t}{m}))}{g(G^{-1}(G(x)-\frac{t}{m}))}dt$$

Letting $m \to \infty$

$$\mathbb{E}[D_n(x)] \longrightarrow \int_{-\infty}^{\infty} e^{-|t|}\,\frac{f(x)}{g(x)}dt = \frac{2f(x)}{g(x)}$$

Hence,

$$\mathbb{E}[D_n(x)] \longrightarrow \frac{2f(x)}{g(x)} \tag{3.3}$$

Next,

Taking $\mathbb{E}[N_n(x)]$ of equation 3.2,

$$\begin{aligned}
\mathbb{E}[N_n(x)] &= \mathbb{E}\left(\frac{m}{n}\sum_{i=1}^{n} Y_i(1-|G(x)-G(X_i)|)^m\right) \\
&= m\mathbb{E}(Y_1(1-|G(x)-G(X_1)|)^m) \\
&= m\int\int v(1-|G(x)-G(u)|)^m f(u,v)dudv \\
&= m\int_{-\infty}^{\infty}(1-|G(x)-G(u)\ |)^m \underbrace{\left(\int vf(v\mid u)dv\right)}_{=r(u)} f(u)du \\
&= m\int_{-\infty}^{\infty}(1-|G(x)-G(u)|)^m f(u)r(u)du
\end{aligned}$$

Putting $G(x)-G(u)=s \Rightarrow u = G^{-1}(G(x)-s)$

$$\Rightarrow du = \frac{-1}{g(G^{-1}(G(x)-s))}ds$$

$$\begin{aligned}
&= m\int_{-(1-G(x))}^{G(x)}(1-\mid s\mid)^m \frac{f(G^{-1}(G(x)-\frac{t}{m}))r(G^{-1}(G(x)-\frac{t}{m}))}{g(G^{-1}(G(x)-\frac{t}{m}))}ds \\
&= \int_{-m(1-G(x)}^{mG(x)}\left(1-\frac{|t|}{m}\right)^m \frac{f(G^{-1}(G(x)-\frac{t}{m}))r(G^{-1}(G(x)-\frac{t}{m}))}{g(G^{-1}(G(x)-\frac{t}{m}))}dt
\end{aligned}$$

[Letting $ms=t$]

Taking $m\to\infty$

$$\mathbb{E}[N_n(x)] \longrightarrow \int_{-\infty}^{\infty} e^{-|t|}\frac{f(x)r(x)}{g(x)}dt = \frac{2f(x)r(x)}{g(x)}$$

Hence,

$$\frac{\mathbb{E}[N_n(x)]}{\mathbb{E}[D_n(x)]} \to \frac{\frac{2f(x)r(x)}{g(x)}}{\frac{2f(x)}{g(x)}} = r(x).$$

**Further,** we will show the consistency of our estimate.

For that we will consider $Var[N_n(x)] \to 0$ and $Var[D_n(x)] \to 0$.

$$Var[D_n(x)] = Var\left(\frac{m}{n}\sum_{i=1}^{n}(1-|G(x)-G(X_i)|)^m\right)$$
$$= \frac{m^2}{n}Var(1-|G(x)-G(u)|)^m$$
$$= \frac{m^2}{n}\left[\mathbb{E}(1-|G(x)-G(u)|)^{2m} - (\mathbb{E}(1-|G(x)-G(u)|)^m)^2\right]$$
$$= \frac{m}{n}\left[m\int_{-\infty}^{\infty}(1-|G(x)-G(u)\ |)^{2m}f(u)du - \frac{1}{m}\left(m\mathbb{E}(1-|G(x)-G(u)|)^m\right)^2\right]$$
$$\left[\text{Putting } G(x)-G(u)=s \Rightarrow u = G^{-1}(G(x)-s)\ \Rightarrow du = \frac{-1}{g(G^{-1}(G(x)-s))}ds\right]$$
$$= \frac{m}{n}\left[m\int_{-(1-G(x))}^{G(x)}(1-|s|)^{2m}\frac{f(G^{-1}(G(x)-s))}{g(G^{-1}(G(x)-s))}ds - \frac{1}{m}\left(m\mathbb{E}(1-|G(x)-G(X_1)|)^m\right)^2\right]$$
$$= \frac{m}{n}\Big[\underbrace{\int_{-m(1-G(x)}^{mG(x)}\left(1-\frac{|t|}{m}\right)^{2m}\frac{f(G^{-1}(G(x)-\frac{t}{m}))}{g(G^{-1}(G(x)-\frac{t}{m}))}dt}_{=A} - \frac{1}{m}\left(m\mathbb{E}(1-|G(x)-G(X_1)|)^m\right)^2\Big]$$

$\big($putting $ms = t\big)$

Let $m \to \infty$, then,

$$A \to \int_{-\infty}^{\infty} e^{-2|t|}\ \frac{f(x)}{g(x)}dt = \frac{f(x)}{g(x)}$$

From assumption (A1),

m $\to \infty$ as $n \to \infty$ and $\frac{m}{n} \to 0$.

Hence,

$$Var[D_n(x)] \to 0$$

$$
\begin{aligned}
Var[N_n(x)] &= Var\left(\frac{m}{n}\sum_{i=1}^{n} Y_i(1-|G(x)-G(X_i\ |)^m\right) \\
&= \frac{m^2}{n}\left[\mathbb{E}(Y_1^2(1-|G(x)-G(X_1)|)^{2m}) - \left(\mathbb{E}(Y_1(1-|G(x)-G(X_1)|)^m)\right)^2\right] \\
&= \frac{m}{n}\left[m\int\int v^2(1-|G(x)-G(u)\ |)^{2m}f(u,v)dudv\right. \\
&\left. -\frac{1}{m}\left(m\mathbb{E}(Y_1(1-|G(x)-G(X_1)|)^m)\right)^2\right] \\
&= \frac{m}{n}\left[m\int_{-\infty}^{\infty}(1-|G(x)-G(X_1)|)^{2m}\underbrace{\left(\int v^2 f(v|u)dv\right)}_{=\zeta(u)}f(u)du\right. \\
&\left. -\frac{1}{m}\left(m\mathbb{E}(Y_1(1-|G(x)-G(X_1)|)^m)\right)^2\right] \\
&= \frac{m}{n}\left[m\int_{-(1-G(x))}^{G(x)}(1-\mid s\mid)^{2m}\frac{f(G^{-1}(G(x)-s))\zeta(G^{-1}(G(x)-s))}{g(G^{-1}(G(x)-s))}ds\right. \\
&\left. -\frac{1}{m}\left(m\mathbb{E}(Y_1(1-|G(x)-G(X_1)|)^m)\right)^2\right]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{m}{n}\left[\underbrace{\int_{-m(1-G(x))}^{G(x)}\left(1-\frac{|t|}{m}\right)^{2m}\frac{f(G^{-1}(G(x)-\frac{t}{m}))\zeta(G^{-1}(G(x)-\frac{t}{m}))}{g(G^{-1}(G(x)-\frac{t}{m}))}dt}_{A}\right. \\
&\left. -\frac{1}{m}\left(m\mathbb{E}(Y_1(1-|G(x)-G(X_1)|)^m)\right)^2\right]
\end{aligned}
$$

Let $m \to \infty$, then,

$$A \to \int_{-\infty}^{\infty} e^{-2|t|}\,\frac{f(x)\zeta(x)}{g(x)}dt = \frac{f(x)\zeta(x)}{g(x)}$$

From assumption (A1),

$m \to \infty$ as $n \to \infty$ and $\frac{m}{n} \to 0$

Hence,

$$Var[N_n(x)] \to 0$$

which implies that $N_n(x) - \mathbb{E}[N_n(x)] \to 0$ in probability and

$D_n(x) - \mathbb{E}[D_n(x)] \to 0$ in probability.

Thus,

$$r_n(x) \to r(x)$$

in probability hence, is consistent.

$\square$

In this paper, we do not investigate the above result for non-absolute continuous distribution $G(\cdot)$.

Now, we state the asymptotic normality of our density and regression estimator.

