

CHANGE-POINT ANALYSIS AND PATTERN
TRACKING IN DATA

BY

DALAL ALI ALZAHIRANI

A Thesis Presented to the
COLLEGE OF GRADUATE AND INTERDISCIPLINARY STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

APPLIED STATISTICS

MAY 2024

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **DALAL ALI ALZHRANI** under the direction of her thesis adviser and approved by her thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN APPLIED STATISTICS**.

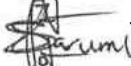
Thesis Committee



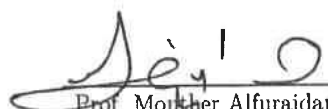
Dr. Maher Boudabra (Adviser)



Dr. Emmanuel Afuecheta (Member)



Dr. Ibrahim Sarumi (Member)



Prof. Morkher Alfuraidean
Department Chairman



Prof. Suliman Alhomidan
Dean of Graduate Studies

Date



©Dalal Ali Alzahrani
2024

*This thesis is dedicated to my family, whose endless love,
support, and encouragement have been my source of
strength*

ACKNOWLEDGMENTS

First and foremost, I want to express my sincere gratitude to Allah, the Almighty, whose constant direction and assistance have been crucial to me throughout every phase of my academic journey. I thank and praise Allah for His boundless mercy, which has lightened my path from the start. Had it not been for His heavenly intervention, this effort would never have commenced. I sincerely thank Almighty Allah for providing me with the health, strength, and ability to successfully complete my thesis.

To my beloved family, I am extremely grateful to my father and mother, Ali Alzahrani and Nora Alzahrani, for their forever love, support, and encouragement throughout my life. Words can't express the extent of my gratitude to them. Their endless prayers, love, and patience have been my guiding light, motivating me to pursue higher education and aim for distinction. Their encouragement for me to pursue further study has been a continual source of inspiration, and I sincerely appreciate that. Only Allah can reward my parents for their boundless kindness and generosity. I also want to thank my siblings, Manal, Abdul Rahman, Abdullah, and Ibrahim, for their support and encouragement. Thank you for standing by me and believing in my dreams. Your encouragement means the world to me.

I want to express great appreciation for providing an encouraging environment for study and research at King Fahd University of Petroleum and Minerals (KFUPM), especially to the Department of Mathematics and Statistics. My sincere gratitude is extended to Prof. Monther Rashed Alfuraidean, the Chairman of the Department of Mathematics and Statistics. My academic journey has been greatly enhanced by your valuable mentorship, strong support, and steadfast encouragement. I am incredibly grateful for how much your wisdom, dedication, and exemplary leadership have impacted and enhanced my academic endeavors.

I would like to sincerely thank Dr. Maher Boudabra, my thesis advisor, for all of his guidance and encouragement in making this work possible. His expertise and advice have greatly influenced the direction of my research. I am pleased to express my heartfelt gratitude to Dr. Emmanuel Afuecheta and Dr. Ibrahim Sarumi for their invaluable insights, time, and commitment to ensuring the level of quality and rigor of this work. Their expertise and feedback have been extremely beneficial to accomplish this thesis more successfully.

My sincere appreciation goes out to my work colleagues, professors, and friends for their encouragement and support throughout the completion of my master's degree.

As I conclude this acknowledgment, I want to express my deepest appreciation and best wishes to everyone who has supported me in completing this thesis. Amidst the challenges and obstacles faced throughout this journey, I am very thankful for the lessons learned from those who may have attempted to hinder my progress. Their

attempts at disrupting my efforts only served to strengthen my resolve and reconfirm my commitment to succeed. I am grateful for the opportunity to overcome difficulties and emerge stronger, more resilient, and more determined than ever before. Thank you to everyone who has contributed to shaping my journey, either positively or negatively, as this experience has assisted me grow and develop.

Contents

ACKNOWLEDGMENTS	vi
LIST OF TABLES	xii
LIST OF FIGURES	xiii
ABSTRACT (ENGLISH)	xv
ABSTRACT (ARABIC)	xvii
 CHAPTER 1 INTRODUCTION	 1
1.1 Background	1
1.2 Preliminaries	3
1.2.1 Foundational Concepts	3
1.2.2 Overview of Techniques and Methods:	15
1.3 Probability Distributions	23
1.4 Problem Statement	26
1.5 Objectives	26
1.6 Thesis Structure	27
 CHAPTER 2 LITERATURE REVIEW	 29
2.1 Background on Change Point Detection	30
2.2 Change-Point Analysis Methodologies	31
2.3 Pattern Tracking Techniques	39
2.4 Applications in Different Domains	40
2.5 Challenges and Limitations	41

CHAPTER 3	PROPOSED METHODOLOGY: CHANGE POINT ANALYSIS	42
3.1	Introduction	42
3.2	The Method	44
3.2.1	Measurement of Error using Euclidean Distance	45
3.2.2	Tracking the change of the patterns using optimal partitions . .	45
3.2.3	Example	47
3.3	Existence of Regression Minimizer in Function Family $\mathcal{F}$	53
3.4	Algorithm: Data Partitioning and Change Point Detection	55
3.4.1	Overview	55
3.4.2	Algorithm Steps	56
3.4.3	Linear Approximation Function	56
3.4.4	Building the R Code for Change Point Detection	56
3.5	Statistical Analysis of Optimal Points $(a_1, \dots, a_n)$	60
3.5.1	Purpose of Statistical Analysis for Cutpoints	65
CHAPTER 4	PROPOSED METHODOLOGY: REAL-TIME TRACKING	69
4.1	The Method: Real Time Tracking	70
4.1.1	Error Threshold Determination	71
4.2	Analysis of Dynamic Data Evolution	72
4.3	Algorithm: Real-Time Tracking for Dynamic Data Analysis	73
4.3.1	Overview	74
4.3.2	Algorithm Steps	74
4.3.3	Flexibility of Model Selection	74
4.3.4	R Code Implementation	75
CHAPTER 5	APPLICATION AND STATISTICAL ANALYSIS	76
5.1	Data Description	76
5.2	Comparison with Global Regression Model	78
5.2.1	Global Regression Model	79

5.2.2	Change Point Detection Methods	82
5.2.3	Method 1: Tracking the change of the patterns using optimal partitions	83
5.2.4	Method 2: Real Time Tracking	87
5.2.5	Statistical Analysis	90
5.3	Application of Bayesian Change Point Analysis	94
CHAPTER 6 REGRESSION WITH DEEP NEURAL NETWORKS		97
6.1	Brief Overview of Neural Networks	98
6.1.1	Significance of neural networks	98
6.1.2	Deep Neural Networks (DNNs)	99
6.1.3	Activation functions	100
6.2	Numerical Example	102
6.3	Discussion	103
CHAPTER 7 CONCLUSION AND FUTURE WORKS		106
7.1	Summary of findings	106
7.2	Future works	108
APPENDIX A R CODES - FIRST METHOD		109
APPENDIX B R CODES - SECOND METHOD		112
REFERENCES		113
VITAE		120

List of Tables

Table 3.1	Mean values for the distribution and cutoff point vector across different distributions.	61
Table 5.1	Performance Metrics for Two Models	86
Table 5.2	KS Test p-values for Different Distributions and Bootstrap Sizes . .	92

List of Figures

Figure 1.1	Scatterplot	4
Figure 1.2	Correlation	5
Figure 1.3	Time Series	6
Figure 1.4	Linear Regression	7
Figure 1.5	Simple Linear Regression	8
Figure 1.6	Multiple Linear Regression	9
Figure 1.7	K-Means Clustering	12
Figure 1.8	Change point	14
Figure 1.9	Trend Analysis	15
Figure 1.10	Bootstrap resampling	20
Figure 2.1	Segmented regression model	37
Figure 3.1	Multimodal scatter plot.	43
Figure 3.2	The point $x = 200$ is the best cut-off point.	43
Figure 3.3	(a)	58
Figure 3.4	(b)	59
Figure 3.5	CDF Plot: Standard Normal	66
Figure 3.6	CDF Plot: Uniform	67
Figure 3.7	CDF Plot: Exponential	67
Figure 3.8	CDF Plot: Gamma	68
Figure 3.9	CDF Plot: Beta	68
Figure 5.1	Time Series Plot	77
Figure 5.2	Scatterplot	78
Figure 5.3	Fitted Line	81

Figure 5.4	Single Change Point	84
Figure 5.5	Two Change Points	84
Figure 5.6	Percentage Error	86
Figure 5.7	Real Time Tracking	89
Figure 5.8	Bootstrap Distribution	91
Figure 5.9	CDF Plots for Different Sample Sizes	93
Figure 5.10	Plot illustrating the identified change points in the mean	95
Figure 5.11	Plot illustrating the identified change points in the variance	96
Figure 6.1	Deep neural network architecture	99
Figure 6.2	Predictions Made by Neural Network	103
Figure 6.3	Absolute Error for Testing Data	104
Figure 6.4	Relative Error for Training Data	105

THESIS ABSTRACT

NAME: Dalal Ali Alzahrani
TITLE OF STUDY: Change-Point Analysis and Pattern Tracking in Data
MAJOR FIELD: Applied Statistics
DATE OF DEGREE: MAY 2024

Change point analysis is an active area of study in various domains, including statistics. The main purpose of this thesis is to identify change points and estimate the number and location of structural changes in data sets. We have developed novel statistical methods to address the limitations of existing methodologies and the challenges associated with change point detection, helping us to understand better the nature of the processes that underlie observable data. In the first proposed method, change points are found and data is partitioned using an optimization approach to minimize errors using the Euclidean Distance. Furthermore, the second proposed method provides a dynamic framework for handling changing data. It takes into account the complicated nature of real-time data evolution and operates on a dataset regardless of its finiteness. This thesis examines change point detection in real datasets (power consumption) and compares the performance of local regression to global regression.

Using advanced statistical techniques and bootstrapping by R software, the study shows that local regression produces considerably smaller Euclidean distances than global regression. These findings apply to a wide range of datasets. Moreover, the data was examined using Deep Neural Networks (DNNs) in the same real data (power consumption) to demonstrate their effectiveness in addressing the research challenge. The neural network approach achieved a similar outcome, providing comparable accuracy. This demonstrates our method's efficacy and validity, particularly in cases when data separating is important to improving statistical models' performance.

ملخص الرسالة

الاسم: دلال بنت علي الزهراني

عنوان الدراسة: تحليل نقاط التغيير وتتبع الأنماط في البيانات

التخصص: الإحصاء التطبيقي

تاريخ الدرجة العلمية: مايو 2024م

يعد تحليل نقاط التغيير مجالاً نشطاً للدراسة في العديد من المجالات، بما في ذلك مجال الإحصاء. الغرض الرئيسي من هذه الأطروحة هو تحديد نقاط تغير النمط وتقدير عددها ومواقع التغييرات الهيكلية في مجموعات البيانات. لقد قمنا بتطوير أساليب إحصائية جديدة لمعالجة القيود المفروضة على الطرق الحالية والتحديات المرتبطة باكتشاف نقاط التغيير، مما يساعدنا على فهم أفضل لطبيعة العمليات التي تكمن وراء البيانات. باستخدام الطريقة المقترحة الأولى، تم تحديد نقاط التغيير وتقسيم البيانات بالاعتماد على أسلوب التحسين optimization method لتقليل الأخطاء وذلك بحساب المسافة الإقليدية Euclidean Distance. توفر الطريقة المقترحة الثانية إطاراً ديناميكياً للتعامل مع البيانات المتغيرة، فهي تركز على الطبيعة المعقدة للبيانات حيث تعمل على مجموعة البيانات بغض النظر عن محدوديتها. تناولت هذه الأطروحة تحديد نقطة التغيير وذلك بالتطبيق على مجموعة بيانات حقيقية لاستهلاك الطاقة وتم مقارنة أداء نموذج الانحدار المحلي بنموذج الانحدار العام. أظهرت نتائج الدراسة أن المسافة الإقليدية الناتجة عن نموذج الانحدار المحلي أصغر بكثير من المسافة الإقليدية الناتجة عن نموذج الانحدار العام وذلك باستخدام الطرق الإحصائية المقترحة. كما أظهرت الدراسة أن النتائج السابقة قابلة للتطبيق على مجموعة واسعة من البيانات باستخدام طريقة bootstrapping. علاوة على ذلك، تم دراسة البيانات المتعلقة باستهلاك الطاقة باستخدام الشبكات العصبية العميقة (DNNs) وذلك لإثبات فعاليتها لتدعيم الهدف البحثي. حققت طريقة الشبكة العصبية نتيجة مماثلة، وهذا يوضح فعالية الطرق الإحصائية المقترحة، لا سيما في الحالات التي يكون فيها تقسيم البيانات مهماً لتحسين أداء النماذج الإحصائية.

CHAPTER 1

INTRODUCTION

This chapter introduces the topics being explored, provides background for how the problem was framed, and states the questions that will be addressed in subsequent chapters.

1.1 Background

Changes happen everywhere in everyday life. Understanding unexpected and significant shifts is a prerequisite for comprehending the mechanics of a variety of scientific activities. Time series data are sequential observations that show the dynamic behavior of a system. Change point detection (CPD), which uses time series observations to identify the time associated with a shift in the temporal development of a system or process, may be affected by a sudden shift in the dynamics or distribution of the underlying system/process that is producing the data. The statistics and machine learning community has shown a great deal of interest in this CPD challenge throughout the years due to its many applications in various sectors,

such as healthcare diagnosis [27], and climate studies [35], among others. Many methodologies and algorithms have been developed as a result of the CPD problem; we provide an in-depth review of the most recent and oldest approaches in Chapter 2.

Disregarding a change point may result in possibly incorrect and misleading results. In today’s data-driven world, the capacity to monitor and comprehend data patterns is essential. These patterns, ranging from scatter plots to time series data, offer critical insights into areas. As the world becomes more interconnected, decision-makers must be able to detect and respond to shifts in data patterns. Effective decision-making requires proficiency in spotting major data changes, such as anomalies, trends, and seasonality [1].

Motivated by the difficulties encountered in dealing with change point detection problems and the limitations of current methodologies, we develop unique statistical methods to solve these issues. Our methodologies are not limited by standard time series approaches and may be used in a variety of data forms, including scatter plots.

This thesis aims to explain the nature of processes hidden behind observable scatter plots by introducing and critically analyzing a variety of novel statistical techniques for change-point identification. Despite their numerous potential uses, the purpose of these techniques is to detect unexpected shifts in data and adaptively partition them into nearly stationary intervals of varying lengths. The ultimate

goal of this research is to improve the interpretability of specific observed scatter plot patterns, which will allow for more perceptive analysis and meaningful data interpretation.

1.2 Preliminaries

This section provides an overview of the key components of our research. Our primary goal is to offer readers a thorough comprehension of the fundamental concepts and research methods that support our argument. By delving into these fundamental components, we want to create a solid foundation for further in-depth study and analysis in subsequent sections.

1.2.1 Foundational Concepts

In this part, we will discuss and explain the essential concepts that will serve as the basis for our research.

1. Scatter plot:

A scatter plot in 2D is a plot of data points on a Cartesian plane, each representing the value of two variables. This visualization form is particularly beneficial for identifying patterns or connections between variables, as well as locating outliers in data.

2. Correlation

Correlation is a statistical measure that describes the relationship between the relative changes of two variables. The points on a line or curve represent how the variables are associated. Three scenarios are possible for demonstrating the relationship between the two variables. **1. A positive correlation. 2. A negative correlation.**

3. No Correlation. When the points on a scatter plot rise and move from left to right, a **positive correlation** is seen. When the points in the scatter graph drop as they move from left to right, it's known as a **negative correlation**. There is **no correlation** when it is difficult to identify whether the values are rising or falling due to scattered points on the graph.

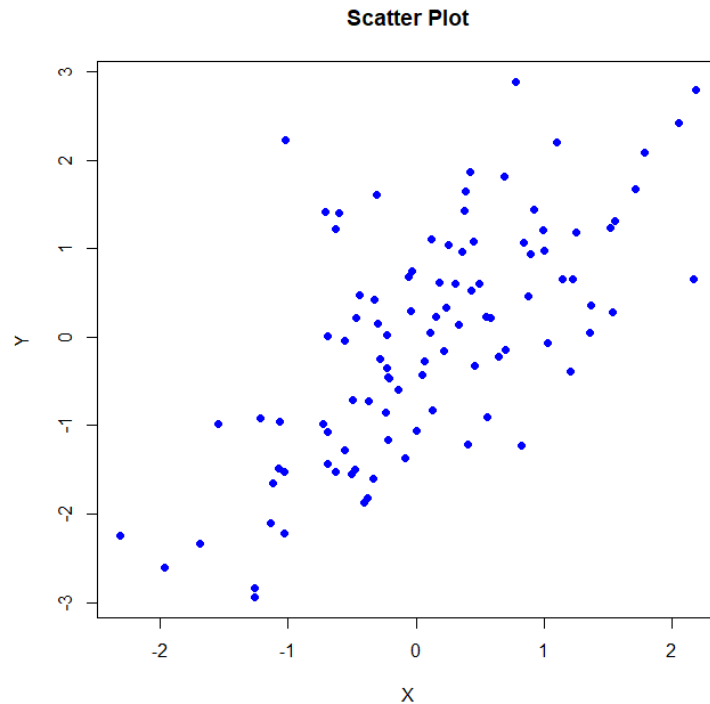


Figure 1.1: Scatterplot

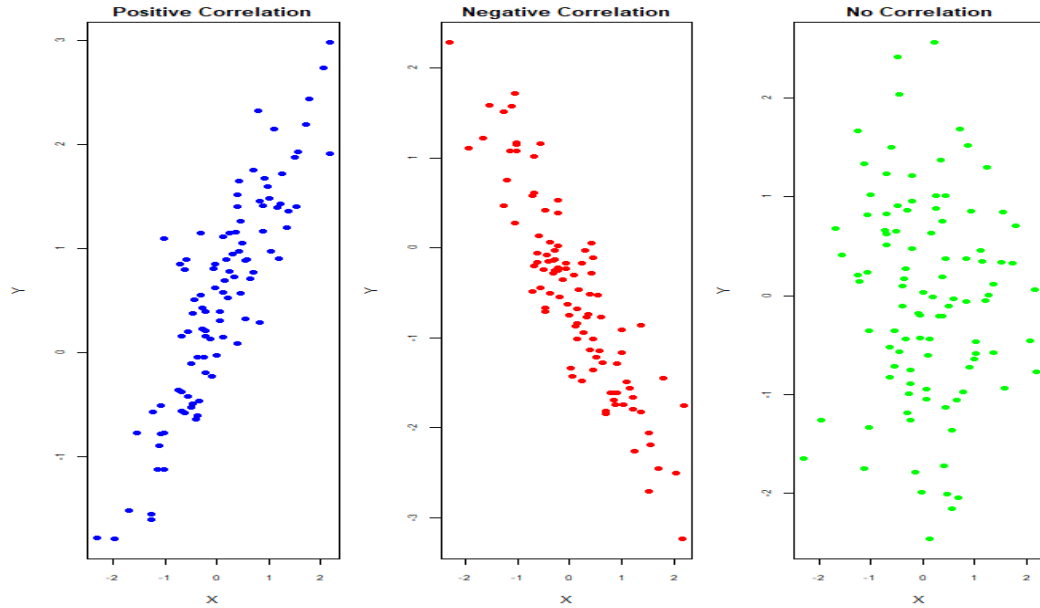


Figure 1.2: Correlation

3. Time-series:

A time series is a collection of data points organized chronologically. One major distinction between time series and other types of data sets is that they are indexed or listed in order of occurrence. Time would always be one of the axes in a graph constructed from the points of the time series. Time series data point graphs are commonly used to demonstrate trends or patterns in a more accessible and natural manner.

Time series analysis is technique of projecting future values based on past data points. Time series data analysis allows for the extraction of valuable statistics and other data characteristics. Economic data, temperature observations, and stock prices are just some of the examples.

Time series analysis focuses on achieving two main objectives:

- 1. Understanding the behavior of a time series:** This includes determining critical parameters and characteristics that accurately represent the behavior of the time series.
- 2. Forecasting future values of the time series:** Using historical and present data, this approach forecasts the time series' future values.

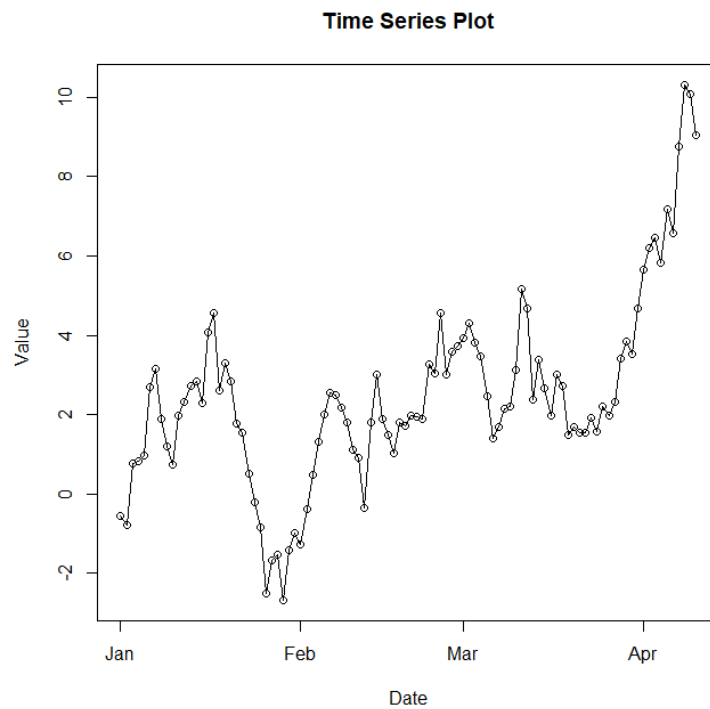


Figure 1.3: Time Series

4. Regression analysis:

Regression analysis is a statistical approach used in financial services, investment management, and other disciplines to determine the nature and strength of a relationship between a group of independent variables and one dependent variable,

usually denoted by Y .

