Genomic Data Analysis: Unravelling the Complexity of Genetic Markers for Disease Prediction

By

Bimal Kilambu

220022956

Under the Guidance of

Dr. Rebecca Jeyavadhanam B, Ph.D., MPhil., MCA., FHEA

Lecturer, Department of Computer Science

A DISSERTATION REPORT

Submitted to the Department of Computer Science

in partial fulfillment of the requirements for the award of the degree of

MASTER OF SCIENCE

IN

COMPUTER SCIENCE

YORK ST JOHN UNIVERSITY

London Campus

United Kingdom

August - 2024

Declaration

I, Bimal Kilambu, hereby certify that the dissertation titled "Genomic Data Analysis: Unravelling the Complexity of Genetic Markers for Disease Prediction," submitted for the degree of MSc in Computer Science with Year in Industry at York St John University, is my original work. This work has not been previously submitted by me for a degree at this or any other academic institution.

All sources of information and literature utilized in this dissertation have been appropriately cited and acknowledged. This dissertation does not contain any material that has been submitted and accepted for the award of any other degree or diploma at any university or other institution.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to everyone who has supported and guided me throughout the journey of completing this project.

First and foremost, I would like to express my deepest gratitude to my parents. Their unwavering support, encouragement, and love have been the foundation upon which I have built my academic and personal life. Their sacrifices and belief in me have been a constant source of motivation, and I am forever indebted to them for everything they have done for me.

I am deeply grateful to my supervisor, **Dr. Rebecca Jeyavadhanam B, Ph.D., MPhil., MCA., FHEA.**, Lecturer, Department of Computer Science, for her invaluable guidance, support, encouragement, suggestions, and guidance throughout the development phases of this dissertation.

I would also like to thank all my friends and colleagues who have supported me in various ways, providing encouragement, advice, and inspiration.

Abstract

Genomic data analysis plays an important role in uncovering the intricate relationships between genetic markers and disease. This dissertation delves into the complexities of genetic markers, particularly focusing on their predictive potential in disease outcomes. The study utilizes data from The Cancer Genome Atlas to explore various genomic markers across ten types of cancer, employing descriptive statistics to elucidate underlying patterns. Subsequently, machine learning models like Extreme Gradient Boosting and Random Forest are trained and assessed to predict disease outcomes based on these markers. The research aims to achieve three primary objectives: first, to identify and describe patterns in genomic data using advanced statistical methods; second, to develop and evaluate the effectiveness of machine learning models for disease prediction; and third, to validate the efficacy of these models using rigorous evaluation metrics like accuracy, precision, and F1-score. The dissertation concludes with insights into the significance of genetic markers in disease prediction, implications for personalized treatment planning, and recommendations for future research directions. This research advances the field of genomics, offering potential implications for early disease detection, targeted therapies, and precision medicine approaches.

Table of Contents

List of Figures

List of Tables

Abbreviations

AI	Artificial Intelligence
CIGAR	Concise Idiosyncratic Gapped Alignment Report
CNNs	Convolutional Neural Networks
DL	Deep Learning
DNA	Deoxyribonucleic Acid
GWAS	Genome-wide Association Studies
KNN	K-Nearest Neighbors
ML	Machine Learning
SNPs	Single Nucleotide Polymorphisms
TCGA	The Cancer Genome Atlas
XGBoost	Extreme Gradient Boosting

Chapter 1: Introduction

Genomic data analysis has emerged as a cornerstone of modern biomedical research, offering unprecedented insights into the intricate relationship between genetic markers and disease. By leveraging vast repositories of genetic information, researchers can uncover patterns that elucidate disease susceptibility, progression, and treatment response. Genetic markers like Single Nucleotide Polymorphisms (SNPs) and structural variants, serve as signposts within the genome, offering valuable clues about disease risk and prognosis (Collins et al., 2003; Manolio, 2010).

Understanding the complexity of genetic markers is paramount for enhancing disease prediction accuracy and developing targeted interventions. This dissertation focuses on delving deep into genomic data analysis techniques, unravelling the intricate web of genetic markers associated with various diseases. By employing advanced computational techniques and machine learning (ML) algorithms, this research aims to discover and characterize genetic markers linked with disease. The findings are expected to improve the understanding of genetic determinants of disease and pave the way for more targeted diagnostic and therapeutic strategies.