**Theorem 3.3.** *Let $m \in \mathbb{N}$ be number of splits and $n$ be sample size. Assuming (A1) holds. Consider any distribution function $G(\cdot)$ with density function $g(\cdot)$, then*

$$\sqrt{\frac{n}{m}}\begin{pmatrix} N_n(x) - \mathbb{E}(N_n(x)) \\ D_n(x) - \mathbb{E}(D_n(x)) \end{pmatrix} \longrightarrow \mathcal{N}_2\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \frac{f(x)\zeta(x)}{g(x)} & \frac{f(x)r(x)}{g(x)} \\ \frac{f(x)r(x)}{g(x)} & \frac{f(x)}{g(x)} \end{pmatrix}\right)$$

*where $\zeta(x) = \mathbb{E}[Y^2|X = x]$.*

Now, similar to Chapter 2 it follows that

$$\sqrt{\frac{n}{m}}\big(r_n(x) - r(x)\big) \to \mathcal{N}\left(0, \frac{Var[Y|X = x]g(x)}{4f(x)}\right).$$

## 3.3 Finding the order of the bias

Considering $Var[D_n(x)]$ from Section 3.2.1, i.e.,

$$Var[D_n(x)] = Var\left(\frac{m}{n}\sum_{i=1}^{n}(1 - |G(x) - G(X_i)|)^m\right)$$
$$= \frac{m}{n}\Big[\int_{-m(1-G(x))}^{mG(x)}\left(1 - \frac{|t|}{m}\right)^{2m}\frac{f(G^{-1}(G(x) - \frac{t}{m}))}{g(G^{-1}(G(x) - \frac{t}{m}))}dt - \frac{1}{m}\left(m\mathbb{E}(1 - |G(x) - G(X_1)|)^m\right)^2\Big]$$

Consider the integral

$$I = \int_{-m(1-G(x))}^{mG(x)} \left(1 - \frac{|t|}{m}\right)^{2m} \frac{f(G^{-1}(G(x) - \frac{t}{m}))}{g(G^{-1}(G(x) - \frac{t}{m}))} dt. \tag{3.4}$$

As the bias only depends on the function $f$, to find the order of the bias, we consider

$$\begin{aligned}
& f\left(G^{-1}\left(G(x) - \frac{t}{m}\right)\right) - f(G^{-1}(G(x))) \qquad (\text{ Let } f(G^{-1}(x)) = h(x)) \\
=& h\left(G(x) - \frac{t}{m}\right) - h(G(x)) \\
=& \left[h(G(x)) - \frac{t}{m}h'(G(x)) + \frac{1}{2}\frac{t^2}{m^2}h''(G(x)) - \cdots\right] - h(G(x)) \\
=& \frac{-t}{m}h'(G(x)) + \frac{1}{2}\frac{t^2}{m^2}h''(G(x)) - \cdots
\end{aligned}$$

Now, from equation (3.4), the integral will approach

$$I \to \int_{-m(1-G(x))}^{mG(x)} \frac{\left(1 - \frac{|t|}{m}\right)^{2m} \left(\frac{-t}{m}h'(G(x)) + \frac{1}{2}\frac{t^2}{m^2}h''(G(x)) + \cdots\right)}{g\left(G^{-1}(G(x) - \frac{t}{m})\right)} dt$$

and hence, the optimum order of $m$ will be (by equating leading terms of variance and squared bias)

$$\frac{m}{n} = \frac{1}{m^2} \quad \implies n = m^3 \text{ or } m = n^{1/3}$$

**Special case**:

If $G(x) = \frac{1}{2}$ (i.e., symmetric $G$ centred at $x$)

then the integral (3.4) will become:

$$I = \int_{-m/2}^{m/2} \frac{\left(1 - \frac{|t|}{m}\right)^{2m} \left( - \frac{t}{m}h'(\frac{1}{2}) + \frac{1}{2}\frac{t^2}{m^2}h''(\frac{1}{2}) + ..\right)}{g\left(G^{-1}(\frac{1}{2} - \frac{t}{m})\right)} dt$$

Therefore, optimum order will be:

$$\frac{m}{n} = \frac{1}{m^4} \implies n = m^5 \quad \text{or } m = n^{1/5}$$

So, the optimal choices are :

$$\begin{cases} \frac{m}{n} = \frac{1}{m^4} \implies m^5 = n & \text{; when } G(x) = 1/2 \\ \frac{m}{n} = \frac{1}{m^2} \implies m^3 = n & \text{; otherwise} \end{cases}$$

The kernel regression and density estimator, for $G(x) = 1/2$, thus becomes

$$N_n(x) = \frac{m}{n}\sum_{i=1}^{n} Y_i(1 - |G(x) - G(X_i)|)^m = \frac{m}{n}\sum_{i=1}^{n} Y_i(1 - |1/2 - G(x - X_i)|)^m$$

$$D_n(x) = \frac{m}{n}\sum_{i=1}^{n}(1 - |G(x) - G(X_i)|)^m = \frac{m}{n}\sum_{i=1}^{n}(1 - |1/2 - G(x - X_i)|)^m$$

# 4 Simulation Experiments

In order to illustrate mathematical results from previous sections, we perform some simulations with R. We try to compare the true densities and estimated densities for uniform and non-uniform case for different values of $m$ (number of splits).

## 4.1 Density Estimation

- **Uniform Kernel Density Estimation**

  For this simulation, we consider the estimator:

$$D_n(x) = A \cdot \frac{m}{n} \sum_{i=1}^{n} (1 - |x - X_i|)^m$$

  where $A = \frac{1}{2}$ is the multiplicative factor to make the density estimator consistent.

  To estimate a uniform kernel, we assume $X \sim Beta(3, 10)$ to be the true distribution with density function,

$$f(x) = \frac{x^2(1-x)^9}{B(3,10)}; 0 \leqslant x \leqslant 1$$

  where $B(3,10) = \frac{\Gamma(3)\Gamma(10)}{\Gamma(13)}$.

  We show the estimate for different values of $m$ (number of splits) in figures below:

$m$= 20 (fig.4.1) , 32 (fig.4.2), 39 (fig.4.3)