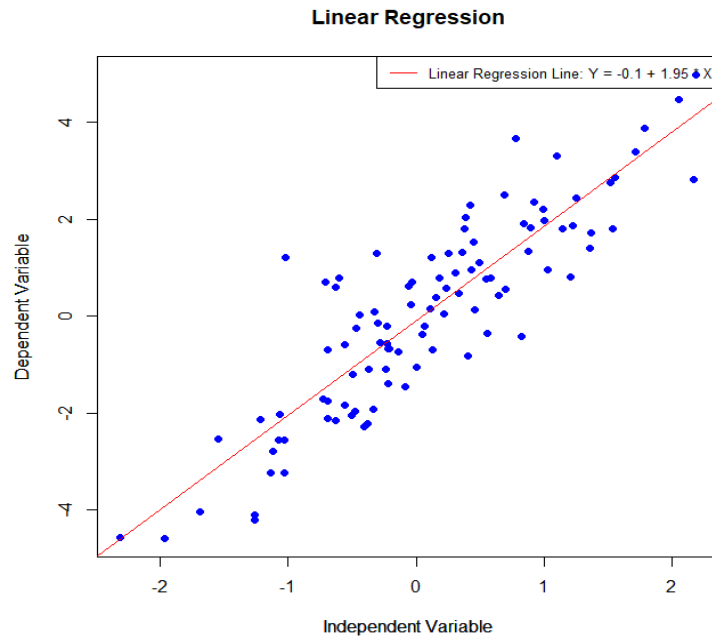


Figure 1.4: Linear Regression

The goal is to determine the extent to which variations in the independent variables correspond to modifications in the dependent variable. Linear regression is a widely used statistical tool for forecasting and prediction. There are two main classifications in linear regression: univariate and multivariate.

4.1 Univariate linear regression

Univariate linear regression is used to discover the association between a single independent and dependent variable.

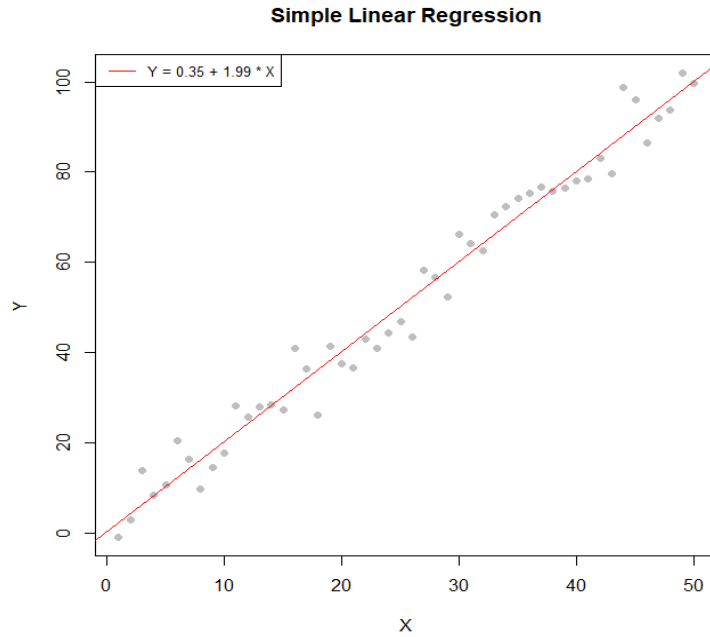


Figure 1.5: Simple Linear Regression

The model for simple regression represents this:

$$y = \alpha + \beta x + \varepsilon \quad (1.1)$$

It is implied that the covariable $\mathbf{x}$ is approximately a linear function of the dependent variable $\mathbf{y}$, where α is the independent term and β is the slope of the straight line. The error measures the degree of disagreement with this approximation because there is noise in the data or an incorrect relationship between the data and the statistical model. The ε represents the error in estimating the observed value $\mathbf{y}$ using the linear estimator obtained from the model.

4.2 Multivariate linear regression

Numerous linear regression models are employed when there are numerous independent or predictor variables. They are a generalization of simple linear regression. As such, the goal of this kind of multiple regression is to look into and quantify the relationship between one or more qualitative predictor elements and a dependent numerical variable.

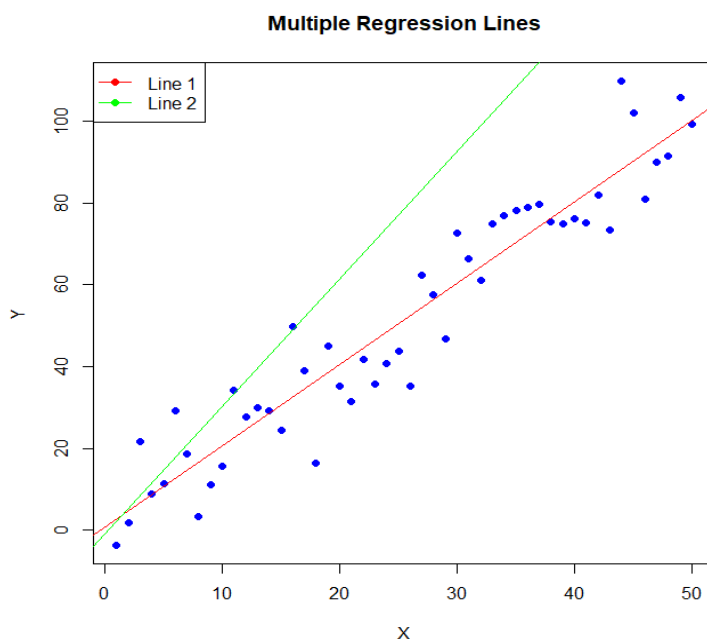


Figure 1.6: Multiple Linear Regression

The structure of a model with multiple linear regression is as follows:

$$y_i = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \dots + \beta_m x_{mi} + \varepsilon_i \quad (1.2)$$

The independent predictors are represented by the notation $x_1, x_2, \dots, x_m$,

whereas the dependent variable is represented by the notation y . The fixed numbers $\beta_1, \beta_2, \dots, \beta_m$ indicate the regression coefficients. The part of the model that cannot be explained by the independent variables is known as the error term, and it is represented by the symbol ε . This part of the model is thought to be representative of a normal distribution.

4.3 Regression Analysis's Least Squares Error:

The least squares approach is commonly used to estimate regression model coefficients by minimizing the sum of squared differences (residuals) between actual and predicted values. The difference between the regression model's predicted and actual values is assessed by the least squares error, often known as the sum of squared errors (SSE). The goal of minimizing this error metric is to determine the line (or curve) that best matches the data.

5. Clustering algorithms:

Data clustering is a common analytical technique that is gaining popularity in domains such as statistics, data mining, and pattern recognition. Data clustering is an exploratory as well as descriptive data analysis method. The cluster is generally defined by its internal cohesiveness (homogeneity) and external isolation (separation). As a result, the cluster is typically viewed as a collection of items that are similar within the cluster but distinct from objects in other clusters. Each data point is assigned to a single cluster using traditional partitioning and clustering techniques

such as K-Means and hierarchical approaches, which construct divided clusters [2]. In order to reduce the total mean square error throughout all K-regressions, the regression at each partition operates on a selection of data that is comparable [3]. Some data clustering methods are described below:

5.1 K-Means Clustering

The partition method identifies spherically formed groups that are mutually exclusive. It generates a specific number of disconnected clusters. Using the statistical approach, clustered categorical data can be grouped and ranked. Rank values were assigned to convert categorical data into numerical values. Using the K-Means approach, things are divided into k groups. Each partition corresponds to a cluster.

Analysts begin with the original set of means and categorize situations based on how far they are from their centers. After recalculating the cluster means with the cases assigned to the clusters, analysts classify each case using the revised options. Analysts will repeat this technique as long as the cluster means stays stable in the following phases. Ultimately, analysts calculate the cluster averages again and place the cases in their permanent clusters.

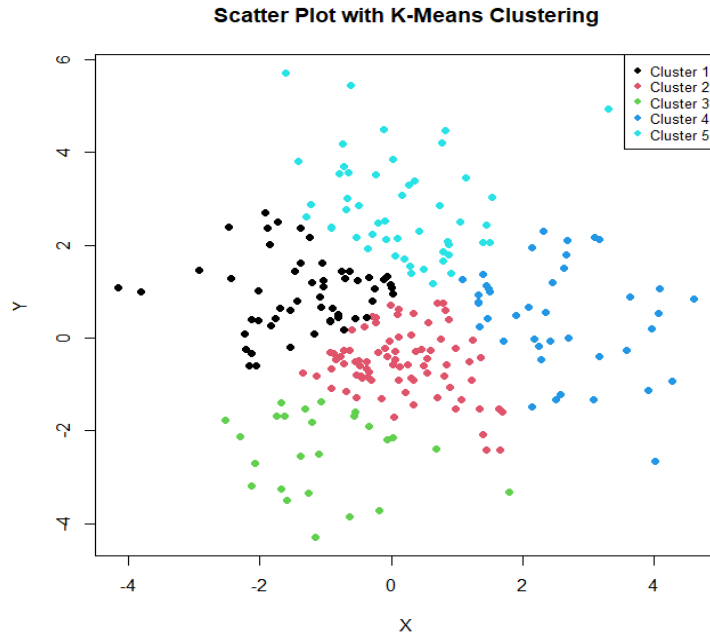


Figure 1.7: K-Means Clustering

5.2 Hierarchical Clustering

A hierarchical approach divides a given set of data items hierarchically. A tree of clusters generates a diagram. Sibling clusters divide the points covered by their common parent, and each cluster node contains child clusters.

When employing the hierarchical clustering method, researchers assign each item to a cluster, resulting in N clusters for any given number of items. Locate the nearest pair of clusters and unite them into a single cluster. Determine the distance between each prior cluster and the newly created cluster. Analysts must repeat these steps until every item has been categorized into K clusters.

6. Change Points Analysis:

A change point is any point in a dataset where the underlying distribution or pattern of the data changes noticeably or suddenly. Finding structural changes or irregularities in data required the identification of change points.

Consider a data series expressed as $f(x)$, where x signifies the index or time point inside the series, in the context of change points analysis. The change point problem, which is essentially a hypothesis-testing scenario, seeks to identify one or more change points in the dataset.

In mathematical terms, this means comparing the alternative hypothesis, H_1 , which asserts that one or more change points exist, to the null hypothesis, H_0 , which states that no change points exist. Formally, the alternative hypothesis posits that the data series undergoes a structural shift at one or more points, whereas the null hypothesis asserts that the data series has a single distribution throughout. Using this method for hypothesis testing, we can statistically evaluate change point evidence and get insight into changes or transitions in the underlying data process.

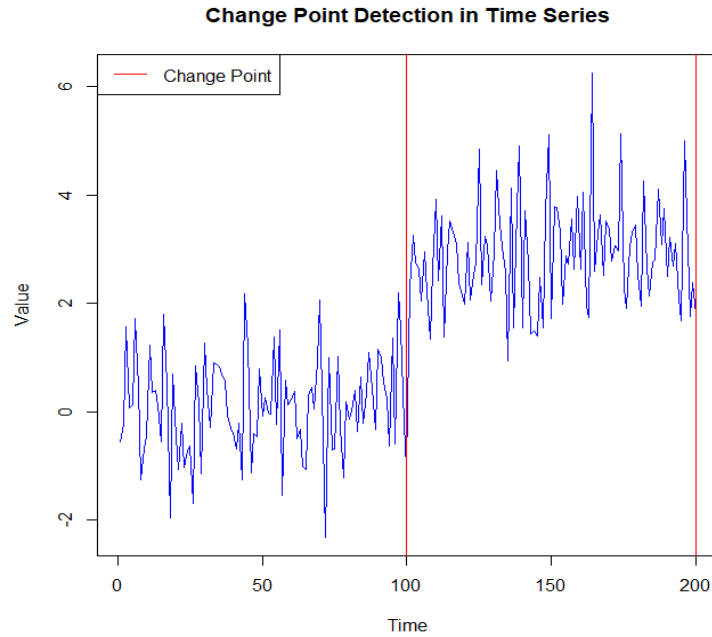


Figure 1.8: Change point

7. Trend Analysis:

In data analysis, a trend refers to the overall direction or tendency that a dataset exhibits over time. There are three types of trends: flat (indicating stability), downward (representing decrease), and upward (expressing increase). Trend analysis is a method of using historical data to predict the progressive changes of future events. Trends are estimated using a number of parametric and nonparametric approaches. Trend analysis procedures are statistical techniques for detecting an underlying behavioral pattern in a time series that may otherwise be masked by noise. As illustrated in the trend estimate, trend analysis can be performed within a typical regression analysis if the trend is assumed to be linear. Non-parametric techniques, such as the Mann-Kendall test, a variation of the Kendall ranking correlation coefficient, can be used to test trends with no linear structure [4].

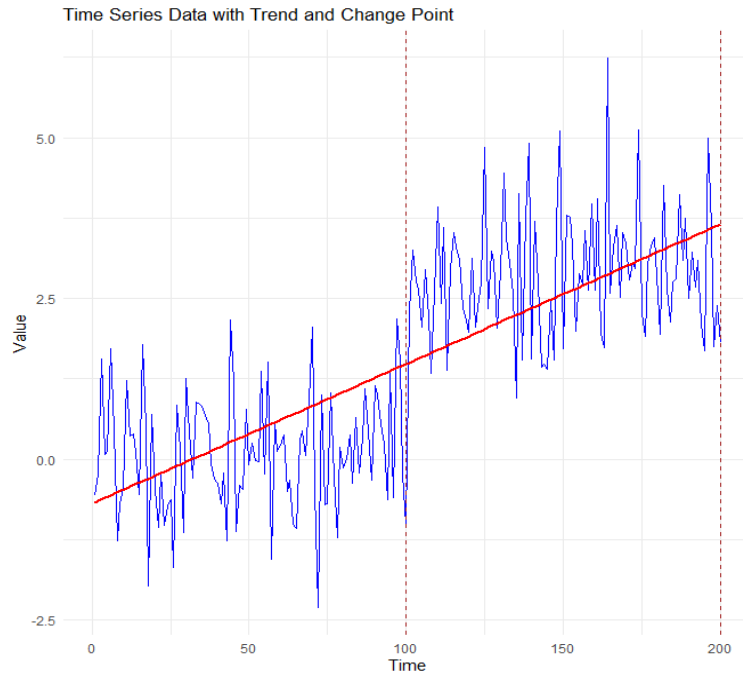


Figure 1.9: Trend Analysis

1.2.2 Overview of Techniques and Methods:

This section provides an important overview of the specific approaches and strategies that have influenced our study methodology. Every methodological choice is carefully studied in terms of its suitability for analyzing complex datasets and extracting meaningful patterns and insights. It allows us to address the specific challenges and objectives inherent in our research. In addition to increasing comprehension, the definitions and explanations included here provide critical background information for understanding how the methods are employed in our empirical investigations.

1. Optimization methods:

Optimization approaches employ a diverse set of algorithms and strategies to determine the best solution to a given problem. These strategies use iterative changes to variables or parameters inside a mathematical model to maximize or minimize an objective function while fulfilling constraints. To improve prediction models and gain meaningful insights from data, we use optimization methodologies in our study to reduce inaccuracy.

2. Minimizing Errors:

This refers to minimizing the disparity between the predictions generated by the model and the actual data. We can improve our models' accuracy and prediction ability by reducing their error measurements. Iteratively modifying model parameters to reduce error and enhance model fit is a common use for optimization techniques such as descent by gradient and least squares.

3. Definition of Norm:

A norm in mathematics is defined as a function that assigns a non-negative value to vectors indicating their size or magnitude. **Norms are key tools in machine learning and optimization** as they quantify the similarity or distance between vectors.

Common Norms:

- Euclidean Norm (L2 Norm): The Euclidean norm, denoted as $\|\mathbf{x}\|_2$, measures the distance between two points in Euclidean space. For a vector $\mathbf{x}$ in $\mathbb{R}^n$, the

Euclidean norm is defined as:

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$$

- **Manhattan Norm (L1 Norm):** The Manhattan norm, denoted as $\|\mathbf{x}\|_1$, measures the sum of the absolute differences between corresponding elements of vectors.

For a vector $\mathbf{x}$ in $\mathbb{R}^n$, the Manhattan norm is defined as:

$$\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$$

Example:

Consider a vector $\mathbf{v} = [3, -4, 5]$ in $\mathbb{R}^3$.

The Euclidean norm of $\mathbf{v}$ is calculated as:

$$\|\mathbf{v}\|_2 = \sqrt{3^2 + (-4)^2 + 5^2} = \sqrt{9 + 16 + 25} = \sqrt{50} \approx 7.07$$

The Manhattan norm of $\mathbf{v}$ is calculated as:

$$\|\mathbf{v}\|_1 = |3| + |-4| + |5| = 3 + 4 + 5 = 12$$

4. Introduction to Bootstrapping Method:

The bootstrap method is a powerful statistical approach that uses a sample of data to estimate population characteristics and features. This part provides a mathematical explanation of the bootstrap method and its application in estimating population parameters such as mean, variance, and correlation coefficient.

4.1 Basic Concepts

Let $X_1, X_2, \dots, X_n$ denote a random sample of size n drawn independently and identically distributed (i.i.d.) from a population with an unknown distribution function F . The empirical distribution function $\hat{F}(x)$ is a step function that assigns probability $\frac{1}{n}$ to each observation less than or equal to x . Mathematically, it is expressed as:

$$\hat{F}(x) = \frac{1}{n} \sum_{i=1}^n I(X_i \leq x)$$

where $I(\cdot)$ is the indicator function.

4.2 Resampling with Replacement

Bootstrapping is a method of generating multiple bootstrap samples $X_1^*, X_2^*, \dots, X_n^*$ by resampling with replacement from the original sample $X_1, X_2, \dots, X_n$, where n observations are randomly drawn from the original sample. Given an interest statistic T calculated based on observed data, the bootstrap estimator $\hat{\theta}$ of a population parameter θ_0 is defined as:

$$\hat{\theta} = \theta(\hat{F}) = \theta(\hat{F}(x))$$

where $\theta(\cdot)$ is a function representing the parameter of interest applied to the empirical distribution function $\hat{F}$.

Example: Estimating the Population Mean μ

The population mean $\mu = \int x dF(x)$ can be estimated using the sample mean $\bar{X}$, which is a functional of the empirical distribution function:

$$\hat{\mu} = \bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

4.3 Bootstrap Resampling Procedure

This method is particularly effective at estimating a statistic's sampling distribution by repeatedly taking thousands of replacement samples from the original dataset. This technique allows us to calculate the amount of uncertainty in our estimates and obtain strong confidence ranges for the parameters that are important.

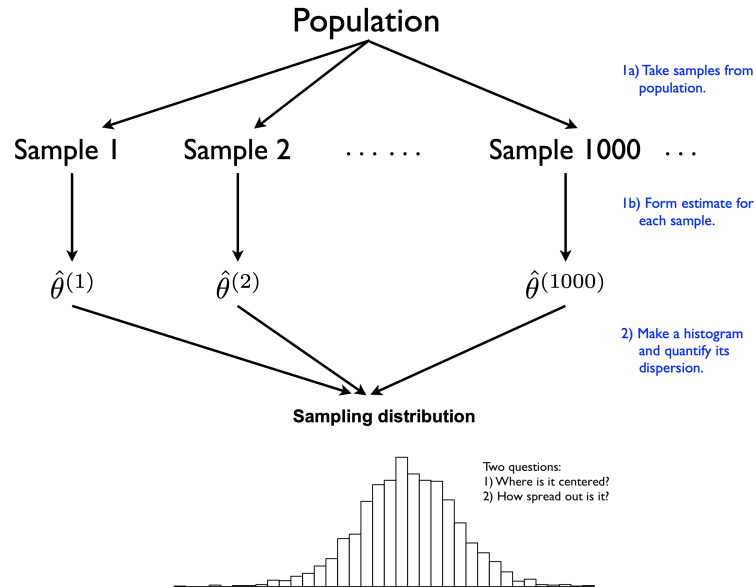


Figure 1.10: Bootstrap resampling

Source: <https://bookdown.org/jgscott/DSGI/the-bootstrap.html>

The procedure of bootstrap resampling:

1. **Original Dataset:** To begin, prepare a dataset with n observations, which are representative samples drawn from the target population.
2. **Sampling with Replacement:** Take n randomly selected observations (with replacement enabled) from the original dataset. In this method, observations are chosen from the dataset multiple times, with each observation having an equal probability of being selected in each draw.
3. **Repetition:** Repeat the sampling method several times, typically between hundreds and thousands of iterations. Each iteration generates an entirely new bootstrap sample, resulting in an extensive series of resampled datasets.
4. **Compute Statistic:** Determine which statistic is important for each bootstrap sample. A number of metrics, including the mean, median, and any other parameter

related to the study matter, can be included in this statistic. A statistical distribution is constructed by averaging these values over all bootstrap samples.

5. Estimate Sampling Distribution: To determine the statistic's sampling distribution, use the assembled set of computed statistics from the bootstrap samples. This distribution reflects the intrinsic variability of a statistic under repeated sampling from the original dataset.

6. Inference: Assume the population parameter based on the anticipated sample distribution. Confidence intervals, for example, can be created to provide a range with a certain level of confidence within which the actual parameter value is predicted to fall.

In this work, bootstrap resampling allows us to analyze the stability and reproducibility of our findings, especially in cases when more standard analytical methods may be insufficient due to data paucity or distributional presumptions. Bootstrap aggregating (also known as bagging) is a machine-learning ensemble strategy that employs bootstrap sampling. It improves the stability of machine learning systems and helps to avoid overfitting.

5. Kolmogorov-Smirnov (KS) test:

The Kolmogorov-Smirnov (KS) test is a non-parametric statistical test that compares an empirical cumulative distribution function (CDF) to the CDF of another sample or a theoretical probability distribution. It is notably useful in identifying whether two samples come from the same distribution or if a sample follows a certain distribution.

The test statistic, represented by D , the largest absolute difference between the sample's empirical and theoretical CDFs is calculated as:

$$D = \max_i |F_n(x_i) - F(x_i)|$$

where $F_n(x_i)$ is the sample's empirical CDF, $F(x_i)$ is the theoretical CDF, and (x_i) represent the sample's ordered values.

The distribution of the test statistics under the null hypothesis, D has a known distribution (either the two samples are from the same distribution or the sample follows the given distribution). Numerical approaches or statistical tables can be used to determine D 's critical values. In our research, we use the KS test to evaluate the goodness of fit between our data and other probability distributions.

1.3 Probability Distributions

Probability distributions serve as the foundation for statistical modeling and data analysis, providing a mathematical framework for describing the likelihood of different outcomes in stochastic events. These distributions enable probabilistic reasoning and inference by capturing the probabilities associated with various values or ranges of values for a given variable. We focus our investigation on eight popular probability distributions from a large number of distributions. Interestingly, we will use these distributions extensively in our subsequent section on applications.

- **Normal Distribution (Gaussian Distribution):** The symmetric bell-shaped curve of the normal distribution is one of its distinguishing features, described by the probability density function (PDF):

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

where μ represents the mean and σ^2 represents the variance. It is commonly employed to model data exhibiting uniformity and symmetry around a central mean.