1.1 Context

Genomics, the study of genomes, is a transformative field in modern medicine that offers profound knowledge about the genetic foundations of health and disease. By examining the complete set of Deoxyribonucleic Acid (DNA) within an organism, genomics allows researchers to identify variations and mutations that contribute to various medical conditions (Feero et al., 2010; Lander, 2011). Genetic markers, such as SNPs, insertions, and deletions, play a vital role to decipher the genetic basis of diseases. These markers can serve as indicators of susceptibility to certain illnesses, including various forms of cancer, enabling more precise disease prediction and personalized treatment strategies (Brookes, 1999; Kiezun et al., 2012). As cancer remains one of the major causes of death globally, understanding its genetic underpinnings is essential for early diagnosis and effective intervention.

Current research in genomic data analysis has significantly advanced our understanding of genetic markers associated with diseases. Large-scale projects like 1000 genomes projects, UK Bio Bank, and The Cancer Genome Atlas (TCGA) have provided extensive datasets that facilitate the identification of genetic variations linked to cancer (Sudlow et al., 2015; International Cancer

Genome Consortium, 2010). Numerous studies have leveraged these datasets to develop ML models for predicting disease risk based on genetic markers (Kourou et al., 2015; Libbrecht & Noble, 2015). However, despite these advancements, significant gaps reamin in the field. Many studies are limited by their focus on a narrow range of genetic markers or specific cancer types, leaving a broader understanding of the genetic landscape incomplete. Additionally, the complexity of genomic data poses challenges in developing models that can generalize across diverse populations. This study seeks to address these gaps by employing comprehensive data analysis techniques to uncover patterns in genetic markers, enhancing the accuracy and reliability of disease prediction.

1.2 This Study

This study is centered on analyzing genomic data to identify and understand genetic markers that can predict disease, with a primary focus on cancer. By leveraging advanced data analysis techniques and ML models, this research aims to unravel the complexities inherent in genomic data, specifically focusing on SNPs, insertions, and deletions. The primary objective is to build predictive models capable of accurately identifying individuals at risk of various cancers, thereby facilitating early diagnosis and personalized treatment strategies. Additionally, this study will use data visualization techniques to identify relevant data patterns to provide a clearer understanding of their role in disease prediction.

The scope of this research is confined to the analysis of publicly available genomic data, with a particular emphasis on cancer-related datasets from TCGA. This study will encompass a variety of cancer types, focusing on the genetic variations that contribute to their development. The data will include detailed information on genetic marker types, variant types, chromosome locations, and the mutations observed, such as A>C, G>T etc. substitutions. By concentrating on these specific elements, the study aims to generate insights that are both broad in application and deep in detail. The research will involve rigorous data preprocessing, feature selection, and the deployment of several ML models to evaluate their effectiveness in predicting cancer risk. Additionally, descriptive statistical analysis will be performed to uncover data patterns, enriching the understanding of genetic markers and their predictive capabilities. Through this focused approach, the study seeks to address existing gaps in the literature and contribute to the field of

genomic data analysis by enhancing our understanding of genetic markers and their predictive capabilities.

1.3 Significance of the Study

This research is pivotal for the field of genomics as it explores the complex relationship between genetic markers and disease prediction. By focusing on cancer, one of the leading global causes of mortality, the study leverages advanced data analysis techniques to explore the intricate patterns of genetic variations. By analyzing large-scale genomic data, this study enhances our knowledge of how genetic variations contribute to cancer development and progression. This research not only enriches the scientific community's understanding of genomics but also sets a foundation for future studies and innovations in the field.

The potential impact of this study is substantial. By identifying key genetic markers associated with cancer, the findings can improve early detection and risk assessment, ultimately leading to better patient outcomes. Predictive models developed through this research could be integrated into clinical workflows, providing healthcare professionals with powerful tools for personalized medicine. This could lead to more accurate and timely interventions, tailored treatment plans, and reduced healthcare costs. Additionally, the study's insights could influence public health strategies and policymaking, emphasizing the importance of genomic research in combating cancer and other genetic diseases.

1.4 Research Aim

The aim of this research is to explore and analyze genomic data to unravel the intricate relationships between genetic markers and disease susceptibility. By examining large-scale genomic datasets, the study aims to identify and characterize significant genetic markers associated with various diseases. Utilizing advanced computational techniques and ML algorithms, the goal is to develop precise predictive models to support early detection and personalized treatment strategies in oncology.

1.5 Research Objectives

The objectives of this study are as follows:

1. Identify and describe patterns in genomic data using descriptive statistics, enhancing understanding of the distribution and traits of genetic markers.

2. Develop, implement, and evaluate multiple ML model performances for disease prediction using genomic data.
3. Validate the predictive efficacy and reliability of the predictive models through extensive evaluation using metrics such as accuracy, precision, recall, F1-score, and confusion matrices.