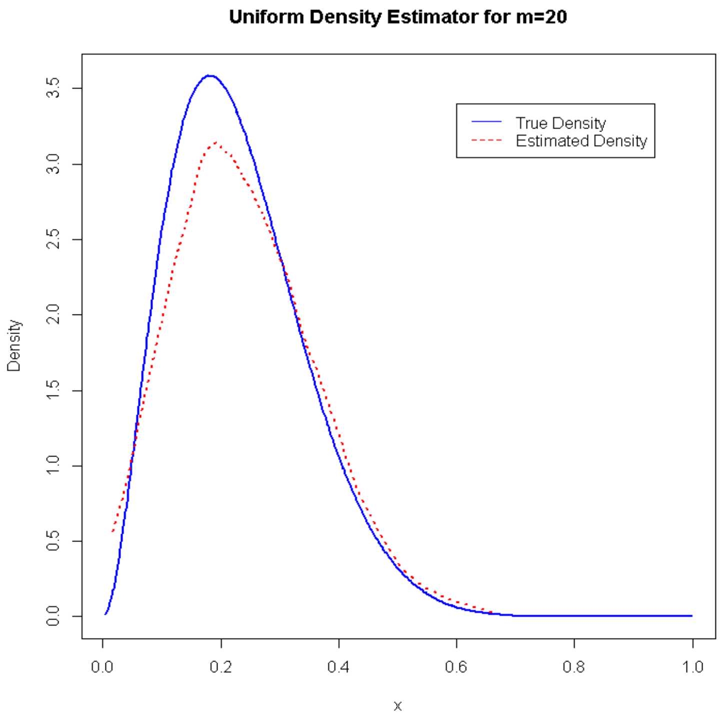



Figure 4.1: Uniform Kernel Density Estimate for $m = 20$

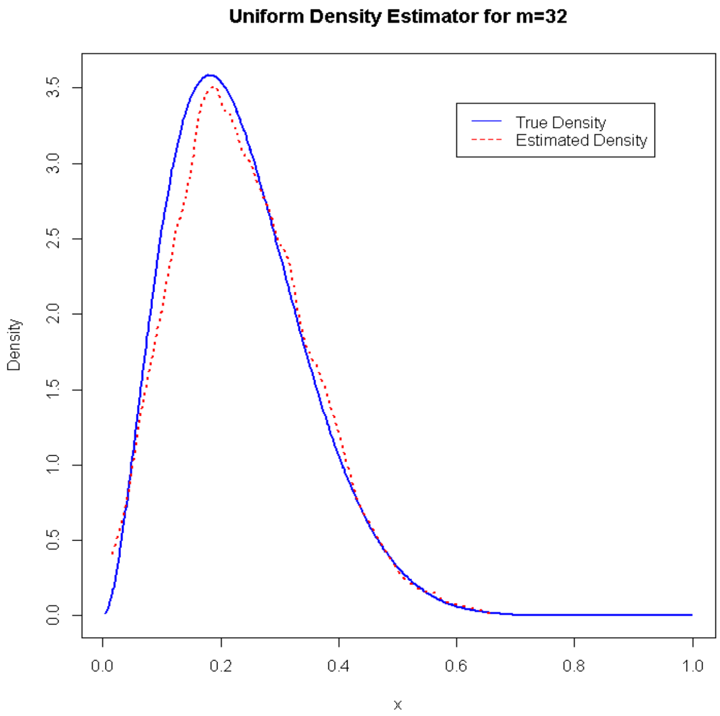



Figure 4.2: Uniform Kernel Density Estimate for $m = 32$

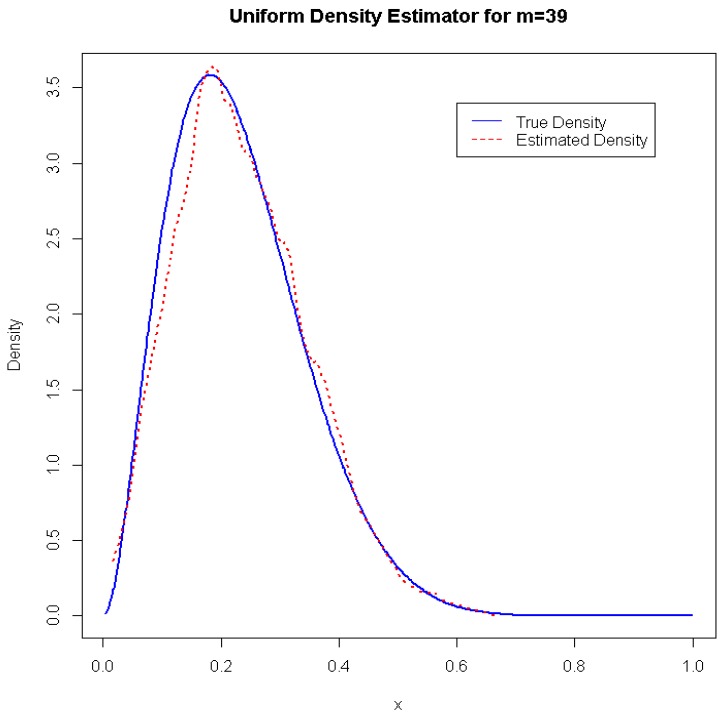



FIGURE 4.3: Uniform Kernel Density Estimate for $m = 39$

From the plots above, it can be observed that as the value of $m$ increases, the estimated density gives a good fit of the true density as expected. However, for $m > 39$, the estimated density tends to exceed the bounds of the true density.

- **Non-Uniform Kernel Density Estimation**

  To estimate a non-uniform kernel, we assume $X \sim Laplace(0, 1)$ to be the true distribution with density function

$$g(x) = \frac{1}{2}e^{-|x|}, \ \infty < x < \infty$$

and distribution function

$$G(x) = \begin{cases} \frac{1}{2}e^{x} & \text{if } x \leqslant 0 \\ 1 - \frac{1}{2}e^{-x} & \text{if } x \geqslant 0 \end{cases}$$

– **Centred Non-Uniform Density Estimation**

Here $G(x)= \frac{1}{2}$ (i.e., symmetric $G$ centred at $x$) and we consider the estimator:

$$D_n(x) = A \cdot \frac{m}{n} \sum_{i=1}^{n} (1 - |1/2 - G(x - X_i)|)^m$$

where $A = \frac{1}{2}g(x)$ is multiplicative factor to make the estimator consistent.

We show the estimate for different values of $m$ (number of splits) in figures below:

$m$= 5 (fig.4.4), 8 (fig.4.5), 15 (fig.4.6).

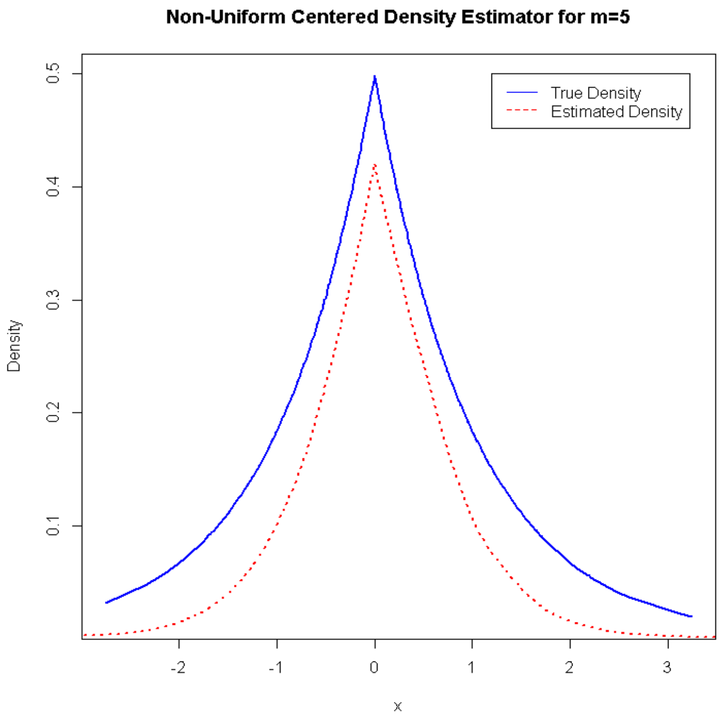



FIGURE 4.4: Centred Non-Uniform Density Estimate for $m = 5$

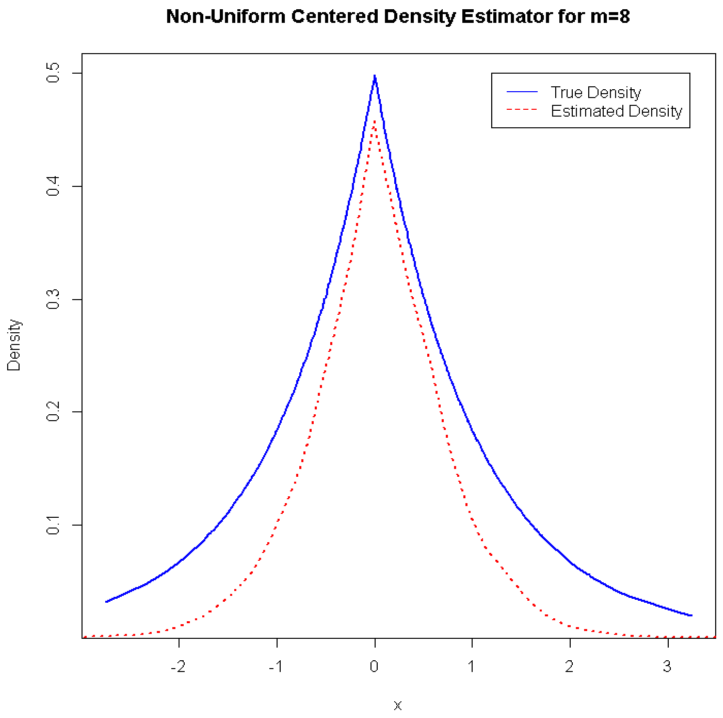



FIGURE 4.5: Centred Non-Uniform Density Estimate for $m = 8$

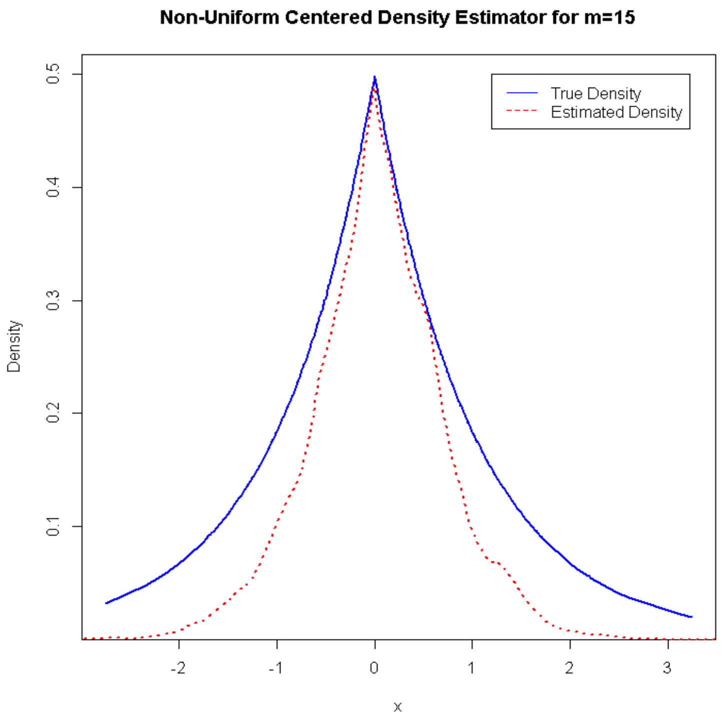



FIGURE 4.6: Centred Non-Uniform Density Estimate for $m = 15$

From the plots above, it can be observed that the estimated density takes the curvature of the true density, but clearly, it does not provide a good fit as $m$ increases.