- **Log-Normal Distribution:** The log-normal distribution is utilized to model positively skewed data whose logarithm follows a normal distribution. Its PDF is given by:

$$f(x|\mu, \sigma^2) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln(x) - \mu)^2}{2\sigma^2}\right)$$

where μ and σ^2 represent the mean and variance of the natural logarithm of x , respectively.

- **Exponential Distribution:** The Exponential distribution is characterized by its probability density function (PDF) given by:

$$f(x|\lambda) = \lambda e^{-\lambda x}, \quad x \geq 0$$

where λ is the rate parameter, controlling the rate of decay.

- **Gamma Distribution:** The Gamma distribution is characterized by its flexibility and is governed by the probability density function (PDF):

$$f(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$$

where α and β are the shape and scale parameters, respectively, and $\Gamma(\alpha)$ is the gamma function.

- **Beta Distribution:** The Beta distribution is characterized by its probability density function (PDF) given by:

$$f(x|\alpha, \beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}, \quad 0 \leq x \leq 1$$

where α and β are shape parameters, and $B(\alpha, \beta)$ is the Beta function.

- **Uniform Distribution:** The Uniform distribution is characterized by its con-

stant probability density function (PDF) over a specified interval:

$$f(x|a, b) = \frac{1}{b-a}, \quad a \leq x \leq b$$

where a and b are the lower and upper bounds of the interval, respectively.

- **Frechet Distribution:** The Frechet distribution, characterized by its heavy-tailed behavior, is described by the PDF:

$$f(x|\alpha, \beta, \gamma) = \frac{\alpha}{\beta} \left(\frac{x - \gamma}{\beta} \right)^{-\alpha-1} \exp \left[- \left(\frac{x - \gamma}{\beta} \right)^{-\alpha} \right]$$

where α , β , and γ represent the shape, scale, and location parameters, respectively.

- **Generalized Extreme Value (GEV) Distribution:** The GEV distribution is employed to model the distribution of extreme values and is characterized by its PDF:

$$f(x|\mu, \sigma, \xi) = \frac{1}{\sigma} \left[1 + \xi \left(\frac{x - \mu}{\sigma} \right) \right]^{-1/\xi-1} \exp \left[- \left(1 + \xi \left(\frac{x - \mu}{\sigma} \right) \right)^{-1/\xi} \right]$$

where μ , σ , and ξ represent the location, scale, and shape parameters, respectively.

1.4 Problem Statement

The need for a more comprehensive and adaptable method has been highlighted by the constraints that arise from using time series analysis alone to track data patterns. When working with high-dimensional and multidimensional data, it can be challenging to capture intricate linkages and changes within a dataset if we only rely on time series data. In response to these limitations, there is a growing trend toward the tracking of trends inside a scatter plot.

To address the limitations of existing approaches, particularly with regard to time series data, the problem is determining the optimal data partitioning strategy for scatter plots, which has a substantial influence on the effectiveness and quality of subsequent analyses. In this thesis, we want to create a method that overcomes these constraints and may be used with larger data sets. Through this, we aim to improve data analysis's efficacy and extract valuable information from various statistical data sets.

1.5 Objectives

The objectives of the thesis can be structured as follows:

1. Survey of Existing Methods:

Conduct a comprehensive evaluation of existing methods constructed particularly for time series data partitioning, assessing their strengths and weaknesses.

2. Introduce Our Method for Optimal Data Partitioning:

Describe our novel method for determining the best partition of data sets, discussing its existence and practical implications, evaluating the uniqueness of partitioning solutions, and assessing its sensitivity to the chosen optimization norm.

3. Statistics of the Optimal Points:

Investigate the joint distribution of the optimal points $(a_1, \dots, a_n)$ acquired by our method. Examine the individual elements' marginal distributions within the optimal points to gain insight into their properties.

1.6 Thesis Structure

The Introduction **Chapter 1** of the thesis begins by emphasizing the significance of real-time data tracking and change-point analysis in machine learning, statistics, and prediction. It outlines objectives and a roadmap for upcoming chapters and highlights the importance of flexible frameworks to manage incomplete datasets and real-time data flow. In addition, certain fundamental concepts that are essential for understanding the concepts and methods discussed in subsequent chapters are covered in this chapter.

Chapter 2 integrates an in-depth review of the literature with an analysis of current approaches and difficulties. It looks at methods, algorithms, and statistical approaches for tracking trends and identifying changing points in a wide range of data forms.

The Optimal Partition Identification Method is presented in **Chapter 3**. The theory, mathematical foundation, and essential concepts for determining the most optimal

partitions are discussed. It also provides a thorough theoretical investigation of the approach, combining theory to statistical propriety by examining aspects such as the existence and uniqueness of optimal partition locations.

In Chapter 4, the proposed real-time tracking methodology is described in depth. Its dynamic framework for tracking continually changing datasets is demonstrated, along with its ability to adapt to changing data distributions and patterns.

Chapter 5, In order to validate the proposed approaches, this chapter combines comparative analysis with empirical validation. It uses real-world datasets to apply these approaches and demonstrates how much better they're performing than current methods. It also performs statistical analysis to interpret the results.

Chapter 6 discusses employing Deep Neural Networks (DNNs) in regression analysis, including its components, structure, and activation functions. It also discusses a real-world numerical example that uses the same dataset as previous methods, enabling an evaluation of results.

Chapter 7 summarises the thesis's results, theoretical contributions, practical implications, and recommendations for future study.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we provide a comprehensive review of the literature on change-point analysis and pattern tracking in data. By combining previous research and discussing concepts and theories, we aim to provide an understanding of the current state of knowledge in this field.

The literature can be classified into five categories:

- . Background on Change Point Detection.
- . Change-Point Analysis Methodologies.
- . Pattern Tracking Techniques.
- . Applications in Different Domains.
- . Challenges and Limitations.

2.1 Background on Change Point Detection

The problem of change point identification is attracting a lot of attention from the academic community. In recent years, there has been a significant increase in interest in change point detection, which has applications in a variety of fields, including technology and science, finance, economy, climate science, and health [5],[6]. Since the 1950s, researchers have been interested in identifying change points in statistical analysis [7].

The "change-point problem" refers to the process of identifying areas in a dataset where the underlying distribution or data behavior changes abruptly and dramatically. To put it another way, it involves identifying the points in the data series where a major shift occurs, causing the pattern or trend to diverge.

The following is a formal definition of the change-point problem:

Given a sequence of observations or data points $X = \{x_1, x_2, \dots, x_n\}$, the objective is to determine the locations $t_1, t_2, \dots, t_k$ where the data distribution or statistical properties undergo a distinct change. These change points divide the dataset into segments or regimes, each distinguished by distinct statistical characteristics or behaviors.

The most difficult part of the change-point problem is distinguishing real shifts from noisy or random fluctuations in the data. To overcome this challenge, a range of statistical techniques, machine learning algorithms, and optimization tactics are utilized to pinpoint change points precisely.

Change point detection methods may generally be divided into two categories: online and offline [8]. The offline changing point approach includes approaches that allow for the study of the entire dataset. In contrast, the online change point framework does analysis step by step as new data becomes available. More subsequent studies deal with more complex circumstances, encompassing both offline and online change point detection issues [9].

The objective of change point analysis is to identify one or more statistically significant locations where changes occur by fitting count data with a number of methods. It is possible to identify an appropriate number of change points for estimation. According to Chen and Gupta’s research [10] there are two significant challenges in statistical inference regarding change points. The first challenge is to detect the presence of a change, and the second is to figure out the number and location of these change points.

2.2 Change-Point Analysis Methodologies

In the review of the existing methods section, several approaches that are relevant to change point and trend methods that are applied depending on the type of data have been examined [7]. The initial change point detection research, reaching back to the 1950s, aims to find a shift in the average of independent and identically distributed (iid) data for industrial applications [7]. In 1954, a frequentist framework was created to address change-point issues. In this research [7], he proposed an online framework for identifying changepoints in time series parameters. Recommend utilizing moving

averages to resemble the mean of the time series and hypothesis testing to evaluate if a CP occurred. In the initial research of CPD, frequentist methods were used to estimate the abundance of CPs across time. Yao (1988) [11] and Lee (1995) [12] proposed employing a penalized least-squares estimate to determine the number of CPs in a time series.

Parametric change point

Parametric change points are common techniques that assume prior knowledge of the distribution pattern before and after the change point and use fixed unidentified parameters. The likelihood function considerably aids the identification of change points in diverse scenarios. Chen and Gupta provide an overview of identifying points of change in normal systems via a range of methods, including the Bayes method and the likelihood ratio approach [13].

The Gaussian, gamma, exponential, and other continuous distributions belong to the exponential family. Using a likelihood ratio test, Siegmund established confidence intervals for a variation point of variables that were independent of an exponential family [14]. In order to identify change points in log-linear structures, [15] employed likelihood ratio testing. These findings were used to determine the test's power and range of confidence for the change point. O'Brien used an optimal categorization technique to determine the number and location of change points. This divided the continuous variable of explanation into distinct parts to clarify the response amount from the distribution of exponential families [16].

Similar methods can be used to identify changes in the parameter values of discrete distributions (such as Poisson and binomial). The simplest scenario in change point identification is identifying a single changing point. The most common technique is the likelihood ratio.

- **Maximum likelihood estimation**

There are numerous maximum likelihood approaches in the literature on change point detection. Hinkley (1970) first proposed the likelihood ratio test for change point models [17]. Detecting single change points is a hypothesis testing that compares the null hypothesis (no changing points) to the alternative (a single changing point).

The likelihood ratio test was used to detect change points in data under various circumstances. Hinkley (1970) applied this method to normally distributed data [17]. Haccou (1987) applied the test to exponentially distributed data to find changes in a rate parameter [18]. Kim and Siegmund (1989) applied the test to regression analysis to find changes in the slope and intercept [19].

They generalize a variety of cost functions and models, such as scaling and mean shifts in normally distributed data [20]. As a rule, the signal that has been observed $y = \{y_1, \dots, y_T\}$ is made up of independent random variables when using the estimation of maximum likelihood for change detection such as

$$y_t \sim \sum_{i=0}^i f(\cdot|\theta_i) \quad (t_i < t < t_{i+1}) \quad (2.1)$$

Change point indexes are represented by t_i , whereas probability density functions are represented by $f(\cdot|\theta_i)$. In this case, if the entire cost equals the negative log-likelihood, change point tracking is equivalent to the estimation of maximum probability [21]. The cost function for a parametric distribution family is defined as

$$c_{\text{i.i.d}}(y_{a..b}) = -\sup \sum_{t=a+1}^b \log f(y_t|\theta) \quad (2.2)$$

- **Piecewise linear regression**

A linear relationship is present between a response variable and covariates, which might vary rapidly at unexpected periods. The determined y signal follows a piecewise linear model with change points at t_i .

$$\forall t, \quad t_i < t < t_{i+1}, \quad y_t = x_t u_i + z_t v + \varepsilon_i \quad (2.3)$$

To identify change points, use a linear regression to each segment of data. This is accomplished by ensuring that the total cost and squared residuals are equal. The cost function, c_{linear} , is defined as follows

$$c_{\text{linear}}(y_{a..b}) = \min \sum_{t=a+1}^b (y_t - x_t u + z_t v)^2 \quad (2.4)$$

Bayesian Detection Methods

A Bayesian technique includes prior knowledge about the position of the change point in the model by considering it. A number of researchers used Bayesian approaches to study change points in a variety of scenarios.

Cherno and Zacks (1964), Broemeling (1972, 1974), and Smith (1975) proposed early Bayesian approaches to the single change-point problem. The researchers examine cases when the distributional kinds of segments prior to and after the transition point are unknown or just one has been determined [22],[23],[24]. Barry and Hartigan's (1992) study on Product Partition Models (PPMs) was especially notable since it served as a foundation for subsequent groundbreaking advances in CPD [25].

Diniz applied Bayesian estimation to fit a segmentation linear regression model with non-homogeneous error variance [26]. Tiwari developed the Bayesian method selecting techniques for estimating the quantity and position of change points [27]. In order to discover points of change in Bernoulli data, Steward used segmented regression using a Bayesian [28].

Non-parametric change point

Non-parametric change point studies consider the case when prior distributional knowledge fails to exist, i.e., when there is no prior knowledge of the data or data following the change. It is the most useful for a variety of reasons, particularly when dealing with experimental data. Orv'ath and Kokoszka examined both the right and

left continuations of the non-linear function to detect change points in non-parametric regression methods [29]. [30] has developed a non-parametric empirical likelihood ratio (ELR) approach to estimate change points using binary data. Matteson and James examined non-parametric methods to determine the number of change points and where they are found in multivariate data that has been grouped [31]. One crucial approach is covered here, which is kernel estimation.

Kernel-based detection

Harchaoui and Cappe have proposed a kernel-based method for carrying out change point identification in a non-parametric setting [32]. The original signal, y , is assigned to reproduce the Hilbert space in order to accomplish this. This is the definition of the corresponding cost function, or c_{kernel} .

$$c_{\text{kernel}}(y_{a..b}) = \sum_{t=a+1}^b \|\phi(y_t) - \bar{\mu}_{ab}\|_H^2 \quad (2.5)$$

Regression analysis change point

Data patterns can change at random times. These change points split the data into numerous segments, each with its own model. Several papers have conducted in-depth studies on the change point problem associated with single linear regression models. In order to estimate the joinpoints between two segments of the linear regression models, Blischke assumed a continuous join point and used least squares estimators [33]. In multiple linear regression models, there might be one or more identified or unknown change points. In various studies related to the multiple linear regression

model, Ferreira used a Bayesian technique to analyze variance in a regression model with a known number of regimes [34].

- **Segmented linear regression analysis**

In order to find one or more statistically significant modification points, segment analysis fits data points with different linear models. As shown in Figure 2.1 [35], when using segmented regression, a new segment will start at the exact moment the previous segment ends. This is due to the fact that possible trends are fixed during specific intervention times. Ansari claims that split linear regression is an effective and reliable method for determining how interventions affect alterations [36]. Zhang employed bootstrapping and delta approaches to perform segmentation regression on time series datasets and compute confidence intervals for the relative changes in the output variable [37].

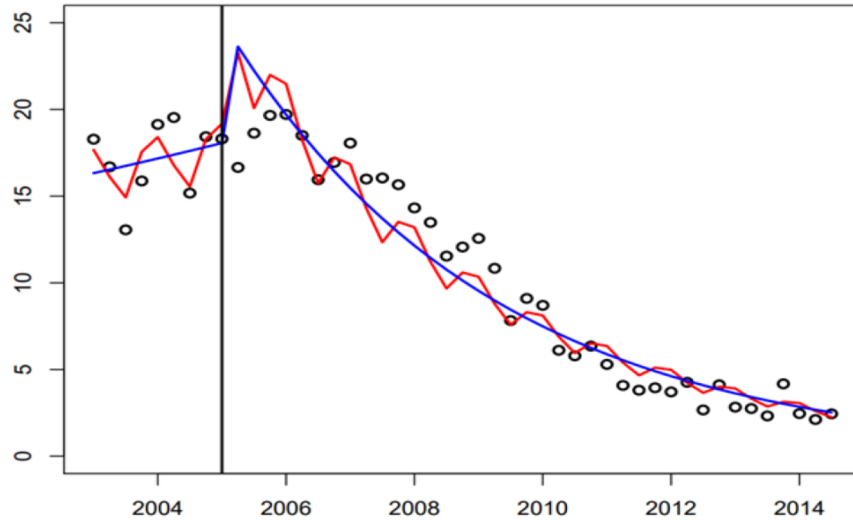


Figure 2.1: Segmented regression model

- **Change point in spline regression models**

Jiang calculated the knot spots and then found the deterioration change points using a spline function [38]. The cubic regression spline was employed by Curtis and Simpson to model ecological data from time series [39].

We go beyond single-change point detection to multi-change point detection. Ideally, we should assess every conceivable combination of change points and their locations. However, the process requires a great deal of processing power. Two methods that are efficient in terms of computation have been developed: **Binary segments** and **Dynamic programming** are discussed.

Binary segments

The Binary Segmentation (BS) technique was initially introduced by Scott and Knott (1974) and Vostrikova (1981) [40],[41]. It is used to indicate different changing points in a range of works. Starting the general binary segmentation process entails defining the interval boundaries based on a test statistic that depends on the time series that might be a change point and a threshold. Ideal threshold is determined by the length of the time series and the particular change point test statistic. The point of change test is applied for both subseries preceding and succeeding the change point. Continue doing this until you find no more change points. Change point identification is carried out in a stepwise manner by BS, which significantly reduces computing complexity.

Dynamical programming

Dynamic programming is a different approach that is often used in multiple changing point detection. By solving an optimization problem for segments that are based on a chosen cost function for each segment, the objective is to minimize the overall cost. The residual sum of squares and negative log-likelihood are two preferred options for cost functions. To estimate multiple change points, Hawkins (2001) and Maboudou-Tchao and Hawkins (2013) propose employing dynamic programming with likelihood-based segmentation [42], [43].

2.3 Pattern Tracking Techniques

In data analysis, the ability to recognize patterns is crucial to understanding the situation and making wise choices. In this section, we review popular techniques for pattern recognition, with a focus on machine learning algorithms such as recurrent neural networks (RNNs) and hidden Markov models (HMMs).

Machine learning algorithms have gained popularity recently because of their ability to quickly and accurately detect complex patterns in sequential data. Of these, recurrent neural networks (RNNs) have emerged as a powerful tool for temporal data processing, enabling the modeling of dependencies and interactions over time steps. Similar to this, hidden Markov models (HMMs) offer a probabilistic framework for capturing latent states and transitions inside sequential data, making them well-suited for tasks like sequence labeling and anomaly detection.

2.4 Applications in Different Domains

Applications for change point detection include monitoring medical conditions, detecting climate change, and analyzing human behavior. When modeling and forecasting time series, it is useful. Data mining, statistics, and computer science all have been examined in CPD within the past few decades. This problem covers many real-world concerns. Here are a few inspiring instances:

- **Monitoring of medical conditions:**

Trend detection in physiological variables such as heart rate, electrocardiogram, and electroencephalogram is required for continuous patient health monitoring. Studies investigate change point detection for particular medical diseases, including epilepsy, problems with sleep, MRI interpretation, and comprehension of brain activity [44]. CPD appears to be useful for independent patient monitoring in medical applications. Patients may get real-time supervision by monitoring physiological parameters such as heart rate and ECG [45].

- **Climate change identification:**

In recent decades, the possibility of climate change and the increase in greenhouse gas concentrations in the atmosphere have made change point detection-based methods for studying, monitoring, and predicting climate change more significant [44].

- **Analysis of human activity:**

Change point identification is the process of determining activity transitions or break-

points using characteristics of observed sensor data from smart homes or mobile devices. Utilizing these transition points can help with activity classification, minimizing social interaction, providing activity-aware services, and spotting behavioral shifts that could indicate a person’s health [44].

2.5 Challenges and Limitations

Change-point analysis and pattern-tracking approaches have potential, but there are a number of difficulties and restrictions [46], especially when managing different kinds of data structures and working with scatter plot data. A significant drawback is that the majority of research to date has been done on time series data, which may not adequately represent the intricate dynamics of data found in scatter plot data. Traditional change-point analysis techniques designed for time series data are challenged by the diverse patterns and structures that scatter plot data frequently display, such as non-linear correlations and heterogeneous distributions.

Furthermore, because fluctuations in data density and distribution can have a substantial impact on the identification of change points, the assumption of homogeneity in data distribution over time may not hold in the case of data in scatter plots. Furthermore, there is a vast range in the length and type of data sequences, from brief, sparse observations to lengthy, heavily sampled datasets. Change-point analysis techniques become more complex due to data variability, necessitating flexible and reliable methodologies that can handle various data structures.

CHAPTER 3

PROPOSED METHODOLOGY: CHANGE POINT ANALYSIS

3.1 Introduction

In the field of complex scatter plots, where the relationship between variables is not a standard function and displays clusters with distinct tendencies, accurate prediction, and meaningful insights are hard to come by. Imagine an instance where (X, Y) naturally clusters, making it impervious to a global regression model. Let's examine the following example in order to assist in illustrating the idea.

The first figure (Figure 3.1) effectively illustrates the intricate structure of the data by highlighting the differences in trends and patterns across data points. The visual representation serves as a concrete illustration of the complexities that we aim to address. It draws attention to the necessity of an elaborate strategy and provides us with the opportunity to look into the optimal segmentation.

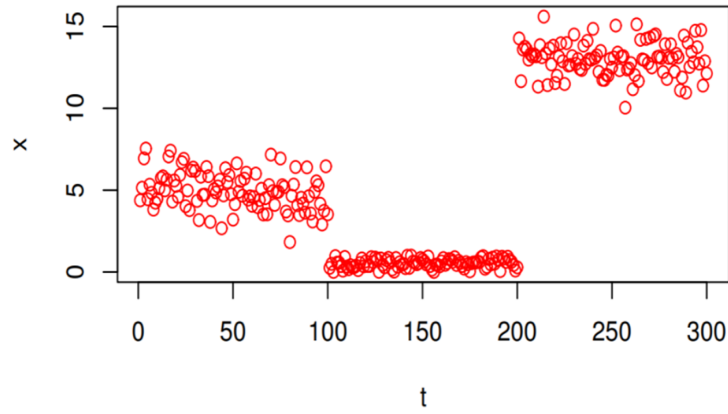


Figure 3.1: Multimodal scatter plot.

Finding the underlying structure in the data is the aim, and it should be noted that a one-size-fits-all regression approach may overlook certain important deviations. By carefully partitioning the scatter plot into clusters, we may more effectively address local trends and patterns.

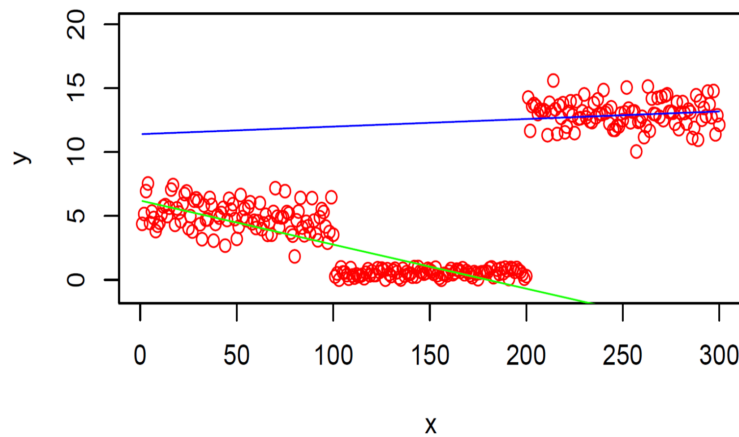


Figure 3.2: The point $x = 200$ is the best cut-off point.