1.6 Dissertation Structure

This dissertation is structured as follows:

1. **Chapter 1: Introduction -** introduces the research topic by discussing the background of genomic data analysis and the significance of genetic markers in disease prediction. This chapter has also stated the research aim and objectives that guide this study.
2. **Chapter 2: Literature Review -** reviews existing literature on genomic data analysis, focusing on genetic markers and their association with disease. This chapter discusses previous studies and identifies gaps in knowledge that this research aims to address.
3. **Chapter 3: Methodology -** details the research design and approach used to achieve the research objectives, encompassing data collection methods, data preprocessing steps, ML models employed, and evaluation metrics utilized. Ethical considerations guiding the research process are also discussed within this chapter and justifies the rationale behind the chosen methodologies.
4. **Chapter 4: Result Analysis and Discussion -** presents and analyzes the findings from the genomic data analysis, focusing on the identification of patterns in genetic markers and the evaluation of ML model performance. It provides interpretations of the results in relation to existing literature and research hypotheses.
5. **Chapter 5: Conclusion -** summarizes the main findings of the study and their significance for the fields of genomics and disease prediction. It discusses the research's challenges and limitations and recommends future research directions based on the study outcomes.

Chapter 2: Literature Review

The field of genomics experienced substantial progress over the past few decades, driven by technological innovations and increasing accessibility to large-scale genomic data (Lander, 2011; Collins & Varmus, 2015). Genomic data analysis has become a cornerstone of modern biomedical research, offering profound insights into the genetic underpinnings of complex diseases. This chapter provides an in-depth review of the current literature on genomic data analysis, focusing on the identification and application of genetic markers for disease prediction, with an emphasis on cancer.

Genetic markers, including SNPs, copy number variations, and other genomic changes, play a pivotal role in understanding disease mechanisms and building predictive models for disease susceptibility and progression (Collins et al., 2003; Manolio et al., 2009). Although there have been significant improvements, the complexities of analysis and interpretation of genomic data remain fraught with challenges. These include managing the sheer volume of data, ensuring data quality, integrating multi-omic data types, and addressing ethical considerations related to data privacy and security (Ritchie et al., 2015). Furthermore, there is a need for more robust methodologies to improve the accuracy and reliability of disease prediction models.

This literature review will explore the fundamental concepts and methodologies of genomic data analysis, discuss the role of genetic markers in disease prediction, and examine recent advancements in ML applications in this field. Additionally, it will highlight recent trends, ongoing challenges, and the gaps that exist in current research. By synthesizing existing knowledge, this review will lay the groundwork for the subsequent chapters, which will detail the methodology and findings of this study on genomic data analysis and disease prediction.

2.1 Genomics and Genetic Markers

2.1.1 Fundamentals of Genomics

Genomics is the study of an organism's entire DNA sequence, known as the genome. In humans, the genome includes information for 22 pairs of chromosomes plus one pair of sex chromosomes, totaling roughly 3 billion base pairs (Ham et al., 2020). In each chromosome, DNA base pairs are arranged in a sequence where each pair is represented by one of four symbols: A, T, C, or G, which correspond to adenine, thymine, cytosine, and guanine, respectively. Genomic analysis involves

comparing these sequences against a reference genome to identify variations, which can provide critical insights into genetic predispositions and disease mechanisms.