– **Non-Centred Non-Uniform Density Estimation**

Here, we consider the estimator:

$$D_n(x) = A \cdot \frac{m}{n} \sum_{i=1}^{n} (1 - |G(x) - G(X_i)|)^m$$

where $A = \frac{1}{2} g(x)$ is multiplicative factor to make the estimator consistent.

We show the estimate for different values of $m$ (number of splits) in figures below:

$m$= 5 (fig.4.7), 10 (fig.4.8), 15 (fig.4.9)

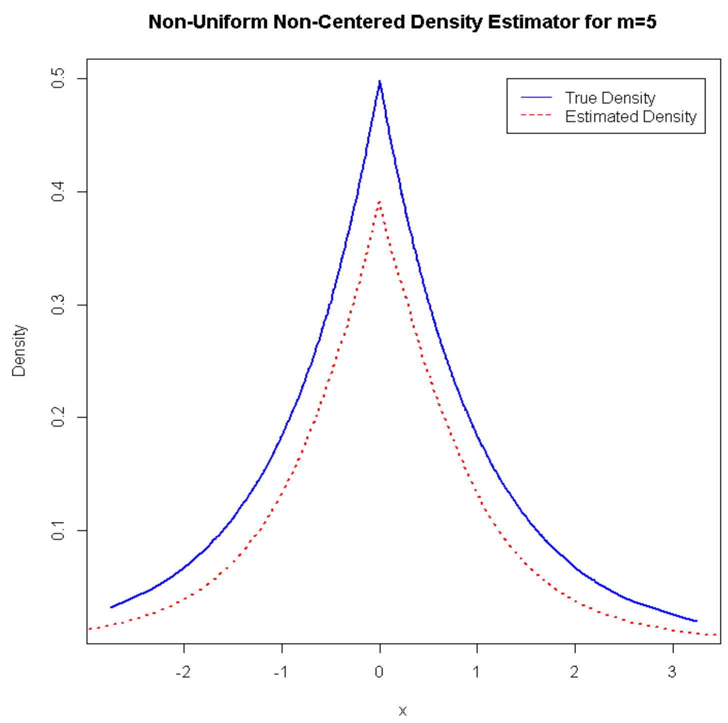



FIGURE 4.7: Non-Centred Non-Uniform Density Estimate for $m = 5$

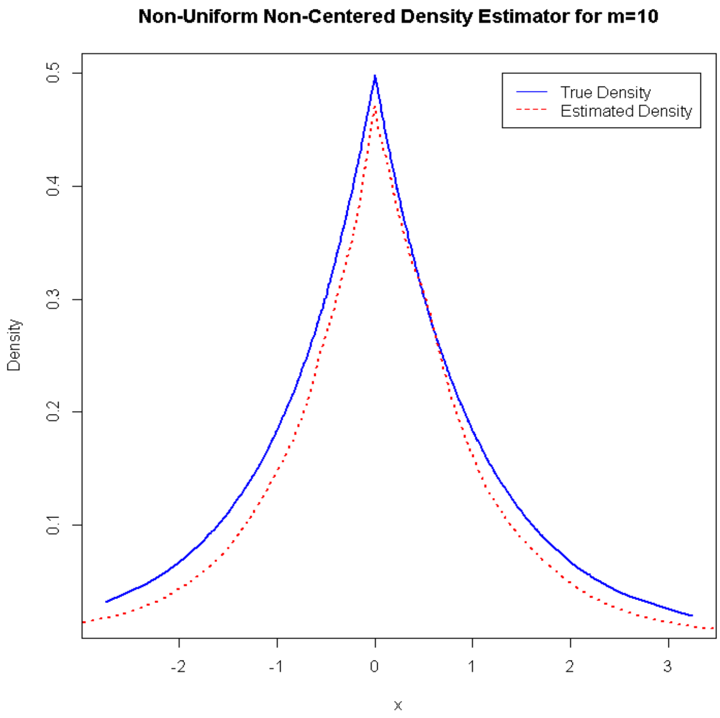



FIGURE 4.8: Non-Centred Non-Uniform Density Estimate for $m = 10$

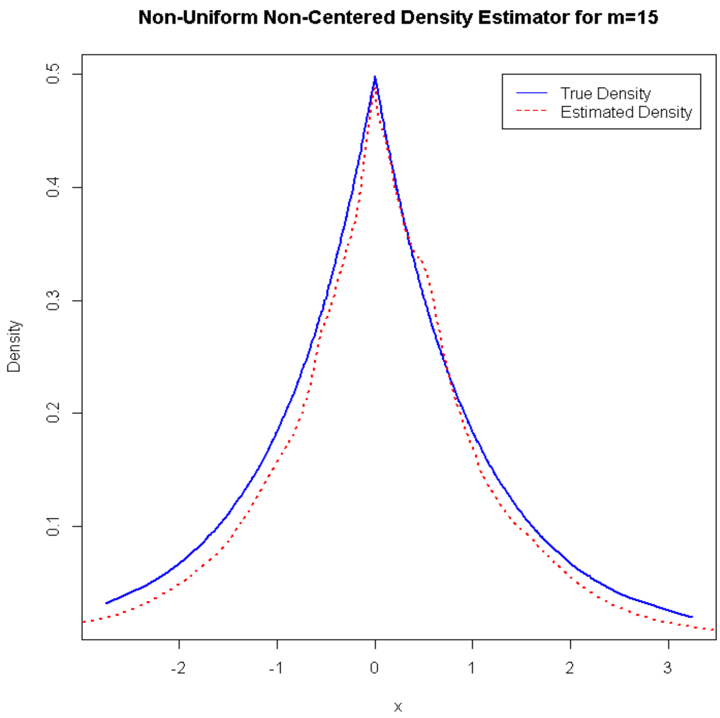



FIGURE 4.9: Non-Centred Non-Uniform Density Estimate for $m = 15$

From the plots above, it can be observed that as we increase the value of $m$, the estimated density gives a good fit of the true density as expected.

## 4.2 Regression Estimation

Consider the regression model

$$Y_i = r(X_i) + 0.075\epsilon_i \quad \text{where, } \epsilon_i \sim \mathcal{N}(0,1)$$

$r(\cdot)$ is the true distribution.

- **Uniform Kernel Regression Estimation**

  Here, we consider the uniform regression estimator

  $$N_n(x) = \frac{m}{n}\sum_{i=1}^{n} Y_i(1 - |x - X_i|)^m$$

  We assume $X \sim Beta(3, 10)$ to be the true distribution with density function

  $$f(x) = \frac{x^2(1-x)^9}{B(3,10)}; 0 \leqslant x \leqslant 1$$

  where $B(3,10) = \frac{\Gamma(3)\Gamma(10)}{\Gamma(13)}$.

  We show the estimate for different values of $m$ (number of splits) in figures below:

  $m$= 15 (fig.4.10), 30 (fig.4.11), 39 (fig.4.12)