It is clear from the examination that two potential cut-off positions, 100 and 200, have

been proposed as shown in Figure 3.2. **However, using the methods discussed later in this chapter, it becomes clear that the optimal cut-off point is 200 .** This decision is based on our methodology, which promotes error minimization. A turning point in our work is indicated by this graphic signal, which tells us to use linear regression inside specific clusters in a customized and efficient manner.

So, the question is: Which partitioning strategy would maximize the performance of linear regression inside each cluster? Beyond simply division, the ideal partitioning seeks to achieve a balance between consistency and precision.

The objective is to identify the clusters where linear regression produces the best representation, with each subset significantly enhancing the overall predictive power.

We must first find hidden structures in the data and then try to understand the intricate relationships that are there in it in order to properly employ linear regression in a locally effective way.

Our research aims to, in essence, identify the partitions that allow predictive modeling to perform to its best ability when faced with complex, clustered scatter plots.

3.2 The Method

Before diving into our data tracking method, we must investigate critical metrics that improve our understanding of the approach.

3.2.1 Measurement of Error using Euclidean Distance

In order to measure the discrepancy between actual $\mathbf{y}$ and predicted values $\mathbf{f}(\mathbf{x})$, we used the Euclidean distance metric in our study. The difference between corresponding vector elements is measured by the Euclidean distance metric, which sums squared differences. The following formula is used to express this metric:

$$\text{Euclidean Distance} = \sum_j |y_j - f(x_j)|^2$$

This formula provides a complete assessment of the overall error by encapsulating the squared discrepancies between the observed and predicted values. We are able to assess our predictions' accuracy throughout the dataset by applying the Euclidean distance.

Remark 1 For $p \geq 1$, we can consider $\sum_j |y_j - f(x_j)|^p$. However, $p = 2$ is better when seeking minima (differentiability)

3.2.2 Tracking the change of the patterns using optimal partitions

Consider the data collection (X, Y) , where $X = (x_i)$ and $Y = (y_i)$ (with the indexation set omitted for simplicity). Let $\mathcal{F}$ denote a family of real-valued functions, referred to as approximators. For a fixed value of n , let $\Gamma_n = \{(a_1, \dots, a_n) \in X^n \mid \min X < a_1 < \dots < a_n < \max X\}$. The objective is to delve into the minimum of the function

defined on Γ_n as follows:

$$\begin{aligned} \Gamma_n &\longrightarrow \mathbb{R}_+ \\ \varphi_n(a_1, \dots, a_n) &\longmapsto \sum_{k=1}^{n+1} \inf_{f \in \mathcal{F}(X_{|I_k})} \Phi_p(Y_{|I_k}, f) \end{aligned}$$

Here, Γ_n represents the set of n -tuples within X satisfying the condition $\min X < a_1 < \dots < a_n < \max X$. The function φ_n maps these n -tuples to non-negative real numbers by considering the sum of infima over certain subsets defined by I_k . The infimum is taken over functions f belonging to the family $\mathcal{F}$, specifically those defined on the restricted domain $X_{|I_k}$. This investigation aims to unravel the characteristics and properties associated with the minimum values of φ_n , shedding light on the interplay between the data structure and the approximating functions within the family $\mathcal{F}$. Where $I_k = \{j \mid x_j \in [a_{k-1}, a_k[\}$, with conventions $a_0 = \min X$ and $a_{n+1} = \max X$. Let $\mathcal{F}(X_{|I_k})$ represent the set of restrictions of functions in $\mathcal{F}$ to $X_{|I_k}$. Additionally, define $\Phi_p(Y_{|I_k}, f) = \sum_{j \in I_k} |y_j - f(x_j)|^p$ for $p \geq 1$.

In essence, the aim is to identify values $a_1, \dots, a_{n-1}$ for which the function $\varphi_p(a_1, \dots, a_{n-1})$ is minimized. This involves considering the sum of infimum over the subsets defined by I_k , where the infimum is taken over functions f in $\mathcal{F}(X_{|I_k})$. The term $\Phi_p(Y_{|I_k}, f)$ quantifies the discrepancy between the observed data and the approximation provided by f for the elements in I_k . The investigation aims to reveal the optimal configuration of $a_1, \dots, a_{n-1}$ that minimizes the overall functional expression φ_p .

3.2.3 Example

In this part, we use a particular dataset and a family of approximation functions to demonstrate how to use our method. The dataset consists of a time interval $[0, T]$ and is represented by the function $f = 1_{\{0,1\}}$, where $1_{\{0,1\}}$ is an indicator function equal to 1 on the set $\{0, 1\}$.

We analyze the function $\varphi(a)$ in our framework, which is the total of the distances between the family of constant functions $\mathcal{F}$ and the observed function f . Constant functions, which are functions with no variation throughout their domain, comprise the family $\mathcal{F}$.

To demonstrate the application of our framework, consider the dataset $[0, T] = [0, 1, 2, 3]$ and the function $f = 1_{\{0,1\}}$. For $n = 1$, We have

$$\varphi(a) = \text{dist}(f|_{\{0,\dots,a\}}, \mathcal{F}) + \text{dist}(f|_{\{a+1,\dots,3\}}, \mathcal{F})$$

The framework yields $\varphi(1) = 0$ and $\varphi(2) = (1 - \frac{2}{3})^2 + (1 - \frac{2}{3})^2 + (0 - \frac{2}{3})^2 + 0^2$.

So 1 is the point where we break down the $[0, T]$. This demonstrates the effectiveness of our framework in detecting meaningful patterns and transitions in the given dataset. Finally, the example that was provided demonstrates how our method may be used in practice to analyze and identify transitions in time-dependent datasets. This approach offers insightful information about how the observed functions behave within the given approximation function family.

In the context of our analysis, we establish an important property regarding the construction of $\Phi_p(Y_I, f)$, which will play a crucial role in the subsequent discussions.

Lemma 1 *The construction of $\Phi_p(Y_{|I}, f)$ is additive. In other words, if $I \cap J = \emptyset$, then*

$$\Phi_p(Y_{|I}, f) + \Phi_p(Y_{|J}, f) = \Phi_p(Y_{|I \cup J}, f).$$

Proof.

Given $I \cap J = \emptyset$.

$$\text{We have } \Phi_p(Y_{|I}, f) = \sum_{i \in I} |y_i - f(x_i)|^p$$

$$\text{and } \Phi_p(Y_{|J}, f) = \sum_{j \in J} |y_j - f(x_j)|^p.$$

$$\text{Then, } \Phi_p(Y_{|I}, f) + \Phi_p(Y_{|J}, f) = \sum_{i \in I} |y_i - f(x_i)|^p + \sum_{j \in J} |y_j - f(x_j)|^p.$$

Since $I \cap J = \emptyset$, we can combine the sums:

$$= \sum_{i \in I \cup J} |y_i - f(x_i)|^p = \Phi_p(Y_{|I \cup J}, f)$$

I

Therefore, the equality holds, confirming the additivity property:

$$\Phi_p(Y_{|I}, f) + \Phi_p(Y_{|J}, f) = \Phi_p(Y_{|I \cup J}, f).$$

The theoretical foundation for Error Reduction and Optimization

Proposition 3.1 *The accuracy of function approximations has been shown to increase with an increase in a single cutoff point using empirical validation. This improvement is supported by the following proof, which defines the function $\varphi(a)$ as follows:*

Proof.

$$\begin{aligned}
\varphi(a) &= \sum_{i=1}^a \inf_{f \in \mathcal{F}_{|I}} \|y - f(x)\|_2^2 + \sum_{i=1+a}^N \inf_{f \in \mathcal{F}_{|J}} \|y - f(x)\|_2^2 \\
&= \sum_{i=1}^a (y - f_1(x))^2 + \sum_{i=1+a}^N (y - f_2(x))^2 \quad (\text{at some } f_1, f_2) \\
&\leq \sum_{i=1}^a (y - f_{|I}(x))^2 + \sum_{i=1+a}^N (y - f_{|J}(x))^2 = \sum_{i=1}^N (y - f(x))^2 \quad \forall f
\end{aligned}$$

Applying inf for both sides,

$$\therefore \varphi(a) \leq \inf_f \sum_{i=1}^N (y - f(x))^2$$

■

It is proved that fine-tuning the function approximation, particularly modifying a single cutoff point, improves accuracy and reduces the error rate when compared to the global error.

Remark 2 *One can now understand why the definition of $\Phi_p(Y_{|I}, f)$ is given by $\sum_{j \in I} |y_j - f(x_j)|^p$ rather than $\sqrt[p]{\sum_{j \in I} |y_j - f(x_j)|^p}$. The choice of the former form is intentional to make the previous proposition work.*

When combining contributions from various sets, the additivity feature plays a crucial role. Examine the later amount: $\sqrt[p]{\sum_{j \in I} |y_j - f(x_j)|^p}$. Unfortunately, this form does not preserve the desired property to get inequality i.e

$$\sqrt[p]{\sum_{j \in I} |y_j - f(x_j)|^p} + \sqrt[p]{\sum_{j \in J} |y_j - f(x_j)|^p} \not\leq \sqrt[p]{\sum_{j \in I \cup J} |y_j - f(x_j)|^p}.$$

This characteristic makes it easier to analyze data in a straightforward way, which is consistent with our general methodology for comprehending and modifying error terms within the framework of the approximation process.

Theorem 1 *The infimum of $\varphi_n(a_1, \dots, a_n)$ is non increasing in terms of n .*

We find an important theorem that explains the relationship, in our investigation for function approximation accuracy, between the number of parameters, n , and the infimum of $\varphi_n(a_1, \dots, a_n)$. This theorem gives a clear explanation of the intricate dynamics of error functions and offers significant insights into how the complexity of our models influences minimal error.

Proof. We have

$$\begin{aligned}
\varphi_{n+1}(a_1, \dots, a_{n+1}) &= \sum_{k=1}^{n+2} \inf_{f \in \mathcal{F}(X_{|I_k})} \Phi_p(Y_{|I_k}, f) \\
&= \inf_{f \in \mathcal{F}(X_{|I_1})} \Phi_p(Y_{|I_1}, f) + \inf_{f \in \mathcal{F}(X_{|I_2})} \Phi_p(Y_{|I_2}, f) + \sum_{k=3}^{n+2} \inf_{f \in \mathcal{F}(X_{|I_k})} \Phi_p(Y_{|I_k}, f) \\
&\leq \Phi_p(Y_{|I_1}, f_1) + \Phi_p(Y_{|I_2}, f_1) + \sum_{k=3}^{n+2} \Phi_p(Y_{|I_k}, f_k) \\
&= \Phi_p(Y_{|I_1 \cup I_2}, f_1) + \sum_{k=3}^{n+2} \Phi_p(Y_{|I_k}, f_k) \\
&= \Phi_p(Y_{|J_1}, f_1) + \sum_{k=2}^{n+1} \Phi_p(Y_{|J_k}, f_k). \\
&= \sum_{k=1}^{n+1} \Phi_p(Y_{|J_k}, f_k)
\end{aligned}$$

Therefore

$$\varphi_{n+1}(a_1, \dots, a_{n+1}) \leq \sum_{k=1}^{n+1} \inf_{f \in \mathcal{F}(X_{|J_k})} \Phi_p(Y_{|J_k}, f) = \varphi_n(b_1, \dots, b_n)$$

and by applying the inf on both sides we get $\inf \varphi_{n+1} \leq \inf \varphi_n$. |

Notes:

1. It is important to note that the points $a_1, \dots, a_n$ that minimize φ_n could not be identical with those that minimize φ_{n+1} .
2. The problem's continuous version is remarkably comparable. Simply said, we use integrals rather than sums. That is

$$\varphi(a_1, \dots, a_{n-1}) = \sum_{k=1}^{n+1} \inf_{f \in \mathcal{F}} \int_{I_k} (f(t) - x_t)^p dt$$

where $I_k = [a_{k-1}, a_k]$ with the convention $a_0 = 0$ and $a_{n+1} = T$. The difference between continuous and discrete is that we can differentiate $\varphi(a_1, \dots, a_{n-1})$ to track the minimizing points.

It is crucial for regression analysis to comprehend how approximation errors behave at various intervals. The following lemma quantifies this behavior:

Lemma 2 *Let I be a subset of J . Then*

$$\inf_{f \in \mathcal{F}} \sum_{i \in I} (f(t_i) - x_{t_i})^2 \leq \inf_{f \in \mathcal{F}} \sum_{j \in J} (f(t_j) - x_{t_j})^2$$

Proof. First, we have

$$\inf_{f \in \mathcal{F}} \sum_{i \in I} (f(t_i) - x_{t_i})^2 \leq \sum_{i \in I} (f(t_i) - x_{t_i})^2 \leq \sum_{j \in J} (f(t_j) - x_{t_j})^2$$

then we obtain the inequality by applying $\inf_{f \in \mathcal{F}}$ again. ▮

Regression analysis is significantly impacted by this lemma. It emphasizes how crucial it is to take localized subsets of data into account when creating regression models. We can capture more localized trends and possibly increase model accuracy by concentrating on subsets. Significantly, it emphasizes that if a single regression function is applied during the whole interval, more data may result in more errors. Furthermore, the lemma directs the choice of model by highlighting the advantages of dividing the dataset into smaller intervals and fitting regression models to each subset independently. This approach may result in enhanced prediction efficacy and a more thorough comprehension of the data dynamics.

To summarize, the lemma is a fundamental concept in regression analysis that offers a valuable understanding of the trade-offs between approximation accuracy and dataset granularity. Researchers and practitioners can make better decisions in regression modeling by comprehending and putting this theory to use.

3.3 Existence of Regression Minimizer in Function Family $\mathcal{F}$

For regression analysis, it is crucial to determine if a function in a particular family $\mathcal{F}$ minimizes the difference between the observed data points and the model predictions. This difference is an essential indicator of the model's accuracy and is frequently measured using an appropriate error metric like the sum of squared differences or the absolute differences.

Consider the set of all potential error values that can be found by applying the observed data points to each function $f \in \mathcal{F}$. This set is a subset of the real line and is represented as $\sum_j |y_j - f(x_j)|^p \mid f \in \mathcal{F}$. This set is notable because it admits an infimum since it is bounded below and non-empty.

To express it in mathematics, there is a function $f_{\min} \in \mathcal{F}$ such that $f_{\min}$ incurs the smallest error of all the functions in $\mathcal{F}$. However, unless further restrictions, such as convexity or linearity, are imposed on $\mathcal{F}$, the uniqueness of $f_{\min}$ is not assured.

As an illustration, if $\mathcal{F}$ is a vector space or a convex set, then $f_{\min}$ is the unique minimizer. This is true because convex sets exhibit a few desirable minimization-related characteristics, namely global optimality and the existence of unique closest points.

Theorem 2 *Distance to a Closed Convex Set: Let H be a Hilbert space, and K a closed non-empty convex subset of H . Then, for every $h \in H$, there exists a unique $k_h \in K$ minimizing the distance of h to K .*

This theorem's classical proof can be found in most textbooks on functional analysis.

Here, we list the essential steps that are involved:

- Consider H as the set of functions from the input space X to the output space Y .
- Assume that $K = \mathcal{F}$, where $\mathcal{F}$ is the set of potential functions from which the

regression model is chosen. A possible model for approximating the relationship between the input and output variables is provided by each function f in $\mathcal{F}$.

- Define the inner product between two functions f and g as $\langle f, g \rangle = \sum_{x \in X} f(x)g(x)$. The inner product between functions f and g over the domain X assesses their similarity or correlation. Consequently, the corresponding norm (squared) of a function f is $\|f\|^2 = \langle f, f \rangle = \sum_{x \in X} f(x)^2$, representing the squared Euclidean distance of f from the origin in the function space.

3.4 Algorithm: Data Partitioning and Change Point Detection

This section explains the algorithm that was applied to partition the data and find the change point. As we discuss in the method's section, our method works on a set of all approximation functions.

3.4.1 Overview

The algorithm seeks to determine the change point, or point at which the underlying pattern shifts, and to partition the data into segments. The data must be finite, and the number of segments must be predetermined. The data structure also influences the function type selection. Due to time constraints, we concentrate on the linear case here, but the technique can be expanded to other function types in the future.

3.4.2 Algorithm Steps

1. **Input:** Data set D , number of segments k , n observations.
2. **Starting Point:** Set up variables and parameters.
3. **Partitioning:** The data collection D should be divided into k segments.
4. **Segmentation:** For each segment, fit a linear approximation function.
5. **Change Point Detection:** Determine the change point based on the differences between adjacent segments.

3.4.3 Linear Approximation Function

The selection of a linear approximation function is impacted by time limitations. Future iterations of the method could, however, expand to include other function types, such as exponential or quadratic functions.

3.4.4 Building the R Code for Change Point Detection

Two cases are shown by the R code in the appendix: one is made to find a single change point, while the other is made to find two change points. It is important to highlight that the number of partitions must be predetermined and that the dataset must be limited. Below is an explanation of the key steps involved in constructing the code.

Input

- **Data:** A dataset with input variables x and output variables y .

Output

- **Change Point:** The detected change point in the dataset.

Steps

1. Prepare Data:

- Open the dataset.
- Make sure the data is finite.
- Specify how many divisions there are.

2. Compute Linear Regression:

- Determine the slope and intercept of a linear regression.
- Compute the sum of squared errors.

3. Detect Change Point:

- Find the best partitioning by going over the dataset iteratively.
- Determine the point of change at which the error is minimized.

Use this $\text{phi}(\mathbf{x}, \mathbf{y})$ to detect a change point with our algorithm.

This presented algorithm simplified the approach for using linear regression to find change points in a dataset. It is important to remember that the analysis can be made

more flexible by adjusting the regression function selection according to the features of the data.

Illustrative Examples of Change Point Analysis

To demonstrate the effectiveness and context sensitivity of our algorithm, we performed an experiment using synthetic data that simulated alternating upward and downward trends. When we used our approach to this dataset, we discovered remarkable findings that shed light on the importance of change points and their consequences for regression analysis.

In the first scenario (a), our algorithm found the point of change at 10.42, dividing the data into segments with separate trends. Figure 3.3 shows the segmented data with two regression lines fitted to every segment.

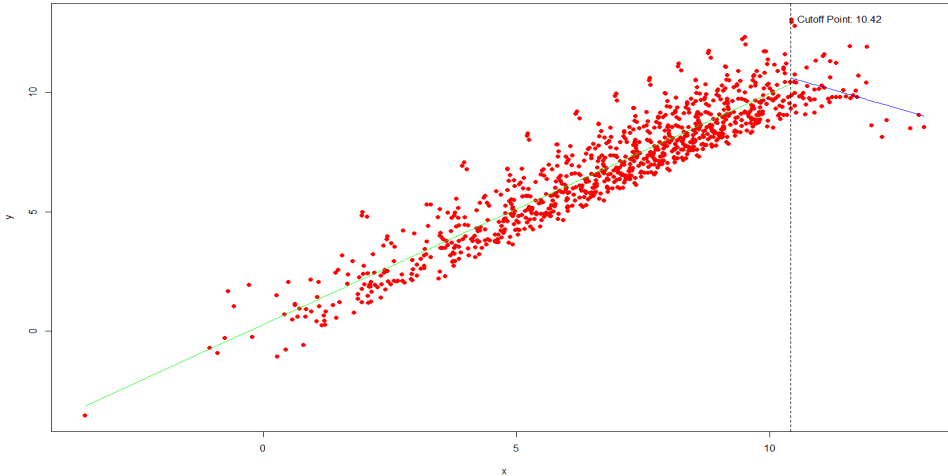


Figure 3.3: (a)

In contrast, in the second Scenario (b), the identified change point occurs at the

end of the observation period. Figure 3.4 shows that this change point makes no substantial contribution to data segmentation. In such circumstances, a single regression line can accurately depict the underlying trend.

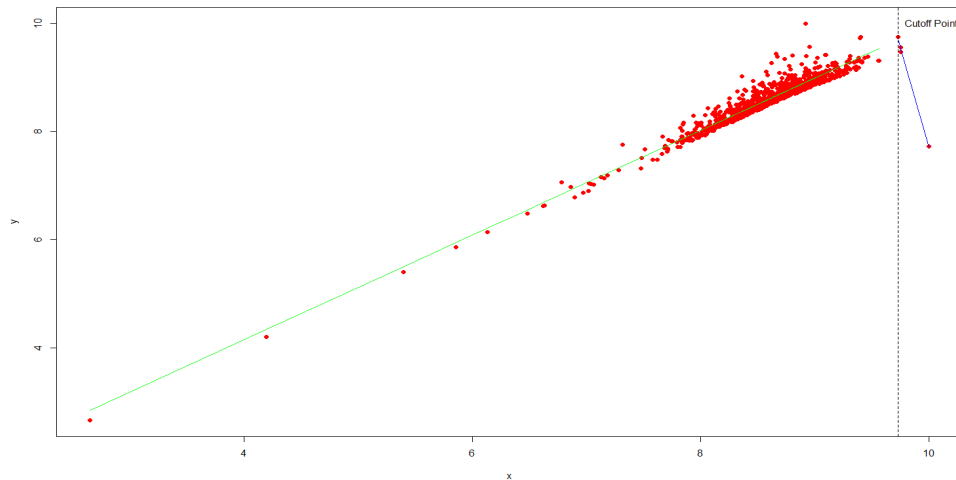


Figure 3.4: (b)

Note: Comparison with Spline and Piecewise Regression:

Our approach provides a systematic and objective way of data segmentation, unlike traditional methods like piecewise regression and spline interpolation.

Piecewise Regression: Piecewise regression depends on manually defined breakpoints, which are frequently visually inspected and judged subjectively. This significantly depends on human judgment and may produce inconsistent outcomes.

Spline Interpolation: The locations of the knots serve as breakpoints for spline interpolation, which fits smooth curves to the data. Despite being flexible it

can have complexity and overfitting problems, particularly when there are a lot of knots.

Our Approach: By using an algorithmic framework, our method reduces the dependence on subjective judgment and maximizes breakpoint detection. We specify criteria, such as minimizing error or maximizing model fit, and then use statistical methods and optimization algorithms to systematically find breakpoints. Compared to manual breakpoint selection or spline interpolation, this method guarantees more effective and consistent outcomes.