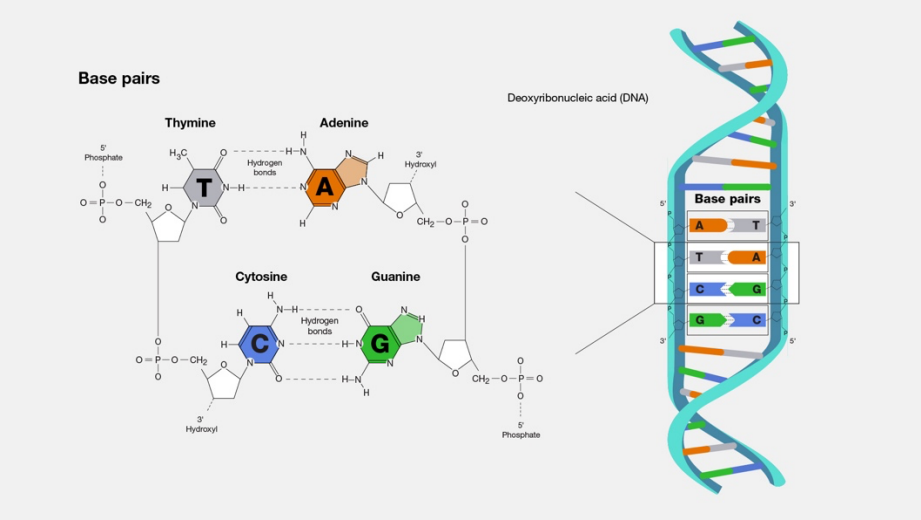


Figure 1: DNA base pair

In the preprocessing stage of genomic analysis, aligned read data plays a key role and includes details such as the chromosome identifier, the position where the read begins on the chromosome, the base pair sequence, and the quality scores. The CIGAR string, which stands for Concise Idiosyncratic Gapped Alignment Report, outlines alignment data with a list of integer-operation pairs, indicating operations such as matches, insertions, deletions, and soft-clippings (Ham et al., 2020).

The rapid advancement in DNA sequencing technologies has led to a tremendous growth in genomic data, projected to reach 40 exabytes by 2025 (Zheng et al., 2016). This massive volume of data can significantly aid disease diagnosis by allowing for similarity-based comparisons between patients' genomic sequences, enabling more accurate diagnoses and tailored treatments (Zheng et al., 2016). Genomic data is particularly valuable as it contains biologically meaningful information, such as an individual's risk of diseases like Alzheimer's, cancer, and schizophrenia (Ki, 2017).

2.1.2 Types of Genetic Markers

Genetic markers are segments of DNA with known locations that can be used to identify individuals or species and to associate inherited traits with specific regions of the genome. Among the most commonly used genetic markers are SNPs, which involve a change in a single nucleotide

at a particular position in the genome (Sinecen, 2019). SNPs are abundant across the genome and are particularly valuable for predicting genetic merit and assessing disease risk.

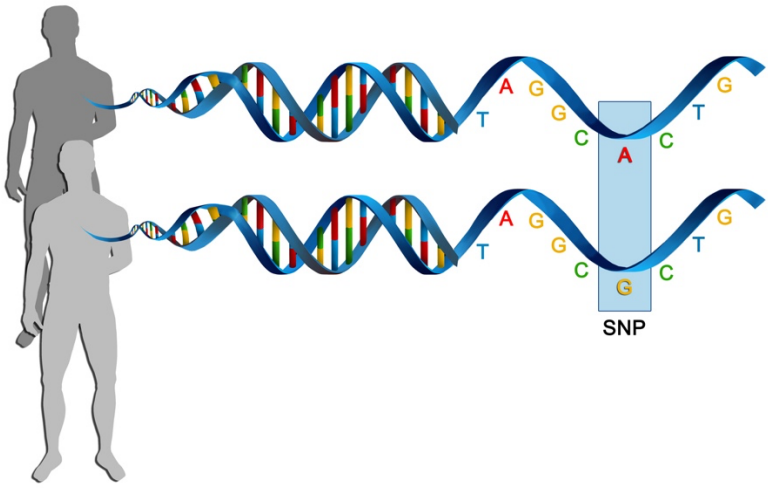


Figure 2: SNP marker

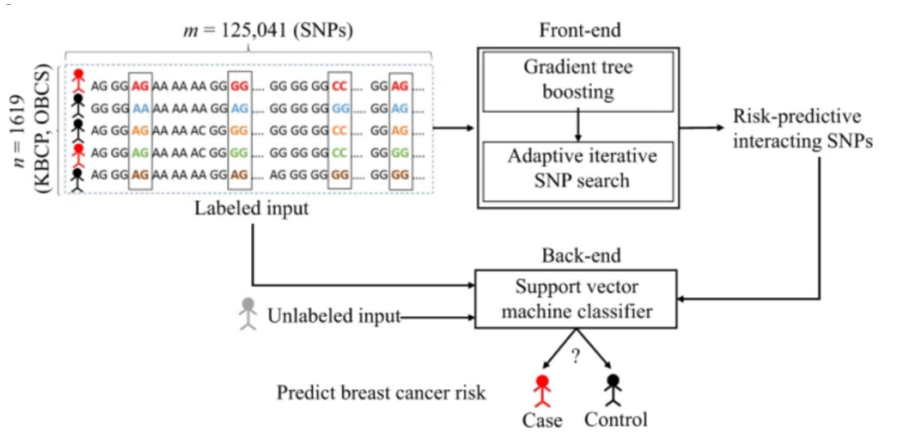
Genetic variations can be classified into three main types: insertions, deletions, and SNPs (Senadheera & Weerasinghe, 2016). The impact of these mutations varies depending on their attributes, especially if they occur within protein-coding regions, potentially altering protein structure and function. This necessitates a process called data enrichment, where mutation files are supplemented with additional information to enhance analysis accuracy. Many datasets, sequenced with older genome build, require extensive metadata analysis to update and refine the data (Senadheera & Weerasinghe, 2016).