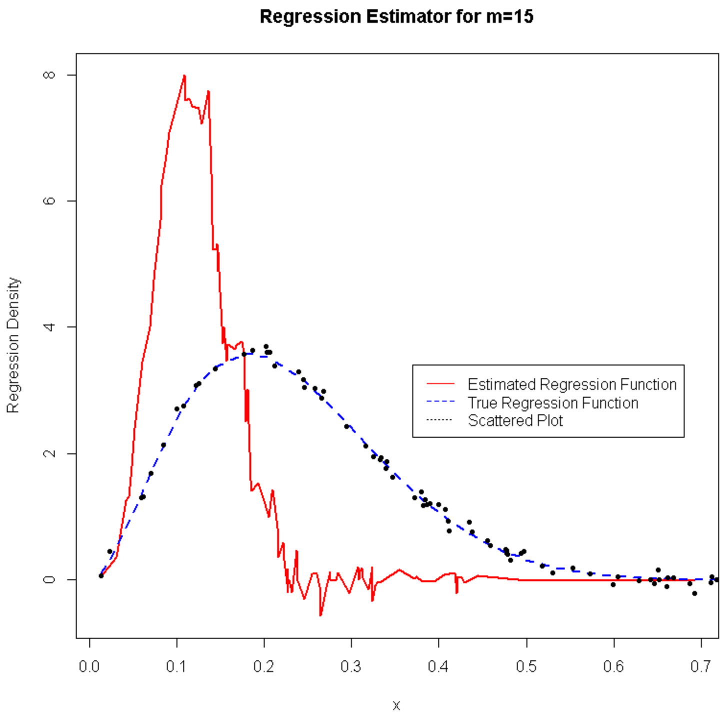



FIGURE 4.10: Uniform Kernel Regression Estimate for $m = 15$

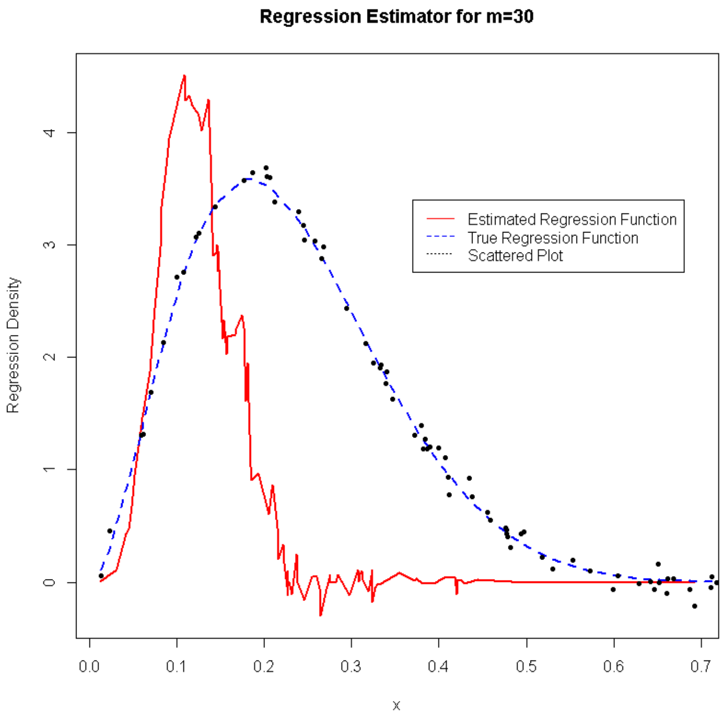



FIGURE 4.11: Uniform Kernel Regression Estimate for $m = 30$

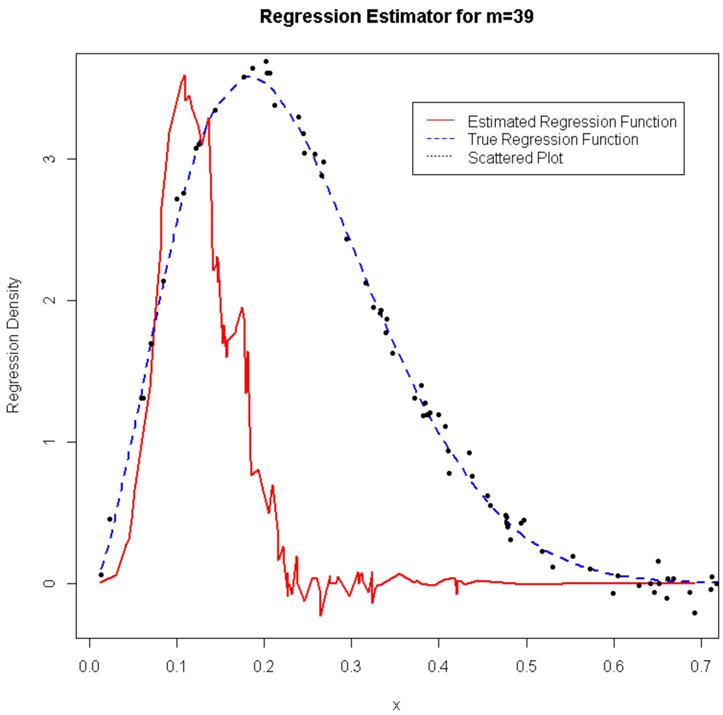



FIGURE 4.12: Uniform Kernel Regression Estimate for $m = 39$

From the plots above, it can be observed that the estimated regression approaches the bounds of the true function. However, it does not provide a good fit.

- **Non-Uniform Kernel Regression Estimation**

  To estimate a non-uniform kernel, we assume $X \sim Laplace(0, 1)$ to be the true distribution with density function

$$g(x) = \frac{1}{2}e^{-|x|}, \ \infty < x < \infty$$

and distribution function

$$G(x) = \begin{cases} \frac{1}{2}e^{x} & \text{if } x \leqslant 0 \\ 1 - \frac{1}{2}e^{-x} & \text{if } x \geqslant 0 \end{cases}$$

– **Centred Non-Uniform Regression Estimation**

Here $G(x)$= $\frac{1}{2}$ (i.e., symmetric $G$ centred at $x$) and we consider the regression estimator

$$N_n(x) = \frac{m}{n}\sum_{i=1}^{n} Y_i(1 - |1/2 - G(x - X_i)|)^m$$

We show the estimate for different values of $m$ (number of splits) in figures below:

$m$= 5 (fig.4.13), 20 (fig.4.14), $n^{1/5}$ (fig.4.15)