3.5 Statistical Analysis of Optimal Points $(a_1, \dots, a_n)$

In this section, we do a statistical analysis of the optimal points produced by our algorithm for data partitioning. The optimal points, denoted as $(a_1, \dots, a_n)$, are critical for segmenting the dataset and detecting change points. We aim to understand the behavior and relevance of these ideal points in data analysis using a variety of statistical measurements and distributions.

We begin our investigation by examining the scenario when $n = 1$, focusing on specific scatter plot distributions to provide a thorough analysis of the most optimal points in various contexts.

The procedure involves generating variables y from standard distributions, including **Standard Normal, Uniform, Beta, Gamma, and Exponential distributions.**

Subsequently, x is expressed in terms of sorted y .

Simulation Setup

We use our algorithm in conjunction with linear regression to find the optimal points to partition the data in order to assess the efficacy of our approach. For each distribution, we run 1000 iterations of simulations to guarantee robustness. This gives us a vector of cutoff points, a , which serve as random variables.

Analysis

Since we have the cutoff point vector in hand, we can examine its statistical characteristics. The mean and the cumulative distribution function (CDF) are the two main metrics that we pay attention to.

By showing the average cutoff point value over several simulations, the mean offers a measure of central tendency. We can learn more about the optimal partitioning points' typical locations and how they vary amongst distributions by examining the mean.

Distribution	Mean of Distribution	Mean of Cutoff Point Vector
Standard Normal	0	0.01007411
Uniform	0.4761584	0.4875203
Beta	0.2934083	0.2891979
Gamma	1.620334	1.526254
Exponential	0.5912789	0.5366055

Table 3.1: Mean values for the distribution and cutoff point vector across different distributions.

As can be seen in Table 3.1, the empirical average of the cutoff points nearly matches the average of each distribution. It's not a given that sampling from a distribution would produce cutoff points that match the distribution's average. Precise

alignment between cutoff points and the distribution's average is not guaranteed by sampling alone. Our investigation reveals a correlation between the two averages, indicating a relationship that goes beyond random variation.

Remark 3 *The observed relationship between the empirical average of cutoff points and the average of each distribution is not necessarily intuitive.*

Consider the following counter-example, which shows that the mean of the minimum statistic is not always equal to the mean of the underlying uniform distribution.

To calculate the mean of the minimum statistic for a uniform distribution from 0 to 1, first derive the probability density function (PDF) of the **minimum statistic** $T = \min(X_1, X_2, X_3, X_4)$, where X_i are independent and identically distributed random variables following a uniform distribution.

For a uniform distribution $U(0,1)$, the PDF $f(t)$ is constant throughout the interval $(0,1)$, that is, $f(t) = 1$ for $0 < t < 1$. The CDF $F(t)$ is defined as $F(t) = t$ for $0 < t < 1$. Now, compute the PDF of the minimum statistic T . The minimum statistic T is the minimum of four independent random variables, each with a uniform distribution $U(0,1)$.

The PDF of T is given by:

$$f_T(t) = n \cdot (1 - F(t))^{n-1} \cdot f(t)$$

Substituting the values, we get:

$$f_T(t) = 4 \cdot (1 - t)^3 \cdot 1$$

Now, to compute the mean of T , integrate $t \cdot f_T(t)$ over its support $(0, 1)$:

$$\text{Mean of } T = \int_0^1 t \cdot 4(1 - t)^3 dt$$

Upon computation, we obtain Mean of $T = \frac{7}{5}$.

When comparing this to **the mean of the uniform distribution, which is $\frac{1}{2}$** , we see that they are not comparable. This indicates that the mean of the minimum statistic is not always equal to the mean of the underlying uniform distribution.

The cumulative distribution function (CDF) is important because it measures the probability that a random variable (in this case, the cutoff points) would have a value less than or equal to a specific point. This attribute is critical because it allows us to evaluate the probability distribution of the cutoff points and understand their variation over simulations. Additionally, studying the CDF allows us to find patterns and trends in the distribution of optimal points, which is useful for decision-making and model selection.

To further understand the behavior of the cutoff points, we calculated the cumulative distribution function (CDF) for each of the five distributions, as shown at

the end of this chapter in Figures 3.5,3.6,3.7,3.8,3.9. Using the inversion method, we produced random variables from these distributions to approximate the cutoff points' distribution and variability. **The inversion method selects random values and uses the inverse CDF to find their corresponding points.**

To demonstrate the inversion approach for the exponential distribution, consider generating random variables from an exponential distribution with a rate parameter of $\lambda = 2$.

The exponential distribution's probability density function (PDF) is as follows:

$$f(x) = \lambda e^{-\lambda x}$$

where λ is the rate parameter and $x \geq 0$.

The CDF for the exponential distribution is:

$$F(x) = 1 - e^{-\lambda x}$$

Now, let's use the inversion method to generate random variables from this distribution:

1. Generate a random variable u from a uniform distribution ranging from 0 to 1.
2. To calculate the value of x , use the exponential distribution's inverse CDF. The CDF is monotonically growing, hence its inverse can be calculated as:

$$x = -\frac{1}{\lambda} \ln(1 - u)$$

3. The generated value of x follows the exponential distribution with the specified rate parameter.

For example, consider generating a random variable from an exponential distribution with $\lambda = 2$:

1. Generate a random value $u = 0.567$ (randomly chosen).
2. Calculate the corresponding value of x :

$$x = -\frac{1}{2} \ln(1 - 0.567) \approx 1.099$$

3. The produced value of x is approximately 1.099, based on an exponential distribution with an exponential rate parameter of 2.

This method can be repeated to generate many random variables from the exponential distribution, allowing us to simulate and examine cutoff point behavior using the underlying distribution.

3.5.1 Purpose of Statistical Analysis for Cutpoints

Traditionally, a scatterplot is segmented into two sections using a simple approach: randomly selecting a cutpoint from the data points, denoted as $a \sim \text{Uni}(x_1, \dots, x_N)$. However, our research goes beyond randomness by attempting to find the precise cutpoint, even if it demands more computational resources.

While the exact approach may be computationally expensive, it has a major

benefit over random selection. Although knowing the distribution of a does not provide the same level of precision as the precise procedure, it does outperform blind random selection. Randomly picked cutpoints tend to cluster in the middle of the data, however understanding the distribution of a allows for selection closer to the true optimal point, on average.

Using the cumulative distribution function (CDF), we gain useful insights about the empirical distribution of the data. This knowledge enables us to produce cutpoints that are consistent with the observed data distribution, improving the accuracy of scatterplot segmentation and analysis.

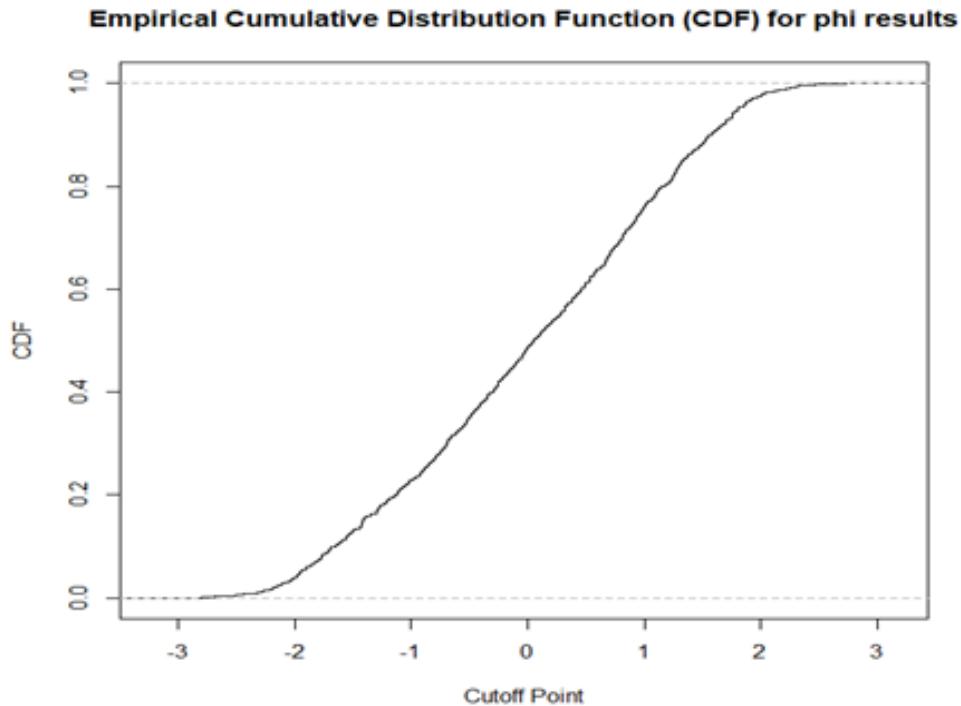


Figure 3.5: CDF Plot: Standard Normal

irical Cumulative Distribution Function (CDF) for phi results (Uniform Disi

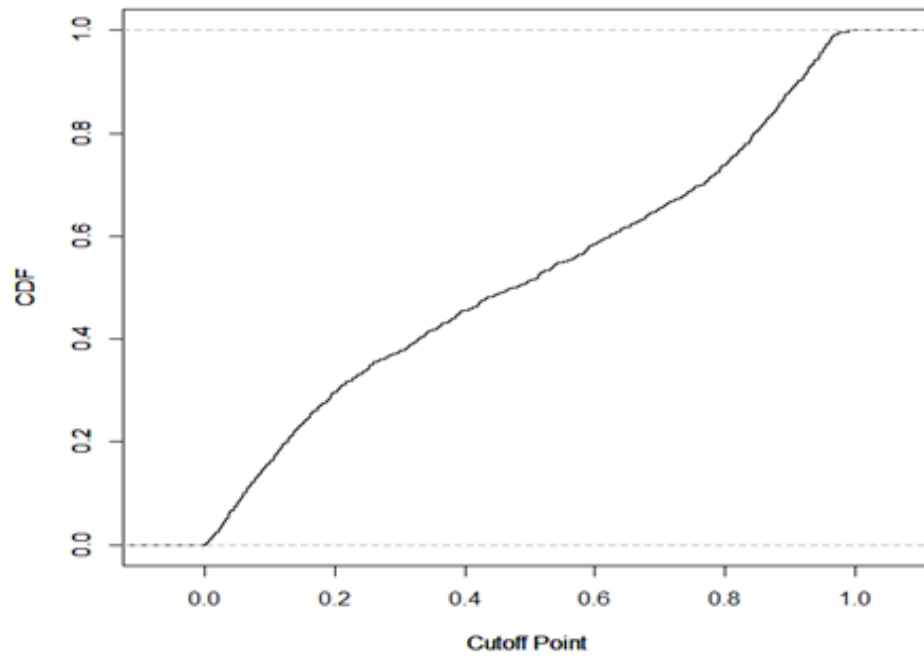


Figure 3.6: CDF Plot: Uniform

cal Cumulative Distribution Function (CDF) for phi results (Exponential D

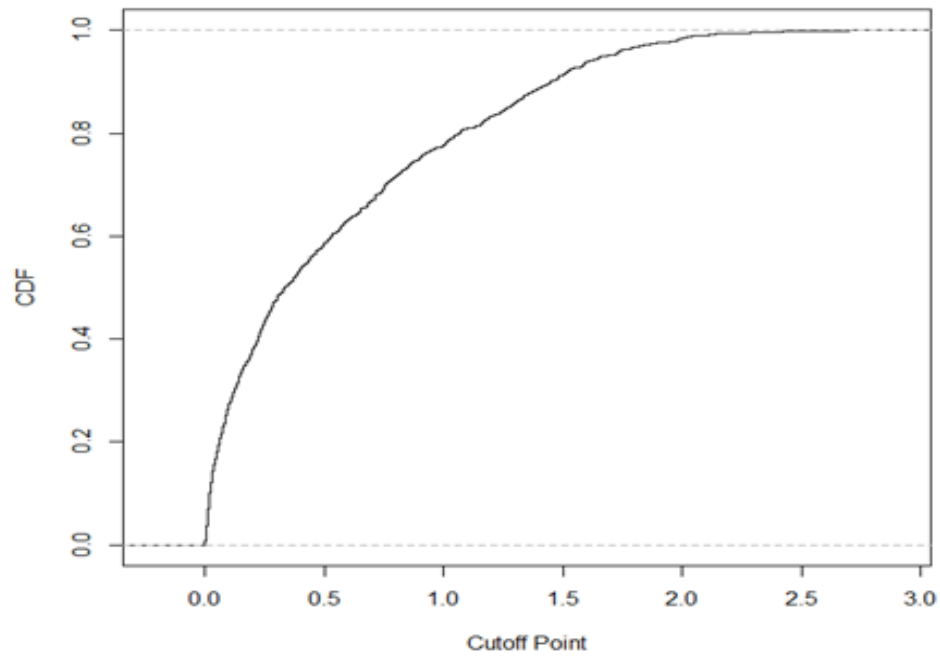


Figure 3.7: CDF Plot: Exponential

Empirical Cumulative Distribution Function (CDF) for phi results (Gamma Disi

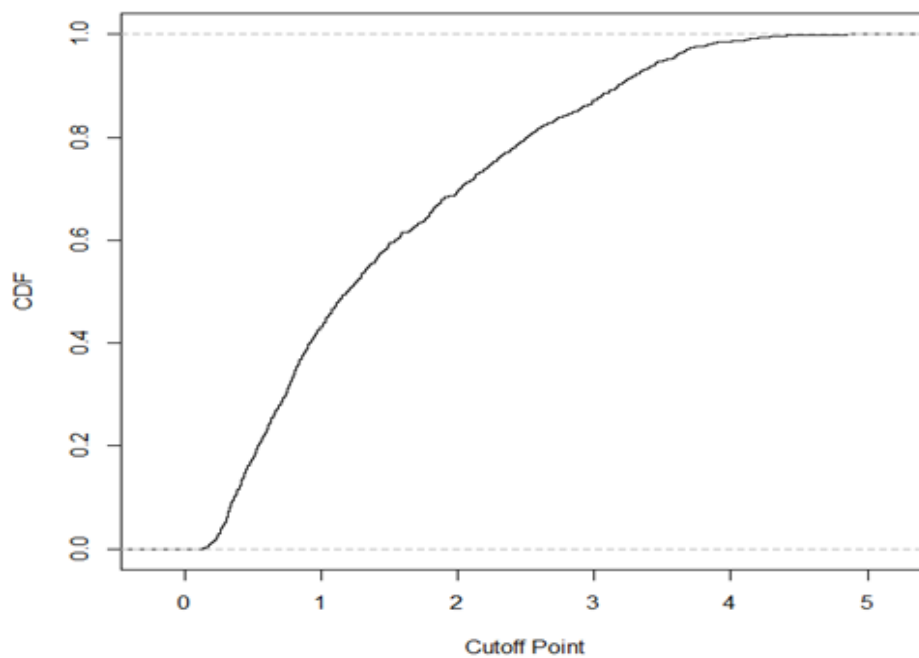


Figure 3.8: CDF Plot: Gamma

Empirical Cumulative Distribution Function (CDF) for phi results

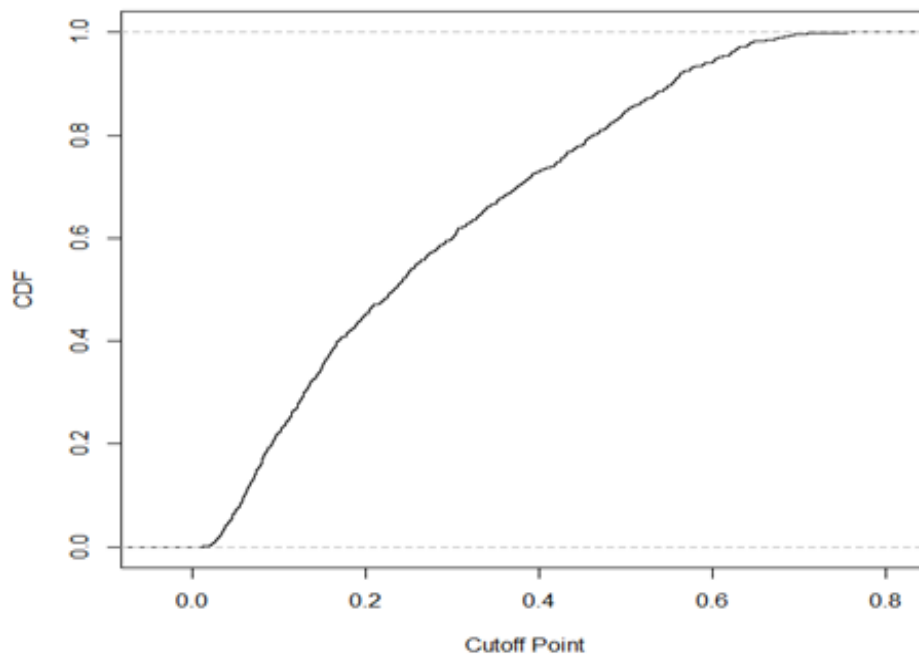


Figure 3.9: CDF Plot: Beta

CHAPTER 4

PROPOSED METHODOLOGY: REAL-TIME TRACKING

In this chapter, we challenge typical data clustering approaches that need particular intervals and the entire dataset. Real-world scenarios usually comprise continuously generated or incomplete data, demanding a more adaptable strategy. We provide a dynamic framework for processing dynamic data and producing continuous intervals. Unlike conventional methodologies, our methodology takes into account the dynamic nature of real-time data streams. This chapter delves into the theoretical basis of this dynamic framework. This study enhances the field of dynamic data clustering by proposing an adaptable approach that considers the complexities of data evolution in the real world.

4.1 The Method: Real Time Tracking

Assume a scatter plot (X, Y) , as usual. Define the following series of random times, with ε a positive integer. Every time step t adds a new data point to the scatter plot, demonstrating the dataset's dynamic character. Using an incremental technique, where ε regulates the granularity of temporal updates, patterns and trends in real-time data can be investigated. When striving to capture the nuances of data evolution and gain insight into the changing interactions between factors, the dynamic scatter plot proves to be an effective tool.

In our study, we present a dynamic approach for assessing a changing scatter plot (X, Y) . We generate a random time series, T_j^ε , where j represents the iteration. These times record when the error function Φ_p surpasses a specific threshold ε in subsets of the dataset. The recursive definition of the series is shown below:

$$\begin{aligned}
T_1^\varepsilon &= \inf\{1 \leq n \leq N \mid \Phi_p(Y_{\{x_1, \dots, x_n\}}, f) \geq \varepsilon\} \\
T_2^\varepsilon &= \inf\{1 + T_1 \leq n \leq N \mid \Phi_p(Y_{\{x_{1+T_1}, \dots, x_n\}}, f) \geq \varepsilon\} \\
T_3^\varepsilon &= \inf\{1 + T_2 \leq n \leq N \mid \Phi_p(Y_{\{x_{1+T_2}, \dots, x_n\}}, f) \geq \varepsilon\} \\
&\vdots \\
T_j^\varepsilon &= \inf\{n \geq 1 + T_{j-1} \leq n \leq N \mid \Phi_p(Y_{\{x_{1+T_{j-1}}, \dots, x_n\}}, f) \geq \varepsilon\}.
\end{aligned}$$

This dynamic sequence reveals the temporal evolution of the error function, allowing us to identify important points when the dataset changes significantly.

Using this method, we may examine the scatter plot adaptively over time and detect

trends that a static analysis might miss.

We note that our method allows us to effectively examine the evolving scatter plot (X, Y) regardless of whether the dataset is finite or infinite. This method provides useful insights into real-time data patterns and trends by gradually updating the scatter plot with new data points. **One of the significant advantages of our method is its ability to automatically identify the number of segments inside the dataset depending on the selected error threshold**, eliminating the need to specify this parameter initially. By defining the error threshold from the start, we ensure exact error handling. Furthermore, **the approach offers both the change points and the number of segments as output, boosting its applicability in real applications.**

4.1.1 Error Threshold Determination

In our method, we determine the error threshold, ε , as the maximum absolute difference between consecutive observations divided by the total number of observations (n). This approach ensures that the error threshold is proportional to the variability of the data, facilitating robust analysis of the scatter plot (X, Y) . We choose the following error ε :

$$\varepsilon = \frac{\max(|y_{i+1} - y_i|)}{n}$$

Where y_i and y_{i+1} are consecutive observations, and n is the total number of

observations in the collection. This method gives a standardized measure for detecting significant changes in the scatter plot, which improves the accuracy of our results.

4.2 Analysis of Dynamic Data Evolution

In this section, we delve into the complexities of dynamic data evolution, proposing and thoroughly demonstrating two essential principles that illuminate the temporal dynamics of datasets.

Proposition 1 *The number of random times is at most $\frac{N+1}{2}$.*

Proposition 1 provides a loose boundary on the occurrence of random times (T_j) during dynamic data analysis. It states that the greatest number of random times observed does not exceed $\frac{N+1}{2}$, where N is the dataset's size. This proposition sheds light on the temporal evolution of datasets, imposing a quantitative constraint on the frequency of important changes across time.

Proof: Let T_{k_N} denote the last random time. We can observe that:

$$N - 1 = T_{k_N} - T_1 = \sum_{j=2}^{k_N} (T_j - T_{j-1}) \geq 2(k_N - 1),$$

which implies $k_N \leq \frac{N-1}{2} + 1 = \frac{N+1}{2}$. Thus, the number of random times is at most $\frac{N+1}{2}$.

Proposition 2 *In terms of ε , the random time T_j is non-increasing.*

Proposition 2 investigates the impact of random times (T_j) on the error threshold (ε).

It asserts that as the error threshold is reduced, random times either remain constant

or drop in value. This discovery emphasizes the association between error thresholds and large changes in the dataset, providing useful insights into its dynamic character.

Proof: Consider $\delta < \varepsilon$. If $\Phi_p(Y_{|\{x_1, \dots, x_n\}}, f) \geq \varepsilon$, then $\Phi_p(Y_{|\{x_1, \dots, x_n\}}, f) \geq \delta$, implying:

$$\{1 \leq n \leq N \mid \Phi_p(Y_{|\{x_1, \dots, x_n\}}, f) \geq \varepsilon\} \subset \{1 \leq n \leq N \mid \Phi_p(Y_{|\{x_1, \dots, x_n\}}, f) \geq \delta\}.$$

Taking the infimum yields $T_1^\delta \leq T_1^\varepsilon$, and similarly, $T_j^\delta \leq T_j^\varepsilon$. Thus, the random times exhibit a non-increasing trend concerning the error threshold.

According to the preceding proposition, reducing the error threshold leads to more frequent occurrences of random times. In essence, decreasing the error threshold increases the probability of meeting substantial changes in the dataset. Through careful analysis and rigorous demonstration of these principles, we acquire deeper insights into the dynamic evolution of datasets, paving the path for better knowledge and future research in this area.

4.3 Algorithm: Real-Time Tracking for Dynamic Data Analysis

This section describes the algorithm for real-time tracking, which allows for dynamic data analysis by recognizing change points and segmenting the dataset.

4.3.1 Overview

The real-time tracking algorithm works on a dataset D , enabling dynamic analysis regardless of whether the data is finite or not. Using an error threshold ε , the algorithm detects change points and splits the dataset accordingly.

4.3.2 Algorithm Steps

- **Input:** Dataset D with error threshold ε .
- **Initialization:** Initialize the parameters and variables.
- **Compute Error Function** Calculate the error function Φ_p for each data point with the chosen model f .
- **Detect Change Points:** Identify instances where the error function exceeds the threshold ε to determine change points.
- **Segmentation:** Divide the dataset into segments depending on the discovered change points.
- **Output:** Number of segments and change points.

4.3.3 Flexibility of Model Selection

While our approach is designed to accept various families of functions, such as linear, quadratic, or exponential functions, for the purpose of simplicity and time restrictions, we concentrate on the linear approximation function in this implementation.

4.3.4 R Code Implementation

The appendix contains R code that demonstrates the implementation of the real-time tracking approach for dynamic data analysis. The code shows how to discover change points, segment the dataset, and derive useful insights from the analysis in real-time.

CHAPTER 5

APPLICATION AND STATISTICAL ANALYSIS

In this chapter, we apply the methodologies introduced in previous chapters to real-world datasets. We conduct statistical analyses to assess the proposed algorithms' performance and effectiveness in real-world circumstances. In addition, we investigate the distribution and features of detection findings in order to get insights into the algorithms' behavior across diverse datasets and conditions.

5.1 Data Description

The considered dataset contains real power consumption data obtained from a nuclear power facility. It is made up of a time series of power consumption figures in megawatts (MW), ranging from 203.2208 to 216.4605 MW. A total of 744 samples were collected from real observations of electricity use at the nuclear power plant. Given the nature of this time series dataset, the variables were sampled hourly for an

entire duration of 744 hours.

Source: The data on power consumption were obtained from the Kaggle dataset available at Electric Power Consumption by Nuclear Power Plant.

To explore the dataset further, we visualize the power consumption data using two different plots: **a time series plot and a scatter plot.**

The time series plot shows the changes in power use over time, revealing any underlying trends or patterns. This visualization shows how power usage varies throughout the duration of the dataset, allowing us to discover any recurring trends or anomalies.

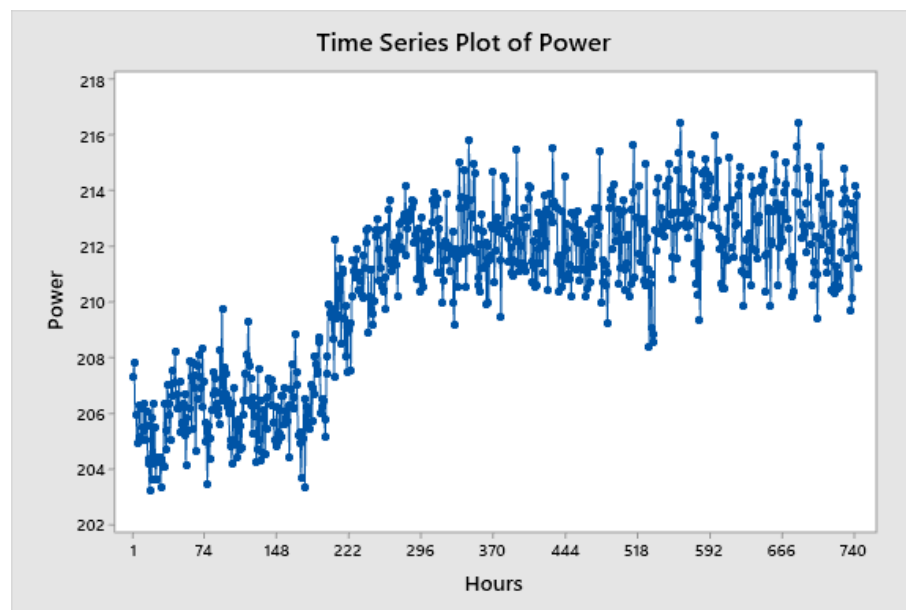


Figure 5.1: Time Series Plot

Meanwhile, the scatter plot depicts the relationship between the input factors (such as the time index) and the resulting power consumption numbers. This plot allows us to evaluate the applicability of our method to the dataset by studying the

distribution and dispersion of data points. Analyzing the scatter plot allows us to acquire a better understanding of how power consumption fluctuates with regard to different input variables, potentially revealing any nonlinear correlations or clusters in the data.

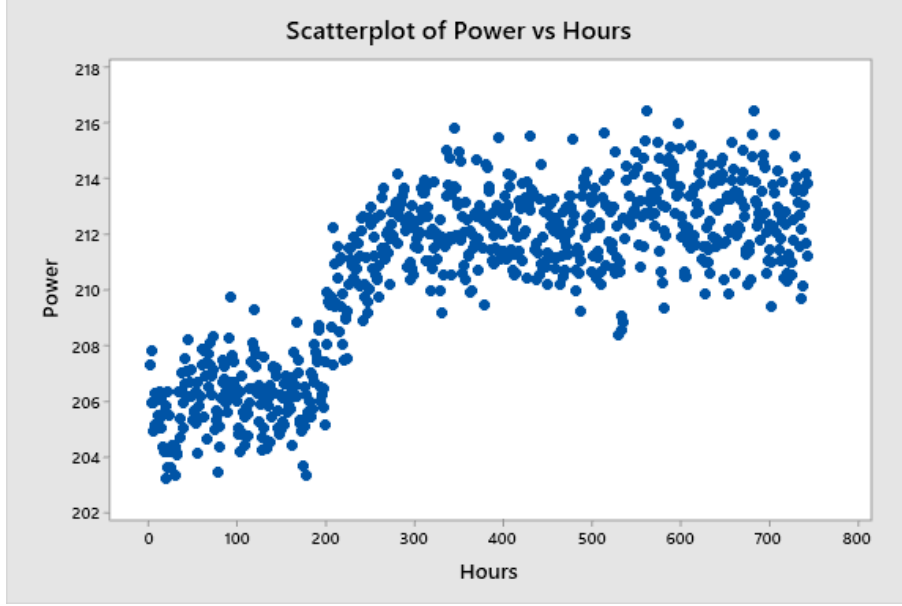


Figure 5.2: Scatterplot

Both visualizations are useful tools for evaluating data distribution and identifying potential patterns that may affect the implementation of our approach. Furthermore, the scatter plot provides useful insights into the power usage pattern, allowing us to make informed judgments on how to implement and optimize our method.

5.2 Comparison with Global Regression Model

In this part, we compare the performance of our change point detection approaches against a global regression model applied to the dataset. The comparison attempts to assess the efficacy and accuracy of our methods for detecting localized trends and

change points.

5.2.1 Global Regression Model

We start by fitting a global regression model to the entire dataset using the linear regression method. This model is used as a baseline for comparison with our segmented regression approach.

Methodology

To fit a single regression line to the dataset, we use the linear regression procedure. Linear regression is a statistical approach for modeling the connection between a dependent variable and one or more independent variables. In our case, we use linear regression to find the best-fitting regression line that depicts the overall trend in the dataset without breaking it down into separate components. The linear regression model is represented as follows:

$$y = \beta_0 + \beta_1 \cdot x + \epsilon$$

In the linear regression model, y is the dependent variable (power consumption), while x is the independent variable (time). The intercept β_0 reflects the value of the dependent variable when the independent variable is zero, whereas β_1 is the slope of the regression line, which indicates the rate of change in the dependent variable for a one-unit change in the independent variable. Finally, ϵ accounts for the discrepancy between observed and predicted values, reflecting variability that cannot be explained

by the model.

The linear regression model aims to determine the values of β_0 and β_1 that minimize the sum of squared residuals, representing a linear relationship between the dependent and independent variables.

We describe the fitted model and show the estimated coefficients and values. The fitted linear regression model is provided by:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 \cdot x \quad (5.1)$$

Where $\hat{y}$ is the estimated dependent variable (power consumption) and x is the independent variable (time). $\hat{\beta}_0$ denotes the estimated intercept of the regression line, while $\hat{\beta}_1$ represents the estimated slope. Using this approach, our fitted linear regression model is as follows:

$$\text{Power} = 206.4 + 0.01107 \cdot \text{Hours} \quad (5.2)$$

In our study, the estimated intercept of 206.4 represents the baseline power consumption at zero time, while the slope of 0.01107 represents the rate of change in power consumption per unit increase in time.

Results

The following are the results obtained by fitting the global regression model.

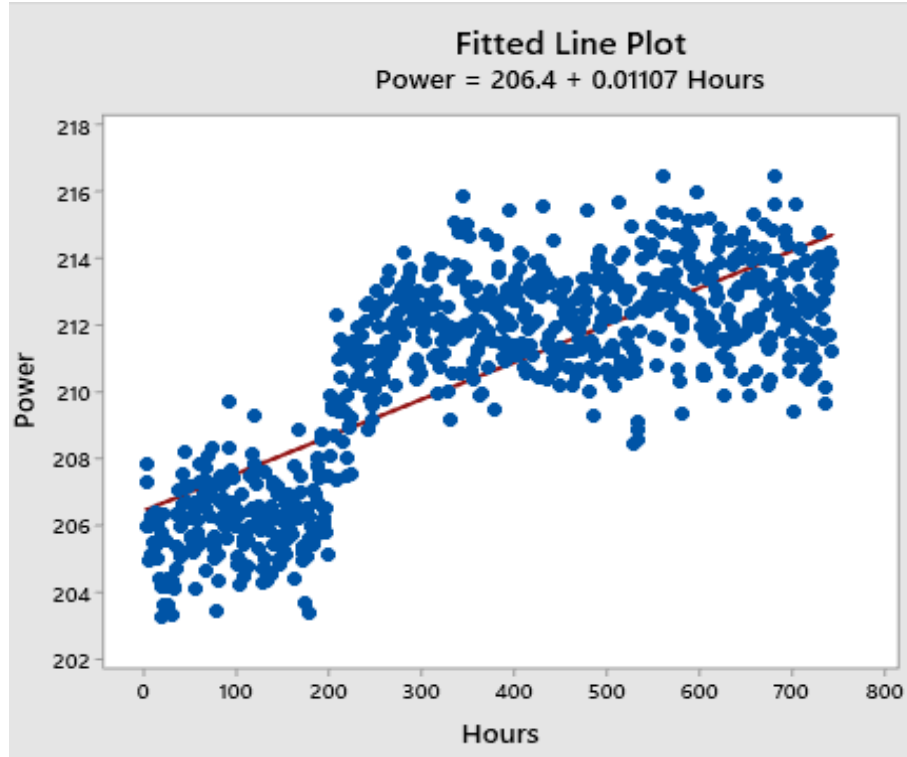


Figure 5.3: Fitted Line

- **Euclidean Distance** = $\sum_j |y_j - f(x_j)|^2$. For the global regression model is **2954.514**
- **Mean Squared Error (MSE)**: The MSE measures the average squared difference between the observed and predicted values. The formula for calculating Mean Squared Error (MSE) is:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

Where n is the number of data points, y_i is the observed value of the dependent variable, and $f(x_i)$ is the predicted value of the dependent variable. For the global regression model, the MSE is **3.98**.

- **Percentage error** is a measure used to assess how accurate an estimate or prediction is. To represent it as a percentage, calculate the absolute difference between the actual value y_i and what was predicted $\hat{y}$, divide that difference by the observed value, and multiply the result by 100. The percentage error formula is:

$$\text{Percentage Error} = \left| \frac{y_i - \hat{y}}{y_i} \right| \times 100\%$$

For the global regression model is **2.575236**.

These results provide insights into the global regression model when applied to the dataset.

5.2.2 Change Point Detection Methods

We use our change point detection methods to partition the dataset and then fit regression lines to each segment. The approaches were introduced and addressed in Chapters 3 and 4. We evaluate the performance of these methods in the global regression model.

To assess the performance of change point detection algorithms, we use metrics such as mean squared error (MSE), Euclidean distance error, and percentage error. These measurements reveal differences in the accuracy and explanatory power of segmented regression models vs the global regression model.

In the following sections, we provide the results of the change point detection methods as well as their implications.

5.2.3 Method 1: Tracking the change of the patterns using optimal partitions

The aim of the algorithm is to partition the data into segments and determine the point at which the underlying pattern shifts. **It needs finite data and a predetermined number of segments.**

Example 1: Single Change Point

Consider a dataset (D) of 744 observations and a segment ($k = 1$). We use this method to find a single change point in the data set. Figure 5.4 depicts the findings. The algorithm recognized a change point at $x = 200$.

Example 2: Two Change Points

Consider the same dataset D with $n = 744$ observations and $k = 2$ segments. We use the method to find two change points in the data set. Figure 5.5 depicts the findings. We discovered that the program identified the change sites at $x = 3$ and $x = 200$.

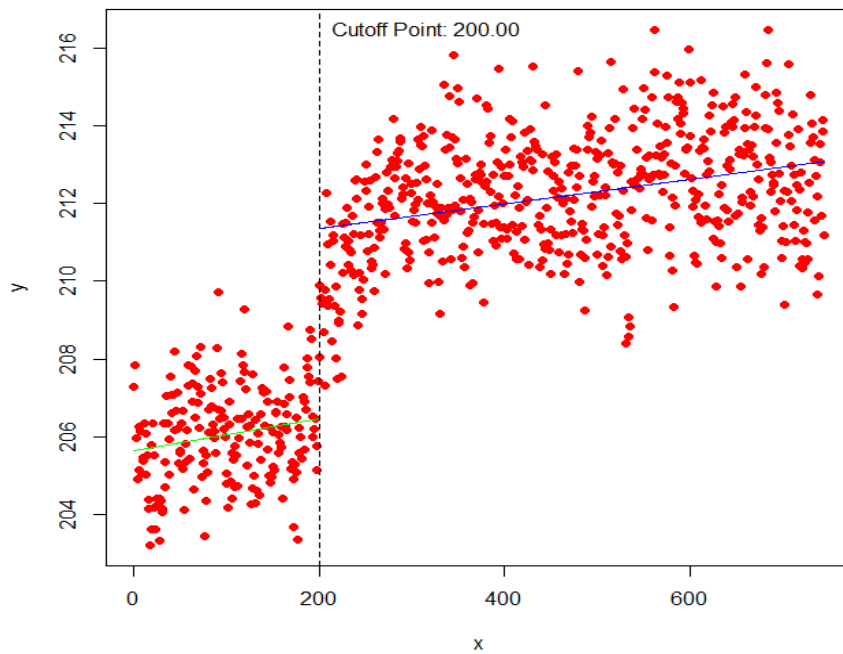


Figure 5.4: Single Change Point

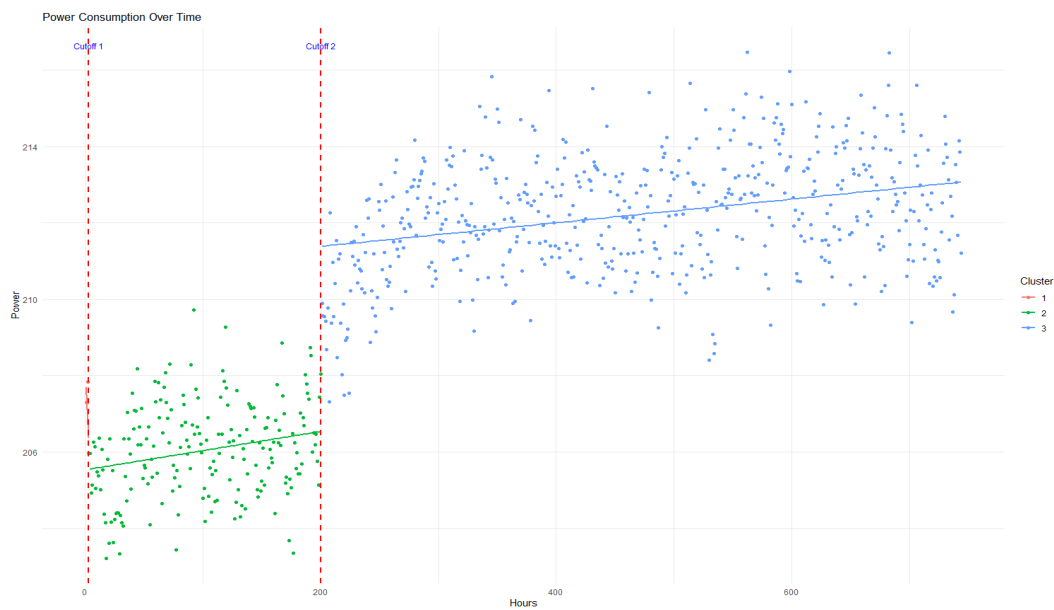


Figure 5.5: Two Change Points

Results and Interpretation

After applying our change point identification technique to the practical example dataset, we examined the regression lines that emerged. Figure 5.5 shows that the first regression line is not clear. **This observation is consistent with our prior discussion in Chapter 3, which emphasized that cutoff points at the beginning or end of a period may not be good.** In the case of the practical example under discussion, a single point is sufficient for our needs.

In the case of a single change point, we obtained two fitted line models reflecting segmented regression analysis:

Model 1 Fitted Line:

$$\text{Power} = 205.6366 + 0.004134994 \cdot \text{Hours} \quad (5.3)$$

Model 2 Fitted Line:

$$\text{Power} = 210.7248 + 0.003167302 \cdot \text{Hours} \quad (5.4)$$

To evaluate the performance of each model, we use error measures such as Mean Squared Error (MSE), percentage error, and Euclidean distance. These metrics provide insight into the accuracy and goodness of fit of each regression model.

Table 5.1: Performance Metrics for Two Models

Model	Euclidean distance	MSE	Percentage error
Model 1	1458.267	1.140012	1.767903%
Model 2		2.165701	1.95915%

After applying our method to identify the single change point at 200, we calculated the Euclidean distance, which came out to be 1458.267. Notably, this value is significantly less than the Euclidean distance of the global error, which was found to be 2954.514. This stark distinction provides substantial evidence in support of our method's efficacy, as mentioned in Chapter 3, displaying improved error minimization performance when compared to the global approach. Furthermore, when we calculated the Mean Squared Error (MSE) for each model, we consistently found that the MSE values were lower than the global MSE. Similarly, the percentage error for each model was found to be less than the global regression error in every case.

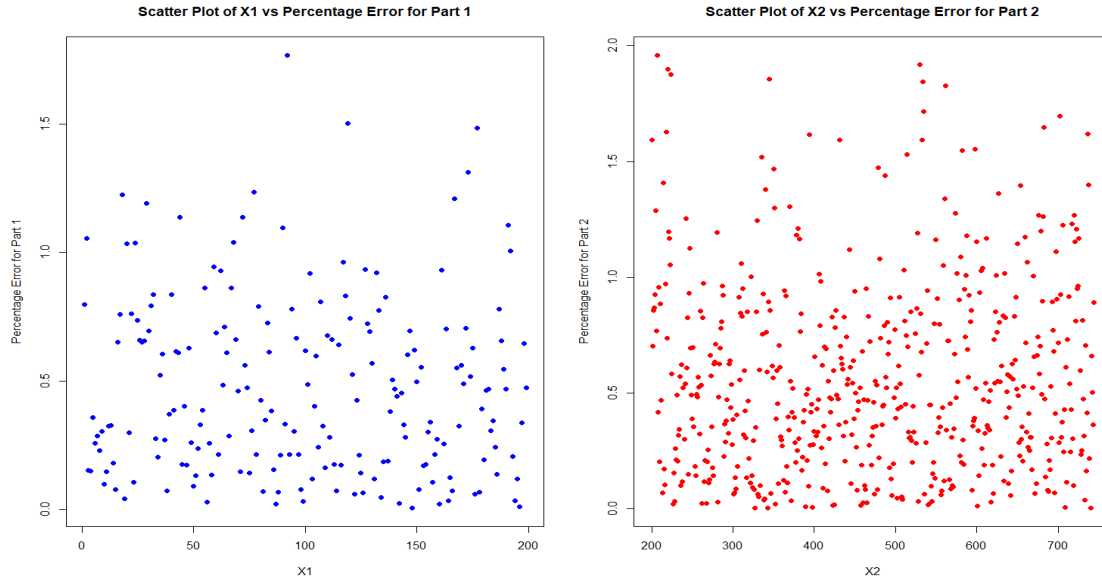


Figure 5.6: Percentage Error

As illustrated in Figure 5.6, we examined the percentage error for two distinct

components: Part 1 and Part 2. In Part 1, which corresponds to the regression model, our investigation found that the majority of errors fall below 1%. This indicates that the regression model is highly accurate in predicting outcomes, with little variations from actual values. In contrast, in Part 2, displayed in the second plot, we found that the percentage error remains primarily below 1.5%. Although the errors in Part 2 are significantly greater than those in Part 1, they remain reasonably low, showing that the model performed satisfactorily in this environment. Overall, these findings highlight both models' ability to collect and predict underlying patterns in data, with minimal discrepancies between projected and actual values.

5.2.4 Method 2: Real Time Tracking

The real-time tracking method attempts to dynamically partition the data into segments and determine the change point at which the underlying pattern shifts. Unlike existing methods, **our methodology does not place constraints on the finiteness of the data.** However, **the error threshold must be specified before starting.** This allows the algorithm to detect significant changes in the data in real-time, resulting in effective pattern identification and segmentation.

Error Calculation

Before starting with our method, we need to calculate the error, represented by ε . The chosen error calculation helps to determine the granularity of temporal updates and the effects of data partitioning. The error is calculated with the following formula:

$$\varepsilon = \frac{\max(|y_{i+1} - y_i|)}{n}$$

where n is the total number of observations, and y_i and y_{i+1} are consecutive data points. This error value is an important element in our approach since it guides the dataset segmentation and makes it easier to identify change points.

For the given dataset with $n = 744$, $y_i = 207.29325$, and $y_{i+1} = 207.84$, the error can be calculated as follows:

$$\varepsilon = \frac{\max(|207.84 - 207.29325|)}{744} = 0.0007356559$$

This error amount is used as a threshold to detect substantial changes in the dataset during real-time tracking.