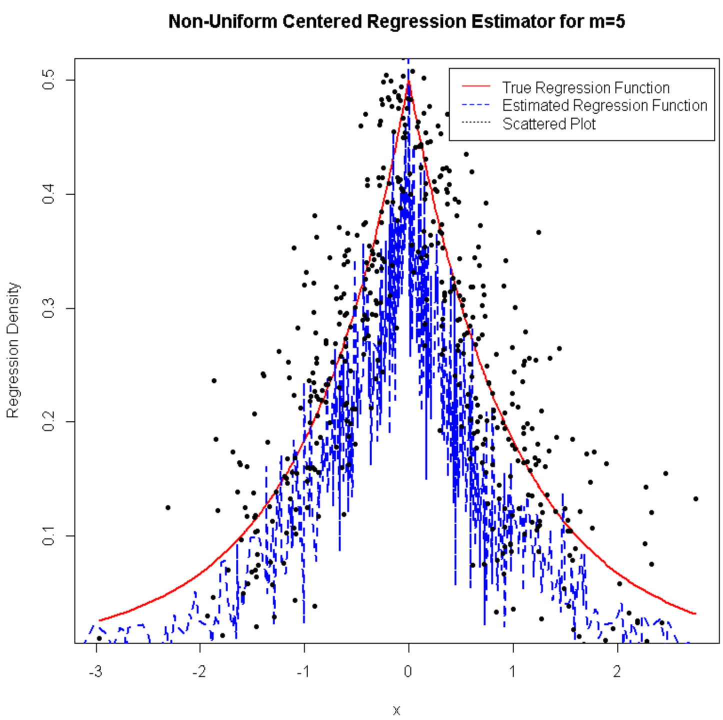



FIGURE 4.13: Centred Non-Uniform Regression Estimate for $m = 5$

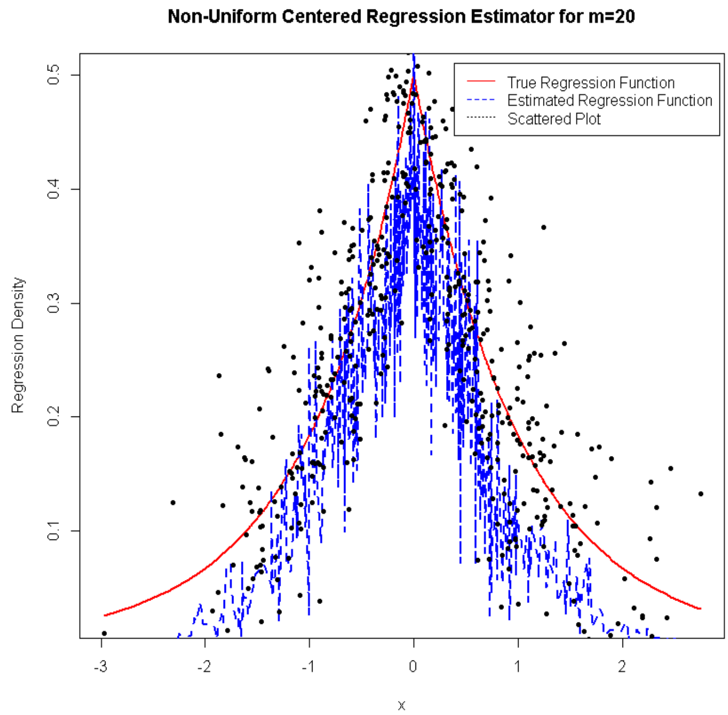



FIGURE 4.14: Centred Non-Uniform Regression Estimate for $m = 20$

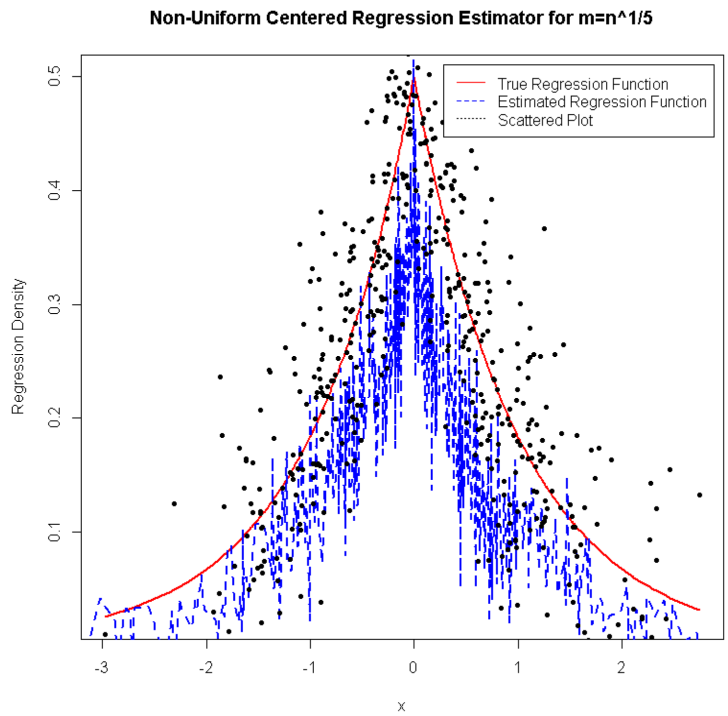



FIGURE 4.15: Centred Non-Uniform Regression Estimate for $m = n^{1/5}$

From the plots above, it can be observed that the estimated regression tries to fit the true function but there is a problem of over-fitting, as it tends to pass through every point. Hence, it does not provide a good fit.

- **Non-Centred Non-Uniform Regression Estimation**

  Here, we consider the regression estimator:

$$N_n(x) = \frac{m}{n} \sum_{i=1}^{n} Y_i (1 - |G(x) - G(X_i)|)^m$$

  We show the estimate for different values of $m$ (number of splits) in figures below:

  $m$= 2 (fig.4.16), 5 (fig.4.17), $n^{1/3}$ (fig.4.18)

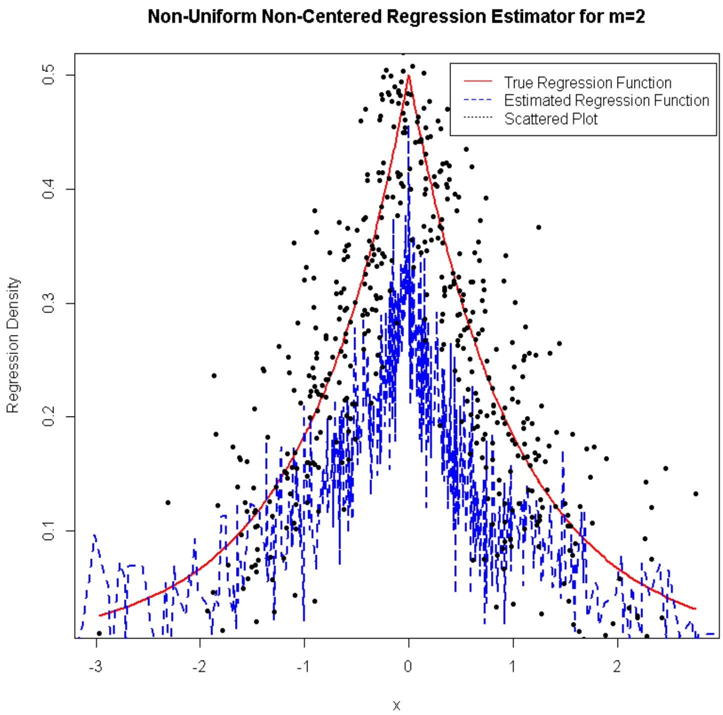



FIGURE 4.16: Non-Centred Non-Uniform Regression Estimate for $m = 2$

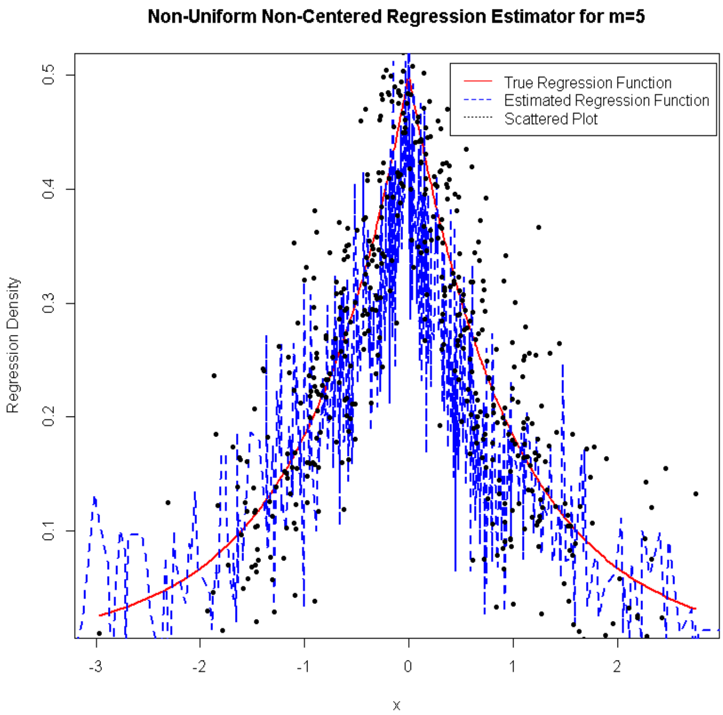



FIGURE 4.17: Non-Centred Non-Uniform Regression Estimate for $m = 5$

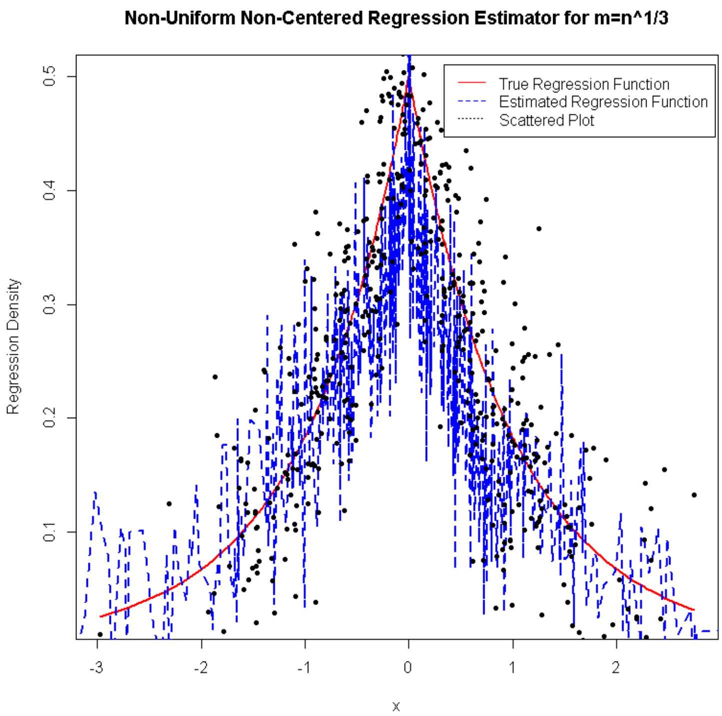



FIGURE 4.18: Non-Centred Non-Uniform Regression Estimate for $m = n^{1/3}$

From the plots above, it can be observed that the estimated regression tries to fit the true function but there is a problem of over-fitting, as it tends to pass through every point. Hence, it does not provide a good fit.

# 5 Conclusion and Future work

In this paper, we sought to define a new estimators for both density and regression kernels. Scornet (Scornet, 2016b) had an expression for kernel $K_k^{uf}$ which only had simple representation for level $k = 1, 2$. However, when levels were greater than 2, the expression became extremely complicated to compute. We used different partitioning scheme to find the general expression for the both uniform and non-uniform kernels and from that we had a density and regression estimators.

We studied the consistency and asymptotic normality of the estimators and finally used simulation to evaluate the performance of the estimators.

In Chapter 4, we realized that the new density estimators provide good fit to the true function but depends heavily on $m$, the number of splits, whiles the new regression estimators did not provide a good fit to the true function.

In future, research can be done on:

- developing procedure for the optimal data-based choice of $m$, and
- extending the results to directional data.
- investigating if theorem 3.2 holds for non-absolute continuous distribution $G$.

# References


Arlot, S., & Genuer, R. (2014). Analysis of purely random forests bias. *arXiv preprint arXiv:1407.3939*.

Biau, G., Devroye, L., & Lugosi, G. (2008). Consistency of random forests and other averaging classifiers. *Journal of Machine Learning Research*, *9*(Sep), 2015–2033.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *Test*, *25*(2), 197–227.