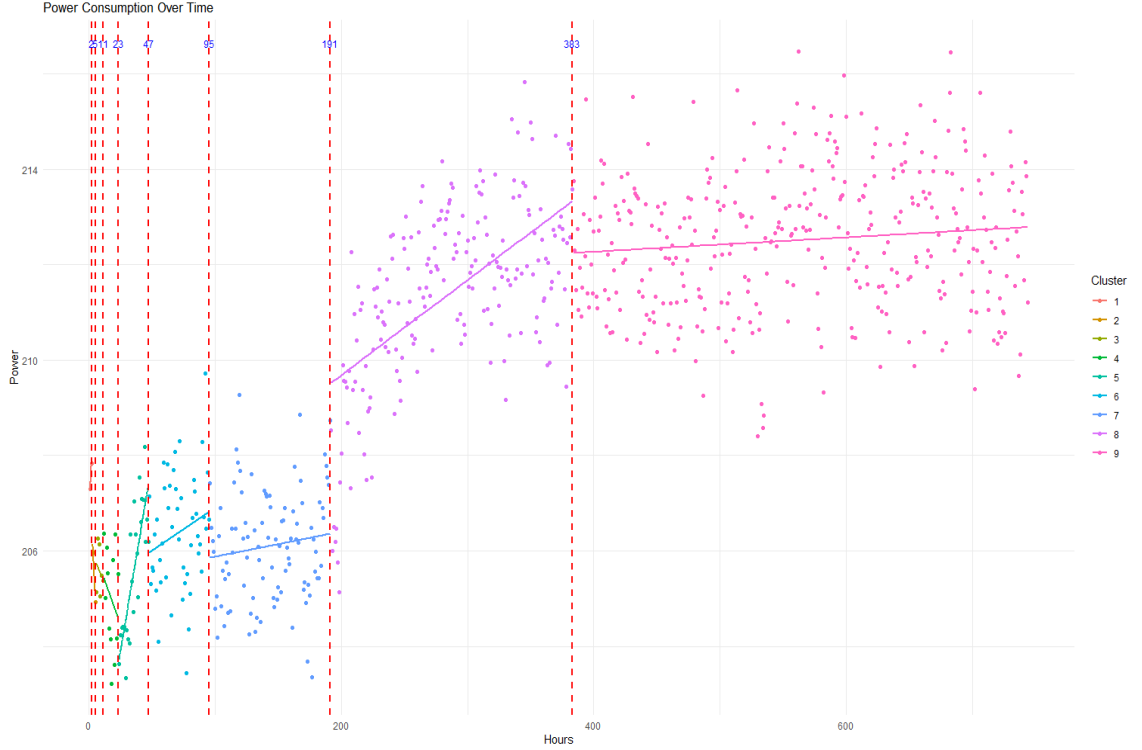


Figure 5.7: Real Time Tracking

Results and Interpretation

Implementing the real-time approach on the practical example dataset yielded eight segments, represented by nine clusters with regression lines. While examining the data, Figure 5.6 revealed that the clarity of the first three regression lines had been compromised. This finding is consistent with our previous discussion, where we noted that cutoff points positioned at the beginning or end of the period may not be important for accurate segmentation.

In the practical scenario under examination, we have nine models resulting from the clustering procedure. **Using the same approach as the first method,**

we evaluated and compared these models to the global regression. This comparative study allows us to evaluate the effectiveness of the real-time algorithm in capturing the underlying patterns and trends in the dataset.

5.2.5 Statistical Analysis

Once real-time tracking methods were implemented, eight discrete **cutoff points** were noted: **2, 5, 11, 23, 47, 95, 191, and 383**. We used the bootstrap resampling technique to thoroughly analyze these cutoff points statistically. By repeatedly sampling from the observed data with replacement, **Bootstrap resampling** is an effective method that enables robust inference. By using this method, we can assess statistics' variability and make inferences about the underlying population distribution.

The Bootstrap Method, which was first introduced in Chapter 1, is now used to investigate the variability of observed cutoff points. This section describes the implementation steps and visualizes the Bootstrap distribution.

Bootstrap Method Implementation

We used the R software environment and the statistical package `boot` to perform the Bootstrap analysis. The following steps were taken to conduct the analysis:

1. To do Bootstrap resampling in R, we first loaded the `boot` library.
2. Then, we created a dataset called `original_data` with the observed cutoff

points for variables **a** and **b** during real-time tracking.

3. For each bootstrap sample, the mean of variables **a** and **b** was computed using a statistic function.
4. We set the number of bootstrap samples to be generated.
5. To evaluate the variability of observed cutoff points, bootstrap resampling was done on the original data using the `boot` function
6. We visualized the Bootstrap distribution of the mean of variable **a** using a histogram.

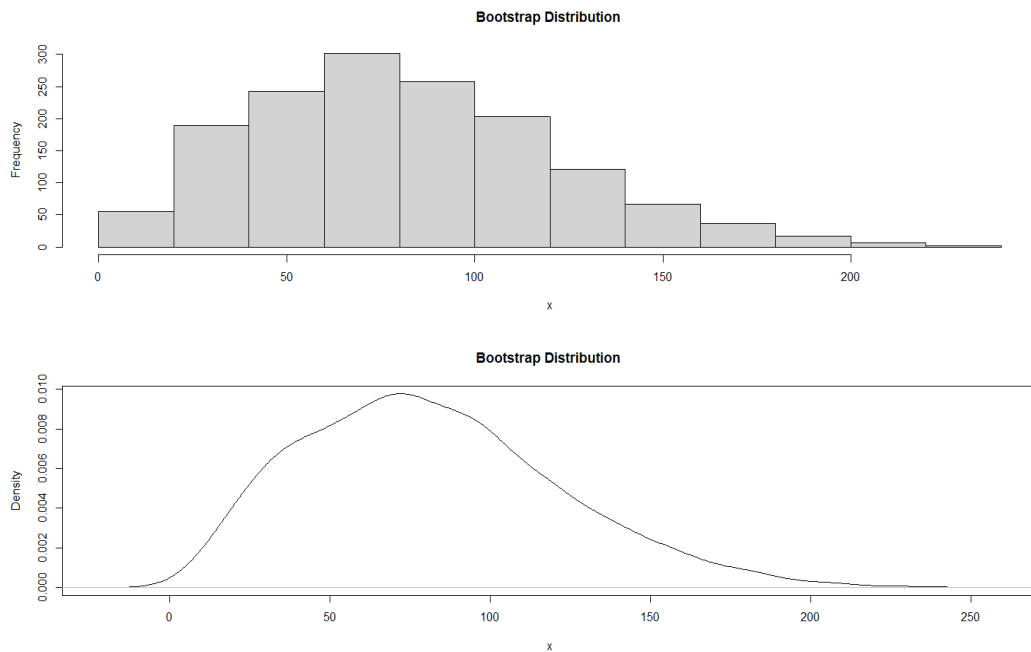


Figure 5.8: Bootstrap Distribution

Figure 5.8's Bootstrap distribution visualization allowed us to see a lengthy, heavy tail to the right, which suggested possible extreme values.

Following bootstrapping, we evaluated the effectiveness of the maximum likelihood estimator (MLE) for three distributions: generalized extreme value (GEV), log-normal, and Frechet distribution. With R's (nlm) function, we conducted this evaluation on several sample sizes: 100, 500, 1000, 1500, and 2000.

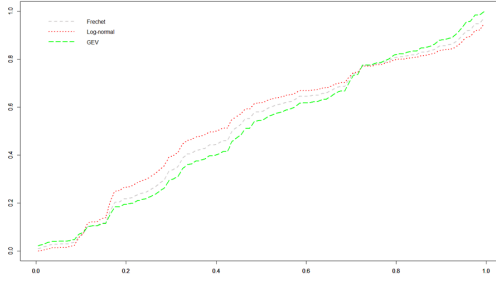
Following the Kolmogorov-Smirnov (KS) test, we obtained corresponding p-values that indicated the degree of concordance between the observed data and the fitted distribution for each distribution (log-normal, GEV, and Fréchet) across various sample sizes (100, 500, 1000, 1500, and 2000). While a higher p-value indicates a more appropriate fit, a low p-value indicates a significant difference between the empirical and theoretical distributions, indicating an inadequate match.

The p-values for the KS test for each distribution at various sample sizes are shown in Table 5.2 below. Across a range of bootstrap sizes, the Fréchet and GEV distributions typically have higher p-values than the log-normal distribution.

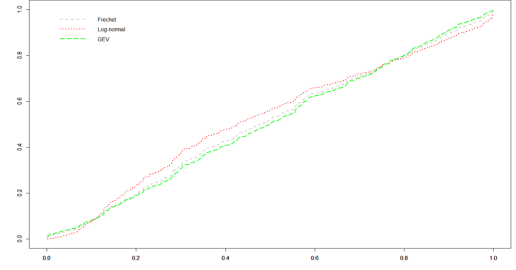
Bootstrap Size	Fréchet Distribution	GEV Distribution	Log-Normal Distribution
100	0.2876	0.8603	0.05425
500	0.3419	0.7897	0.0005809
1000	0.0069	0.2167	4.346e-08
1500	0.04527	0.2082	2.979e-07
2000	0.03751	0.4555	2.684e-10

Table 5.2: KS Test p-values for Different Distributions and Bootstrap Sizes

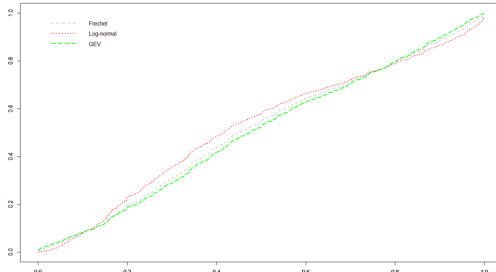
Following that, in order to visually evaluate each distribution's degree of fit to the observed data, we showed the cumulative distribution function (CDF) for the log-normal, GEV, and Fréchet distributions across a range of sample sizes.



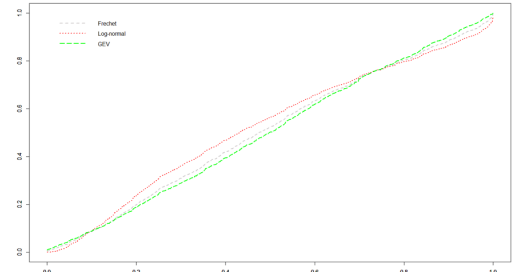
(a) Sample Size 100



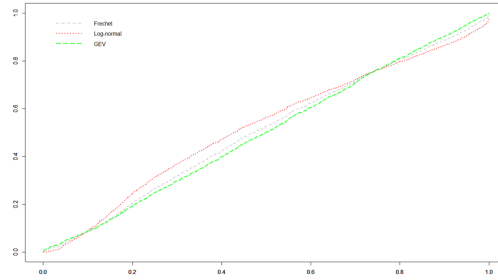
(b) Sample Size 500



(c) Sample Size 1000



(d) Sample Size 1500



(e) Sample Size 2000

Figure 5.9: CDF Plots for Different Sample Sizes

The bootstrapping used in this work provides two key purposes. First of all, it is used to increase the sample size, allowing for a more thorough investigation of various distributions for the optimal values. By increasing the sample size through bootstrapping, we essentially enlarge the dataset accessible for analysis, improving statistical power and allowing for a more robust identification of the underlying distribution of the optimal values.

Secondly, bootstrapping is used to validate our results by demonstrating their reproducibility. This includes demonstrating that our findings are not dependent on a specific dataset but are applicable to other datasets. By resampling from the original dataset, we demonstrate that our conclusions remain reliable and consistent even when applied to different datasets with a comparable data generation process, strengthening the credibility of our research results.

5.3 Application of Bayesian Change Point Analysis

In this section, we extend our investigation into change point identification by performing Bayesian Change Point Analysis on the same dataset that was previously studied using optimization-based approaches. Our examination aims to determine the effectiveness and reliability of Bayesian approaches in identifying changes in both the mean and variance of time series.

Detection of Changes in Mean

Bayesian Change Point Analysis is used to detect changes in the mean of a time series. This method divides the data into segments, each with a distinct mean, and uses Bayesian inference techniques to determine the probable locations of change points. Using the `bcp` function from the `bcp` package in R, the following change points were identified in the mean: **Change Point 1: 35, Change Point 2: 199, Change Point 3: 225, Change Point 4: 260, Change Point 5: 529, Change Point 6: 535, and Change Point 7: 696**, as illustrated in Figure 5.10.

Detection of Changes in Variance

Similarly, Bayesian Change Point Analysis is used to detect changes in the variance of the time series. By assuming that variance varies across segments, the approach predicts the posterior distribution of change points, indicating large changes in observational variability. Using the `changepoint` or `bcp` packages in R, the analysis found the following change points in the variance: **Change Point 1 at 200, Change Point 2 at 276, and Change Point 3 at 744**, as shown in Figure 5.11.

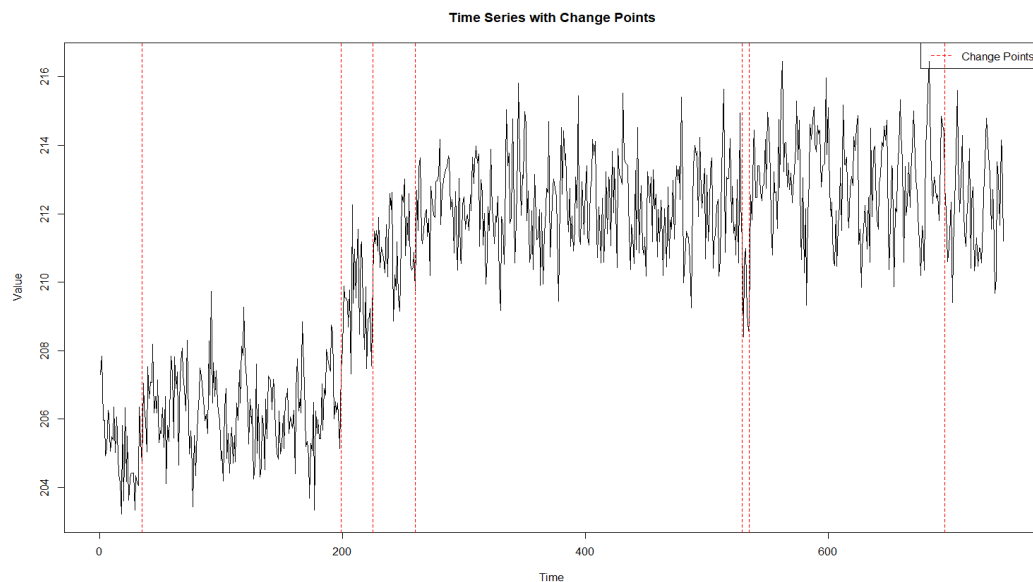


Figure 5.10: Plot illustrating the identified change points in the mean

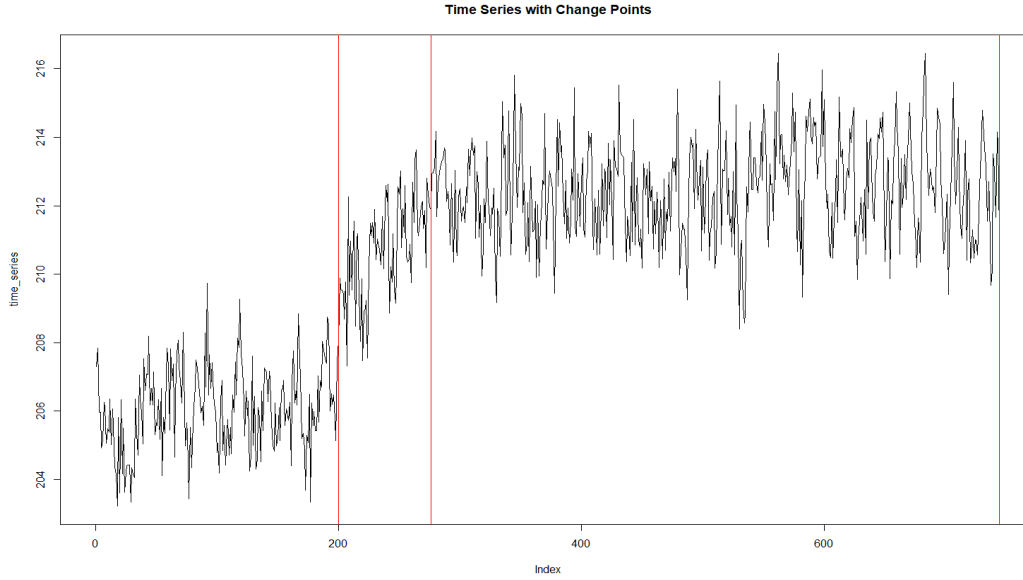


Figure 5.11: Plot illustrating the identified change points in the variance

Comparative Analysis

When we compared the results of Bayesian Change Point Analysis to those of our optimization-based approaches, we discovered significant disparities. Specifically, the quantity and position of identified change spots differed between the two methodologies. Furthermore, due to its sensitivity to errors, our optimization-based method may generate more change points than the Bayesian method. As a result, it is expected that our method will discover more change points than the Bayesian approach. The discrepancy is attributable to the essence of both approaches. Ours is not dependent on point distribution, whereas the others are based on data set statistics.

CHAPTER 6

REGRESSION WITH DEEP NEURAL NETWORKS

In this chapter, we delve into neural networks, specifically Deep Neural Networks (DNNs) for regression analysis. We present an overview of neural networks, emphasizing their significance and the evolution that led to the emergence of deep learning. We delve into the fundamentals and structure of DNNs, defining crucial concepts such as neurons, activation functions, and layers. Furthermore, we discuss different forms of activation functions and their implications for model training.

Throughout this chapter, we aim to provide readers with a comprehensive understanding of DNNs and their application in regression analysis. Furthermore, we will demonstrate these concepts using a practical case, comparing our discussion with the approaches explored previously in this thesis. This example demonstrates the efficacy of Deep Neural Networks in regression problems.

6.1 Brief Overview of Neural Networks

Neural networks are a type of machine learning algorithm based on the structure and function of the human brain. These computational models are made up of interconnected nodes, or neurons, these neurons are grouped into input, hidden, and output layers. Each neuron gets input signals that are weighted according to their connection strength. These weighted inputs are processed with an activation function to generate output signals. During training, the network changes these weights to reduce the discrepancies between expected and observed outputs, hence boosting data-driven learning. The structure allows for the effective detection of complicated patterns via interactions between neurons across layers. Weights in artificial neural networks (ANNs) represent the strength of neurons' connections, which is critical for the effective recognition of patterns and learning [47].

6.1.1 Significance of neural networks

- Neural networks are capable of modeling complicated data patterns.
- They can identify intricate patterns in data without relying on established relationships or assumptions.
- Neural networks perform at image recognition, natural language processing, and predictive analytics due to their flexibility.
- Neural networks can generalize and draw conclusions from previously learned patterns, allowing them to deliver significant insights and predictions.

6.1.2 Deep Neural Networks (DNNs)

Deep Neural Networks (DNNs) are neural networks with numerous hidden layers that excel in learning hierarchical data representations by abstracting higher-level characteristics from raw input. Their depth allows them to handle large, high-dimensional data and extract valuable insights, making them useful in various applications [48]. This thesis focuses on DNNs due to their superior performance and versatility in modeling complex data structures.

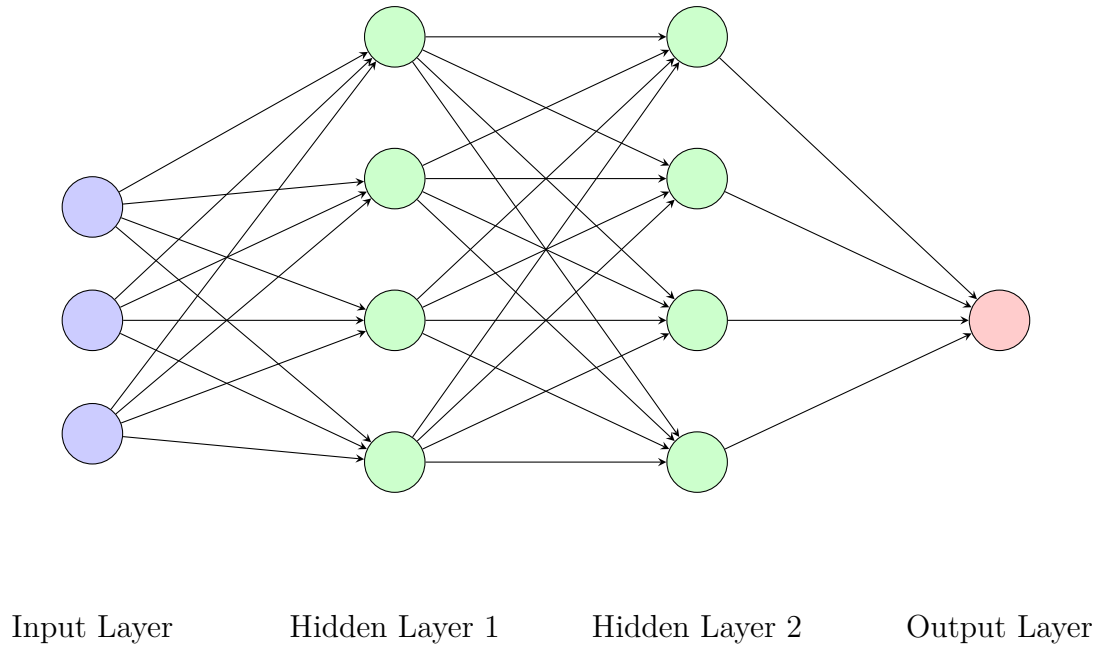


Figure 6.1: Deep neural network architecture

In regression analysis with neural networks, the mathematical formula represents the relationship between input features x to each layer and predicted output y from the same layer.

$$y = b + W^T x \quad (6.1)$$

In this equation 6.1, y represents the projected output and b represents the bias term. The coefficients (weights) w for each input characteristic x_i and the number of input items n are represented.

Evolution and Background History

Deep neural networks (DNNs) as we know them today were developed with significant contributions from the historical development of neural network research. An outline of neural network theory's development is provided in this section, starting with McCulloch and Pitts' foundational work in the 1940s [49]. The 1980s saw a rebirth of neural networks due to the development of backpropagation, a crucial multi-layer perceptron training approach [50]. Deep neural networks emerged in the 21st century, driven by breakthroughs in computer power and big datasets [51].

6.1.3 Activation functions

Activation functions are critical components of neural networks that introduce nonlinearity into the network's output. This nonlinearity enables neural networks to simulate complicated interactions between inputs and outputs, allowing them to learn and represent highly nonlinear patterns in data. Without activation functions, the network would return to a linear regression model, greatly limiting its expressive ability. Three of the most widely used activation functions in neural networks are Rectified Linear Unit (ReLU), sigmoid, and hyperbolic tangent.