Breiman, L. (2000). *Some infinity theory for predictor ensembles* (Tech. Rep.). Technical Report 579, Statistics Dept. UCB.

Denil, M., Matheson, D., & Freitas, N. (2013). Consistency of online random forests. In *International conference on machine learning* (pp. 1256–1264).

Mentch, L., & Hooker, G. (2016). Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *The Journal of Machine Learning Research*, *17*(1), 841–881.

Nadaraya, E. A. (1964). On estimating regression. *Theory of Probability & Its Applications*, *9*(1), 141–142.

Scornet, E. (2016a). On the asymptotics of random forests. *Journal of Multivariate Analysis*, *146*, 72–83.

Scornet, E. (2016b). Random forests and kernel methods. *IEEE Transactions on Information Theory*, *62*(3), 1485–1500.

Wager, S. (2014). Asymptotic theory for random forests. *arXiv preprint arXiv:1405.0352*.

Watson, G. S. (1964). Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, 359–372.

# A R Codes

## A.1 Uniform density estimator

- **For m=20**

```
set.seed(16)
n=500
x=sort(runif(n))
#true den.
fx= dbeta(x,3,10)

rx=function(v,m,xi){
  b=beta(3,10)
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(2*n))*sum((1-abs(v[i]-xi))^m)
  }
  return(a)
}

xi = rbeta(n,3,10)
v = sort(rbeta(n,3,10))
```

```
vx = rx(v,m=20,xi)

plot(x,fx,type = 'l',col= "blue",lwd=2,ylab = "Density",
     main = "Uniform Density Estimator for m=20")
lines(v,vx,col="Red", lty=3,lwd=2)
legend(0.6,3.4,legend = c("True Density","Estimated Density"),
       col = c('Blue',"red"),lty=1:2)
```

- **For m=32**

```
set.seed(16)
n=500
x=sort(runif(n))
#true den.
fx= dbeta(x,3,10)

rx=function(v,m,xi){
  b=beta(3,10)
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(2*n))*sum((1-abs(v[i]-xi))^m)
  }
  return(a)
}

xi = rbeta(n,3,10)
v = sort(rbeta(n,3,10))
vx = rx(v,m=32,xi)
```

```
plot(x,fx,type = 'l',col= "blue",lwd=2,ylab = "Density",
     main = "Uniform Density Estimator for m=32")
lines(v,vx,col="Red", lty=3,lwd=2)
legend(0.6,3.4,legend = c("True Density","Estimated Density"),
       col = c('Blue',"red"),lty=1:2)
```

- **For m=39**

```
set.seed(16)
n=500
x=sort(runif(n))
#true den.
fx= dbeta(x,3,10)

rx=function(v,m,xi){
  b=beta(3,10)
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(2*n))*sum((1-abs(v[i]-xi))^m)
  }
  return(a)
}

xi = rbeta(n,3,10)
v = sort(rbeta(n,3,10))
vx = rx(v,m=39,xi)
```

```
  plot(x,fx,type = 'l',col= "blue",lwd=2,ylab = "Density",
       main = "Uniform Density Estimator for m=39")
  lines(v,vx,col="Red", lty=3,lwd=2)
  legend(0.6,3.4,legend = c("True Density","Estimated Density"),
         col = c('Blue',"red"),lty=1:2)
```

## A.2 Non-Uniform Density Estimator

### A.2.1 Centered Density Estimator

- **m=5**

```
set.seed(16)
n=500
library(rmutil)
#true density
x=sort(rnorm(n))
gx= dlaplace(x)


rx=function(v,m,xi){
a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(dlaplace(v[i])/2)*(m/n)*sum((1-abs(0.5-pnorm(v[i]-xi)))^m)
  }
  return(a)
}
```

```
xi = rnorm(n)

v = sort(rlaplace(n))

vx = rx(v,m=5,xi)


plot(x,gx,type = 'l',col= "blue",lwd=2,ylab = "Density",

    main = "Non-Uniform Centered Density Estimator for m=5")

lines(v,vx,col="Red", lty=3,lwd=2)

legend(1.2,0.5,legend = c("True Density","Estimated Density"),

       col = c('Blue',"red"),lty=1:2)
```

- **m=8**

```
set.seed(16)

n=500

library(rmutil)

#true density

x=sort(rnorm(n))

gx= dlaplace(x)



rx=function(v,m,xi){

a = rep(0,length(v))

  for(i in 1:(length(v))){

    a[i]=(dlaplace(v[i])/2)*(m/n)*sum((1-abs(0.5-pnorm(v[i]-xi)))^m)

  }

  return(a)

}
```

```
xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(v,m=8,xi)

plot(x,gx,type = 'l',col= "blue",lwd=2,ylab = "Density",
    main = "Non-Uniform Centered Density Estimator for m=8")
lines(v,vx,col="Red", lty=3,lwd=2)
legend(1.2,0.5,legend = c("True Density","Estimated Density"),
       col = c('Blue',"red"),lty=1:2)
```

- **m= 15**

```
set.seed(16)
n=500
library(rmutil)
#true density
x=sort(rnorm(n))
gx= dlaplace(x)


rx=function(v,m,xi){
a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(dlaplace(v[i])/2)*(m/n)*sum((1-abs(0.5-pnorm(v[i]-xi)))^m)
  }
  return(a)
}
```

```
xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(v,m=15,xi)

plot(x,gx,type = 'l',col= "blue",lwd=2,ylab = "Density",
    main = "Non-Uniform Centered Density Estimator for m=15")
lines(v,vx,col="Red", lty=3,lwd=2)
legend(1.2,0.5,legend = c("True Density","Estimated Density"),
       col = c('Blue',"red"),lty=1:2)
```

### A.2.2 Non-Centered Density Estimator

- **m= 5**

```
set.seed(16)
n=500
library(rmutil)
#true density
x=sort(rnorm(n))
gx= dlaplace(x)

rx=function(v,m,xi){
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(dlaplace(v[i])/2)*(m/n)*sum((1-abs(plaplace(v[i])-pnorm(xi)))^m)
  }
  return(a)
}
```

```
xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(v,m=5,xi)
plot(x,gx,type = 'l',col= "blue",lwd=2,ylab = "Density",
main = "Non-Uniform Non-Centered Density Estimator for m=5")
lines(v,vx,col="Red", lty=3,lwd=2)
legend(1.3,0.5,legend = c("True Density","Estimated Density"),
       col = c('Blue',"red"),lty=1:2)
```

- **m= 10**

```
set.seed(16)
n=500
library(rmutil)
#true density
x=sort(rnorm(n))
gx= dlaplace(x)

rx=function(v,m,xi){
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(dlaplace(v[i])/2)*(m/n)*sum((1-abs(plaplace(v[i])-pnorm(xi)))^m)
  }
  return(a)
}
```

```
xi = rnorm(n)

v = sort(rlaplace(n))

vx = rx(v,m=10,xi)

plot(x,gx,type = 'l',col= "blue",lwd=2,ylab = "Density",

main = "Non-Uniform Non-Centered Density Estimator for m=10")

lines(v,vx,col="Red", lty=3,lwd=2)

legend(1.3,0.5,legend = c("True Density","Estimated Density"),

       col = c('Blue',"red"),lty=1:2)
```

- **m= 15**

```
set.seed(16)

n=500

library(rmutil)

#true density

x=sort(rnorm(n))

gx= dlaplace(x)


rx=function(v,m,xi){

  a = rep(0,length(v))

  for(i in 1:(length(v))){

    a[i]=(dlaplace(v[i])/2)*(m/n)*sum((1-abs(plaplace(v[i])-pnorm(xi)))^m)

  }

  return(a)

}


xi = rnorm(n)
```

```
v = sort(rlaplace(n))
vx = rx(v,m=15,xi)
plot(x,gx,type = 'l',col= "blue",lwd=2,ylab = "Density",
main = "Non-Uniform Non-Centered Density Estimator for m=15")
lines(v,vx,col="Red", lty=3,lwd=2)
legend(1.3,0.5,legend = c("True Density","Estimated Density"),
       col = c('Blue',"red"),lty=1:2)
```

## A.3 Uniform Regression Estimator

- **m= 15**

```
set.seed(1)
n=500
x=sort(runif(n))
#true density
fx= dbeta(x,3,10)
x1=sort(rbeta(n,3,10))
yi=fx+(0.075)*rnorm(n)
rx=function(y,v,m,xi){
  #b=beta(3,10)
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(15-(1/(b*n)))/n)*sum(y[i]*(1-abs(v[i]-xi))^m)
  }
  return(a)
}
```

```
xi = rbeta(n,3,10)
v = sort(rbeta(n,3,10))
vx = rx(y=yi,v,m=15,xi)

plot(v,vx,type = "l", col='red', lwd=2,xlab="x", ylab =
'Regression Density',
main = "Regression Estimator for m=15") #estimated regression
lines(x,fx, lwd=2, col='Blue', lty=2) #True regrssion function
points(x,yi, col='black', pch=20) #Scattered
legend(0.37,3.4,legend = c("Estimated Regression Function",
"True Regression Function"
,"Scattered Plot"),col = c('red',"blue","black"),lty=c(1,2,3))
```

- **m= 30**

```
set.seed(1)
n=100
x=sort(runif(n))
#true density
fx= dbeta(x,3,10)
x1=sort(rbeta(n,3,10))
yi=fx+(0.075)*rnorm(n)
rx=function(y,v,m,xi){
  b=beta(3,10)
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=((15-(1/(b*n)))/n)*sum(y[i]*(1-abs(v[i]-xi))^m)
```

```
  }
  return(a)
}

xi = rbeta(n,3,10)
v = sort(rbeta(n,3,10))
vx = rx(y=yi,v,m=30,xi)

plot(v,vx,type = "l", col='red', lwd=2, ylab = 'Regression Density',
main = "Regression Estimator for m=30") #estimated regression
lines(x,fx, lwd=2, col='Blue', lty=2) #True regression function
points(x,yi, col='black', pch=20) #Scattered plot
legend(0.37,3.4,legend = c("Estimated Regression Function",
"True Regression Function","Scattered Plot"),
col = c('red',"blue","black"),lty=1:2:3)
```

- **m=39**

```
set.seed(1)
n=100
x=sort(runif(n))
#true density
fx= dbeta(x,3,10)
x1=sort(rbeta(n,3,10))
yi=fx+(0.075)*rnorm(n)
rx=function(y,v,m,xi){
  b=beta(3,10)
  a = rep(0,length(v))
  for(i in 1:(length(v))){
```

```
    a[i]=((15-(1/(b*n)))/n)*sum(y[i]*(1-abs(v[i]-xi))^m)
  }
  return(a)
}
xi = rbeta(n,3,10)
v = sort(rbeta(n,3,10))
vx = rx(y=yi,v,m=39,xi)

plot(v,vx,type = "l", col='red', lwd=2, ylab = 'Regression Density',
main = "Regression Estimator for m=39") #estimated regression
lines(x,fx, lwd=2, col='Blue', lty=2) #True regression function
points(x,yi, col='black', pch=20) #Scattered plot
legend(0.37,3.4,legend = c("Estimated Regression Function",
"True Regression Function","Scattered Plot"),
col = c('red',"blue","black"),lty=1:2:3)
```

## A.4 Non-Uniform Regression Estimator

### A.4.1 Centered Regression Estimator

- **m=5**

```
set.seed(7)
n=500
x=sort(rnorm(n))
#true density
fx= dlaplace(x)
yi=fx+(0.075)*rnorm(n)
```

```
rx=function(y,v,m,xi){
    a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(2*n))*sum(y[i]*(1-abs(0.5-pnorm(v[i]-xi)))^m)
    }
  return(a)
}

xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(y=yi,v,m=5,xi)

plot(x,fx,type = "l", col='red', lwd=2,xlab="x", ylab =
'Regression Density',
main = "Non-Uniform Centered Regression Estimator for m=5")
#True regression
lines(v,vx, lwd=2, col='Blue', lty=2) #Estimated regression function
points(x,yi, col='black', pch=20) #Scattered

legend(0.383,0.51,legend = c("True Regression Function",
"Estimated Regression Function","Scattered Plot"),
col = c('red',"blue","black"),lty=c(1,2,3))
```

- **m=20**

```
set.seed(7)
n=500
x=sort(rnorm(n))
```

```
#true density
fx= dlaplace(x)
yi=fx+(0.075)*rnorm(n)

rx=function(y,v,m,xi){
    a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(2*n))*sum(y[i]*(1-abs(0.5-pnorm(v[i]-xi)))^m)
    }
  return(a)
}

xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(y=yi,v,m=20,xi)

plot(x,fx,type = "l", col='red', lwd=2,xlab="x", ylab =
'Regression Density',
main = "Non-Uniform Centered Regression Estimator for m=20")
#True regression
lines(v,vx, lwd=2, col='Blue', lty=2) #Estimated regression function
points(x,yi, col='black', pch=20) #Scattered

legend(0.383,0.51,legend = c("True Regression Function",
"Estimated Regression Function","Scattered Plot"),
col = c('red',"blue","black"),lty=c(1,2,3))
```

- **m=n**$^{1/5}$

```
library(rmutil)
set.seed(7)
n=500
x=sort(rnorm(n))
#true density
fx= dlaplace(x)
# x1=sort(rbeta(n,3,10))
yi=fx+(0.075)*rnorm(n)

rx=function(y,v,m,xi){
    a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(2*n))*sum(y[i]*(1-abs(0.5-pnorm(v[i]-xi)))^m)
  }
  return(a)
}

xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(y=yi,v,m=(n^(1/5)),xi)

plot(x,fx,type = "l", col='red', lwd=2,xlab="x", ylab =
'Regression Density',
 main = "Non-Uniform Centered Regression Estimator for m=n^1/5")
 #True regression
lines(v,vx, lwd=2, col='Blue', lty=2) #Estimated regrssion function
points(x,yi, col='black', pch=20) #Scattered
```

```
legend(0.28,0.51,legend = c("True Regression Function",
"Estimated Regression Function","Scattered Plot"),
col = c('red',"blue","black"),lty=c(1,2,3))
```

### A.4.2 Non-Centered Regression Estimator

- **m=2**

```
set.seed(7)
n=500
x=sort(rnorm(n))
#true density
fx= dlaplace(x)
yi=fx+(0.075)*rnorm(n)

rx=function(y,v,m,xi){
  a = rep(0,length(v))
  for(i in 1:(length(v))){
    a[i]=(m/(1.8*n))*sum(y[i]*(1-abs(plaplace(v[i])-pnorm(xi)))^m)
  }
  return(a)
}

xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(y=yi,v,m=2,xi)

plot(x,fx,type = "l", col='red', lwd=2,xlab="x", ylab =
```

```
'Regression Density',
 main = "Non-Uniform Non-Centered Regression Estimator for m=2")
 #True regression
lines(v,vx, lwd=2, col='Blue', lty=2) #estimated regression function
points(x,yi, col='black', pch=20) #Scattered
legend(0.4,0.51,legend = c("True Regression Function",
"Estimated Regression Function","Scattered Plot"),
col =c('red',"blue","black"),lty=c(1,2,3))
```

- **m=5**

```
set.seed(7)
n=500
x=sort(rnorm(n))
#true density
fx= dlaplace(x)
yi=fx+(0.075)*rnorm(n)

rx=function(y,v,m,xi){
a = rep(0,length(v))
for(i in 1:(length(v))){
a[i]=(m/(1.8*n))*sum(y[i]*(1-abs(plaplace(v[i])-pnorm(xi)))^m)
}
return(a)
}

xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(y=yi,v,m=5,xi)
```

```
plot(x,fx,type = "l", col='red', lwd=2,xlab="x", ylab =
'Regression Density',
 main = "Non-Uniform Non-Centered Regression Estimator for m=5")
 #True regression
lines(v,vx, lwd=2, col='Blue', lty=2) #estimated regression function
points(x,yi, col='black', pch=20) #Scattered
legend(0.4,0.51,legend = c("True Regression Function",
"Estimated Regression Function","Scattered Plot"),
col = c('red',"blue","black"),lty=c(1,2,3))
```

- **m=n**$^{1/3}$

```
set.seed(7)
n=500
x=sort(rnorm(n))
#true density
fx= dlaplace(x)

yi=fx+(0.075)*rnorm(n)

rx=function(y,v,m,xi){
a = rep(0,length(v))
for(i in 1:(length(v))){
a[i]=(m/(2*n))*sum(y[i]*(1-abs(plaplace(v[i])-pnorm(xi)))^m)
}
return(a)
}
```

```
xi = rnorm(n)
v = sort(rlaplace(n))
vx = rx(y=yi,v,m=(n^(1/3)),xi)

plot(x,fx,type = "l", col='red', lwd=2,xlab="x", ylab =
'Regression Density',
main = "Non-Uniform Non-Centered Regression Estimator for m=n^1/3")
#True regression
lines(v,vx, lwd=2, col='Blue', lty=2) #estimated regrssion function
points(x,yi, col='black', pch=20) #Scattered
legend(0.34,0.51,legend = c("True Regression Function",
"Estimated Regression Function","Scattered Plot"),
col =c('red',"blue","black"),lty=c(1,2,3))
```