The ReLU activation function is described as:

$$f(x) = \max(0, x)$$

ReLU is common because it is simple and effective in addressing the problem of vanishing gradients that might occur during deep neural network training.

Sigmoid activation function is defined as:

$$f(x) = \frac{1}{1 + e^{-x}}$$

The sigmoid function limits input values to the range $[0, 1]$, making it appropriate for binary classification problems that require the output to be interpreted as a probability.

Tanh's activation function is defined as follows:

$$f(x) = \frac{e^x + e^{-x}}{e^x - e^{-x}}$$

Tanh produces values between -1 and 1, which makes it symmetric around the origin.

Tanh is helpful in identifying non-linearities in the data, both positive and negative.

The choice of an activation function is determined by the task’s specific features and the network architecture, each function having advantages and limits.

6.2 Numerical Example

In this section, we provide a practical case study to demonstrate deep neural network concepts using the power consumption dataset. We discuss our findings considering the results of the previous method discussed in this thesis.

This example demonstrates the effectiveness of deep neural networks in addressing regression difficulties.

In the numerical example, we used an organized strategy consisting of the following steps:

- **Data preparation:** The dataset was divided into two sets: 70% for training and 30% for testing. This division enabled us to train the model on a subset of the data while evaluating its performance on previous data.
- **Neural Network Architecture:** Our neural network was methodically designed, with a focus on depth and complexity. We used a pyramid structure across **6 hidden layers**, starting with 64 neurons in the first layer and then halving the neuron count **for each subsequent layer (64, 32, 16,...)**. This design decision looked to gradually extract and abstract aspects from the input data as it moved deeper into the network.

- **Activation function:** (ReLU) activation function was used throughout the neural network. This non-linear activation function provides nonlinearity into the network, allowing it to describe complicated relationships within the data.
- **Implementation:** The neural network model was created using built-in Python functions and packages from PyTorch library. This library offers complete tools for rapidly designing, training, and assessing neural network models.

6.3 Discussion

This section discusses the results from our numerical example, including the predictions made by the neural network.

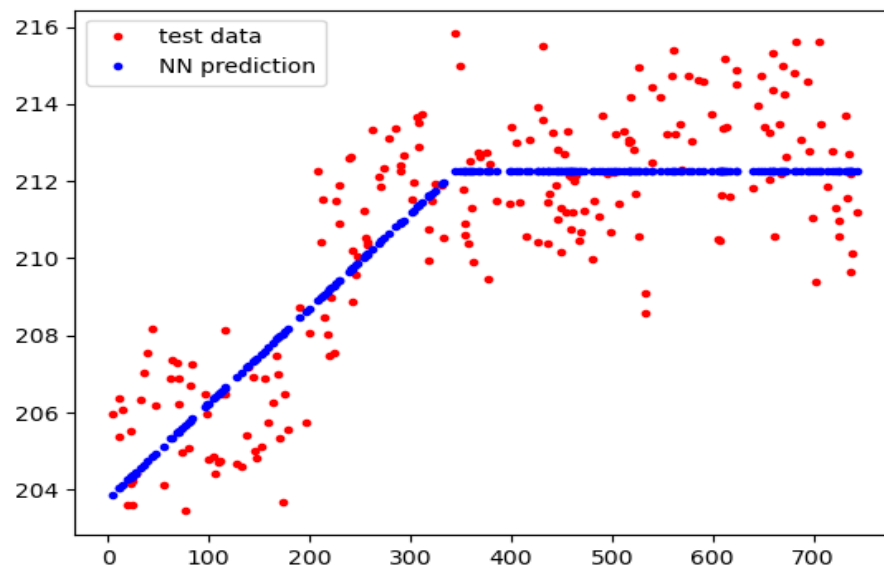


Figure 6.2: Predictions Made by Neural Network

Mean Squared Error (MSE) of Testing Data

The MSE of the testing data was found to be 2.9. Figure 6.3 illustrates the plot of absolute error for the testing data.

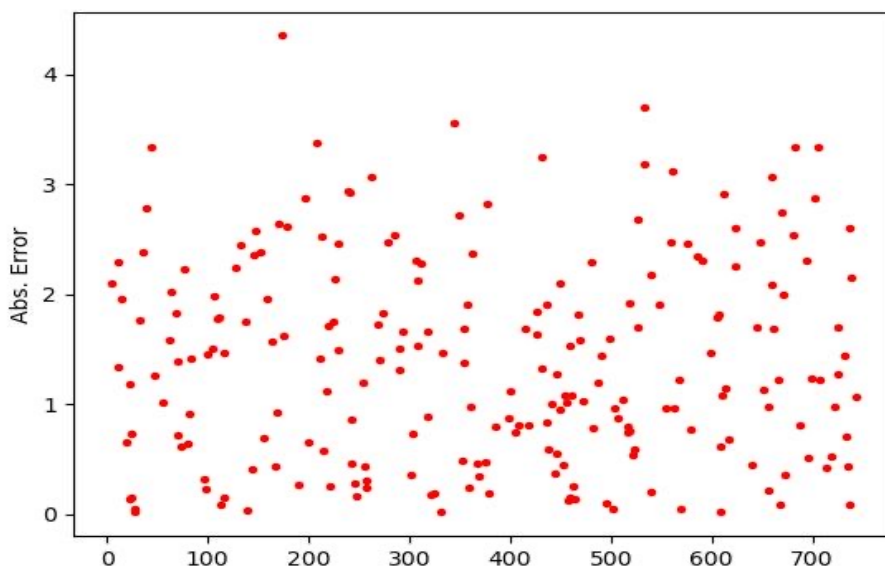


Figure 6.3: Absolute Error for Testing Data

Mean Squared Error (MSE) of Training Data

In the same way, 2.28 was determined to be the MSE of the set used for training. Figure 6.4 displays the training data's relative error plot.

Interpretation

- It's worth noting that our approach of partitioning the data, which focuses on minimizing error, yielded an **MSE of 1.14 for part 1 and 2.1 for part 2, for a total of 3.24**, which is less than the MSE of the global regression (**3.98**).

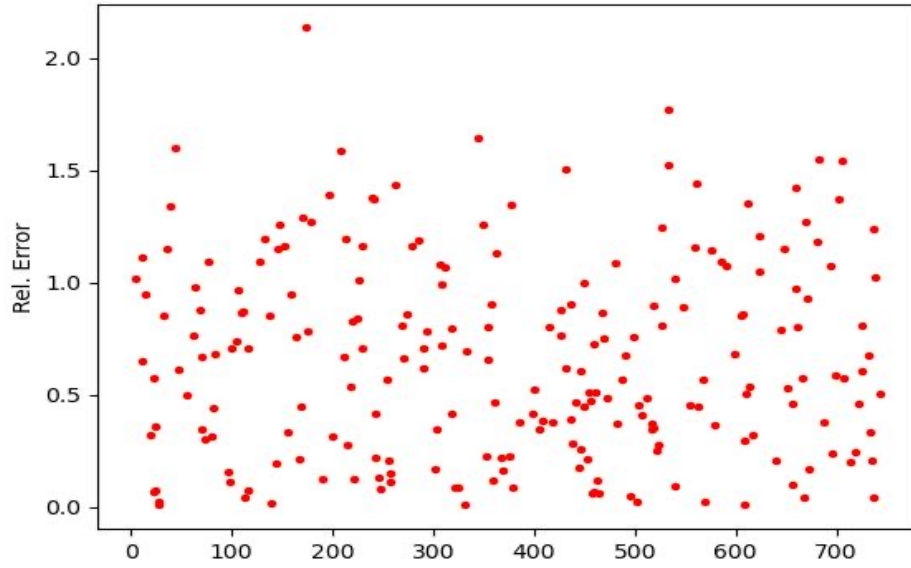


Figure 6.4: Relative Error for Training Data

Furthermore, the neural network approach produced similar findings, resulting in comparable accuracy. This demonstrates our method’s effectiveness and legitimacy, especially in scenarios where data partitioning is critical to enhancing model performance.

- The use of **neural networks serves as a validation tool** for our proposed method. By obtaining similar findings using the neural network methodology, we demonstrate our method’s credibility and efficiency in addressing the underlying problem.
- Another significant fact is that neural networks perform better on larger datasets. Our results show that the neural network method performed similarly to our approach, implying that **the neural network model thrives when confronted with a large amount of data.**

CHAPTER 7

CONCLUSION AND FUTURE WORKS

In this concluding chapter, we summarize the thesis's main objectives, findings, and contributions to capture the essence of our research. In addition, we make recommendations for future work and areas of investigation in the discipline to encourage further advanced.

7.1 Summary of findings

Everywhere in daily life, there are always changes. It is essential to understand unexpected and major changes in everyday life in order to understand scientific discoveries and decision-making. Decision-makers need to recognize and respond to data patterns, including anomalies, trends, and seasonality, as the world grows more interconnected. Unique statistical methods have been developed to address the limits of existing methodologies and challenges with change point detection. This

thesis provides and critically evaluates novel statistical techniques for change-point identification with the aim of explaining processes hidden behind observable scatter plots.

The first proposed method focuses on utilizing an optimization approach to identify change points and partition data in order to minimize errors using the Euclidean distance. When working with complex, clustered scatter plots, predictive modeling performs better because it is applicable to all approximation functions. Additionally, the second proposed approach offers a dynamic framework for handling changing provided data and generating continuous intervals. It operates on a dataset, regardless of finiteness, and considers the complex nature of real-time data evolution. To ensure effective local trends and patterns, the method uses an error threshold ε to detect change points and partition the dataset accordingly.

The application chapter uses methods for finding change points in real data sets (power consumption) and determining the Euclidean distance of local regression which is substantially less than the Euclidean distance of the global regression. For statistical purposes, the bootstrapping approach is employed, and the R software environment and statistical package are used for bootstrap analysis, from this approach, which involves proving that our findings are relevant to different datasets and do not depend on a specific one. When we applied Bayesian Change Point Analysis to the same data set at the end of the application chapter, we found several significant differences. More specifically, there were differences between the two

techniques in the number and location of detected change spots. Additionally, our optimization method may result in more change points than the Bayesian method because of its sensitivity to errors.

Using the same dataset, we evaluated findings in a real-world numerical example by utilizing Deep Neural Networks (DNNs), which we also discussed in the previous chapter.

7.2 Future works

As discussed in **Chapters 3 and 4**, our methods were designed to promote a variety of approximation families. Future studies could expand the algorithm’s capacity to handle various function types and investigate alternate partitioning methods. Due to time restrictions, our current implementation focuses on linear approximation functions. However, future iterations may widen the scope to cover other function types such as exponential or quadratic functions.

Based on the results of the neural network in **Chapter 6**, future research directions could include studying alternative activation functions or improving the network architecture to improve performance further.

APPENDIX A

R CODES - FIRST METHOD

One point

```
Linear <- function(x, y) {  
  a = cov(x, y) / var(x)  
  b = mean(y) - a * mean(x)  
  return(norm(y - (a * x + b), type = "2")^2)}  
  
phi <- function(x, y) {  
  A = cbind()  
  l = length(x)  
  for (i in 2:(l - 2)) {  
    A = cbind(A, c(x[i], Linear(x[1:i], y[1:i])  
      + Linear(x[(i + 1):l], y[(i + 1):l])))}  
  B = t(A)  
  min_error = min(B[, 2])  
  cutoff_point = B[which.min(B[, 2]), 1]
```

```

X1 = x[1:which.min(B[, 2])]
Y1 = y[1:which.min(B[, 2])]
X2 = x[(which.min(B[, 2]) + 1):l]
Y2 = y[(which.min(B[, 2]) + 1):l]
a1 = cov(X1, Y1) / var(X1)
b1 = mean(Y1) - a1 * mean(X1)
a2 = cov(X2, Y2) / var(X2)
b2 = mean(Y2) - a2 * mean(X2)

phi(x, y)}

```

Two points

```

Linear <-function(x,y){
  x=sort(x)
  a=(var(x)!=0)*cov(x,y)/var(x)+0*(var(x)==0)
  b=mean(y)-a*mean(x)
  return(norm(y-(a*x+b),type="2"))
  #print(norm(y-(a_s*x+b_s),type = "")^2/(1+a_s^2))}

phi<-function(x,y){
  x=sort(x)
  A=cbind()
  l=length(x)
  for (i in 3:(l-4)) {
    for(j in (i+1):l-2) {

```

```

A=cbind(A,c(x[i],x[j],Linear(x[1:i-1],y[1:i-1])+Linear(x[i:j],y[i:j])+
Linear(x[(j+1):l],y[(j+1):l]))) } }

B=na.omit(t(A))

k=which.min(B[,3])

print(c(B[k,1],B[k,2],B[k,3]))}

```

APPENDIX B

R CODES - SECOND METHOD

```
e <- 0.0007 # value error

d <- 0 # Initial value of d

# Define the Linear function

Linear <- function(x, y) {

  a = cov(x, y) / var(x), b = mean(y) - a * mean(x)

  return(norm(y - (a * x + b), type = "2"))}

cutoff <- function(x, y, e) {

  l = length(x), j = 1

  cutoff_points <- c(), while (j < l) {

    d = 0, i = j + 1

    while (d < e && i < l) {

      i = i + 1

      d = Linear(x[j:i], y[j:i]) }, j = j + i

    cutoff_points <- c(cutoff_points, x[i - 1])}

  return(cutoff_points)}
```

REFERENCES

- [1] I. H. Sarker, “Data science and analytics: an overview from data-driven smart computing, decision-making and applications perspective,” *SN Computer Science*, vol. 2, no. 5, p. 377, 2021.
- [2] J. Han, J. Pei, and H. Tong, *Data mining: concepts and techniques*. Morgan kaufmann, 2022.
- [3] B. Zhang, “Regression clustering,” in *Third IEEE International Conference on Data Mining*. IEEE, 2003, pp. 451–458.
- [4] S. Sharma, D. A. Swayne, and C. Obimbo, “Trend analysis and change point techniques: a survey,” *Energy, Ecology and Environment*, vol. 1, pp. 123–130, 2016.
- [5] V. A. Siris and F. Papagalou, “Application of anomaly detection algorithms for detecting syn flooding attacks,” *Computer communications*, vol. 29, no. 9, pp. 1433–1442, 2006.
- [6] D. Liu, X. Chen, Y. Lian, and Z. Lou, “Impacts of climate change and human activities on surface runoff in the dongjiang river basin of china,” *Hydrological*

- Processes: An International Journal*, vol. 24, no. 11, pp. 1487–1495, 2010.
- [7] E. S. Page, “Continuous inspection schemes,” *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.
 - [8] M. Basseville, I. V. Nikiforov *et al.*, *Detection of abrupt changes: theory and application*. prentice Hall Englewood Cliffs, 1993, vol. 104.
 - [9] A. Miller, *Subset selection in regression*. CRC Press, 2002.
 - [10] J. Chen and A. K. Gupta, “Parametric statistical change point analysis: with applications to genetics, medicine, and finance,” 2012.
 - [11] Y.-C. Yao, “Estimating the number of change-points via schwarz’criterion,” *Statistics & Probability Letters*, vol. 6, no. 3, pp. 181–189, 1988.
 - [12] C.-B. Lee, “Estimating the number of change points in a sequence of independent normal random variables,” *Statistics & probability letters*, vol. 25, no. 3, pp. 241–248, 1995.
 - [13] J. Chen and A. K. Gupta, “On change point detection and estimation,” *Communications in statistics-simulation and computation*, vol. 30, no. 3, pp. 665–697, 2001.
 - [14] D. Siegmund, “Confidence sets in change-point problems,” *International Statistical Review/Revue Internationale de Statistique*, pp. 31–48, 1988.
 - [15] C. R. Loader, “A log-linear model for a poisson process change point,” *The Annals of Statistics*, pp. 1391–1411, 1992.

- [16] S. M. O'Brien, "Cutpoint selection for categorizing a continuous predictor," *Biometrics*, vol. 60, no. 2, pp. 504–509, 2004.
- [17] D. V. Hinkley, "Inference about the change-point in a sequence of random variables," 1970.
- [18] P. Haccou, E. Meelis, and S. Van De Geer, "The likelihood ratio test for the change point problem for exponentially distributed random variables," *Stochastic processes and their applications*, vol. 27, pp. 121–139, 1987.
- [19] H.-J. Kim and D. Siegmund, "The likelihood ratio test for a change-point in simple linear regression," *Biometrika*, vol. 76, no. 3, pp. 409–423, 1989.
- [20] M. Lavielle, "Detection of multiple changes in a sequence of dependent variables," *Stochastic Processes and their applications*, vol. 83, no. 1, pp. 79–102, 1999.
- [21] C. Truong, L. Oudre, and N. Vayatis, "A review of change point detection methods. arxiv," *arXiv preprint arXiv:1801.00718*, 2018.
- [22] H. Chernoff and S. Zacks, "Estimating the current mean of a normal distribution which is subjected to changes in time," *The Annals of Mathematical Statistics*, vol. 35, no. 3, pp. 999–1018, 1964.
- [23] I. BROEMELING, "Bayesian procedures for detecting a change in a sequence of random variables." 1972.
- [24] A. F. Smith, "A bayesian approach to inference about a change-point in a sequence of random variables," *Biometrika*, vol. 62, no. 2, pp. 407–416, 1975.

- [25] D. Barry and J. A. Hartigan, “Product partition models for change point problems,” *The Annals of Statistics*, pp. 260–279, 1992.
- [26] C. A. R. Diniz, L. A. Milan, and J. Mazucheli, “Bayesian inference of a linear segmented regression model,” *Brazilian Journal of Probability and Statistics*, pp. 1–16, 2003.
- [27] R. C. Tiwari, K. A. Cronin, W. Davis, E. J. Feuer, B. Yu, and S. Chib, “Bayesian model selection for join point regression with application to age-adjusted cancer rates,” *Journal of the Royal Statistical Society Series C: Applied Statistics*, vol. 54, no. 5, pp. 919–939, 2005.
- [28] R. M. Steward, S. E. Rigdon, and R. Pan, “A bayesian approach to diagnostics for multivariate control charts,” *Journal of Quality Technology*, vol. 48, no. 4, pp. 303–325, 2016.
- [29] L. Orváth and P. Kokoszka, “Change-point detection with non-parametric regression,” *Statistics: A Journal of Theoretical and Applied Statistics*, vol. 36, no. 1, pp. 9–31, 2002.
- [30] C. Zou, Y. Liu, P. Qin, and Z. Wang, “Empirical likelihood ratio test for the change-point problem,” *Statistics & probability letters*, vol. 77, no. 4, pp. 374–382, 2007.
- [31] D. S. Matteson and N. A. James, “A nonparametric approach for multiple change point analysis of multivariate data,” *Journal of the American Statistical Association*, vol. 109, no. 505, pp. 334–345, 2014.

- [32] Z. Harchaoui and O. Cappé, “Retrospective mutiple change-point estimation with kernels,” in *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*. IEEE, 2007, pp. 768–772.
- [33] W. R. Blischke *et al.*, “Least squares estimators of two intersecting lines,” 1961.
- [34] P. E. Ferreira, “A bayesian analysis of a switching regression model: known number of regimes,” *Journal of the American Statistical Association*, vol. 70, no. 350, pp. 370–374, 1975.
- [35] T. M. Jawa, “Statistical methods of detecting change points for the trend of count data,” 2017.
- [36] F. Ansari, K. Gray, D. Nathwani, G. Phillips, S. Ogston, C. Ramsay, and P. Davey, “Outcomes of an intervention to improve hospital antibiotic prescribing: interrupted time series with segmented regression analysis,” *Journal of Antimicrobial Chemotherapy*, vol. 52, no. 5, pp. 842–848, 2003.
- [37] F. Zhang, A. K. Wagner, S. B. Soumerai, and D. Ross-Degnan, “Methods for estimating confidence intervals in interrupted time series analyses of health interventions,” *Journal of clinical epidemiology*, vol. 62, no. 2, pp. 143–148, 2009.
- [38] R. Jiang, “Determination of degradation change point using spline function,” in *2012 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*. IEEE, 2012, pp. 900–904.

- [39] C. J. Curtis and G. L. Simpson, “Trends in bulk deposition of acidity in the uk, 1988–2007, assessed using additive models,” *Ecological Indicators*, vol. 37, pp. 274–286, 2014.
- [40] A. J. Scott and M. Knott, “A cluster analysis method for grouping means in the analysis of variance,” *Biometrics*, pp. 507–512, 1974.
- [41] L. Y. Vostrikova, “Detecting “disorder” in multidimensional random processes,” in *Doklady akademii nauk*, vol. 259, no. 2. Russian Academy of Sciences, 1981, pp. 270–274.
- [42] D. M. Hawkins, “Fitting multiple change-point models to data,” *Computational Statistics & Data Analysis*, vol. 37, no. 3, pp. 323–341, 2001.
- [43] E. M. Maboudou-Tchao and D. M. Hawkins, “Detection of multiple change-points in multivariate data,” *Journal of Applied Statistics*, vol. 40, no. 9, pp. 1979–1995, 2013.
- [44] S. Aminikhanghahi and D. J. Cook, “A survey of methods for time series change point detection,” *Knowledge and information systems*, vol. 51, no. 2, pp. 339–367, 2017.
- [45] M. Staudacher, S. Telser, A. Amann, H. Hinterhuber, and M. Ritsch-Marte, “A new method for change-point detection developed for on-line analysis of the heart beat variability during sleep,” *Physica A: Statistical Mechanics and its Applications*, vol. 349, no. 3-4, pp. 582–596, 2005.

- [46] M. Olteanu, F. Rossi, and F. Yger, “Challenges in anomaly and change point detection,” *arXiv preprint arXiv:2212.13520*, 2022.
- [47] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, *Machine Learning, Neural and Statistical Classification*, illustrated ed., ser. Ellis Horwood series in artificial intelligence. Ellis Horwood, 1994, digitized on 10 Jan 2018.
- [48] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [49] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The bulletin of mathematical biophysics*, vol. 5, pp. 115–133, 1943.
- [50] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [51] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

VITAE

- **Name:** DALAL ALI ALZHRANI

- **Nationality:** SAUDI

- **Date of Birth:** 23/04/1998

- **Email:** *alidalal2200@gmail.com*

- **Qualifications:**

B.S. degree in Mathematics from Imam Abdulrahman Bin Fasil University, Dammam, Saudi Arabia, May 2020.

M.S. degree in Applied Statistics from King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, May 2024.

- **Position:**

Graduate Assistant, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia.

- **Permanent Address:** P.O.Box 7312, Jubail Industrial City 35718, Saudi Arabia