Genomic data inherently holds significant biological information regarding an individual's risk for diseases. By examining a person's genetic information, latent genetic diseases like Alzheimer's, cancer, and schizophrenia can be uncovered, underscoring the importance of comprehensive genomic analysis in modern medicine (Ki, 2017).

2.2 Disease Prediction Using Genetic Markers

Genetic markers are crucial for predicting and diagnosing diseases, providing valuable information about an individual's risk for various conditions. Genome-wide association studies (GWAS) have traditionally been used to identify genetic biomarkers, including SNPs, linked to a wide range of traits and diseases (Silva et al., 2022). By analyzing the frequency of these SNPs in affected versus unaffected individuals, GWAS can identify genetic variations that contribute to disease risk.

Genomic data analysis entails the computational processing of enormous datasets from DNA sequencing. With billions of nucleotide bases, each human genome provides a comprehensive blueprint of an individual's specific traits, susceptibilities, and disease risks (Jones, 2024). The role of bioinformatics is crucial here, as it develops and employs computational tools and algorithms to manage, interpret, and extract meaningful insights from large datasets, helping to identify genetic variations associated with diseases.



Source: https://www.nature.com/articles/s41598-018-31573-5

Figure 3: Breast cancer prediction using SNPs.

The increasing volume of genomic data, driven by advancements in DNA sequencing technologies, underscores the need for efficient data processing pipelines. These pipelines typically include stages such as sequence alignment, data conversion, and advanced analysis (Liu et al., 2015). The analysis of a single human genome, with approximately 3 billion base pairs, traditionally takes considerable computational time, highlighting the necessity for optimized algorithms and high-performance computing resources.

Mutation data analysis is a critical objective in genomic data analysis, particularly in the context of cancer research. Cancer, being a genetic disease, originates from accumulated mutations within specific organs (Senadheera & Weerasinghe, 2016). Analyzing SNPs and other genetic variations can provide insights into the genesis of cancer and other diseases, helping to guide the design of targeted therapeutic approaches. However, the process is labor-intensive and requires meticulous identification and interpretation of relevant mutations.

The introduction of artificial intelligence (AI) methods, such as ML and deep learning (DL), into genomic data analysis, has significantly bolstered the ability to manage large-scale and complex

datasets. AI techniques are increasingly being used to transform large genomic datasets into clinically actionable knowledge, forming the basis of precision medicine (Xu et al., 2018). For instance, AI models can predict disease risks based on genetic information, aiding in early diagnosis and improved prognosis.

Genomic data is at the forefront of personalized healthcare, helping to identify genetic factors that contribute to various diseases. However, incorporating genomic data into healthcare systems encounters several challenges, including social, ethical, legal, educational, economic, and technical issues (Al Kawam et al., 2018). Despite these challenges, the potential of genomic data to enhance disease detection, diagnosis, and treatment is immense.

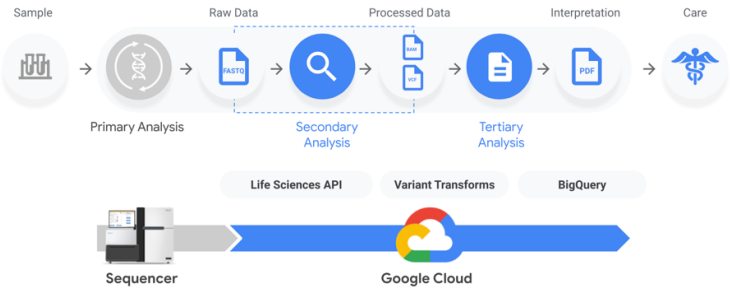
For specific diseases, genetic markers play a vital role in prediction and diagnosis. For instance, in kidney disease, biomarkers like cystatin C and creatinine are vital for developing predictive scores used by clinicians (Widen et al., 2021). Similarly, SNPs can help forecast quantitative values of biomarkers, aiding in the assessment of common disease risks. AI and ML models have demonstrated potential in early disease diagnosis across various conditions, such as breast cancer, diabetes, and Alzheimer's disease (DeGroat et al., 2024). Biomarkers, ranging from genetic mutations to histopathological images, are essential for cancer prognosis, with histopathology considered the gold standard due to its ability to provide detailed morphological attributes related to cancer aggressiveness (Shao et al., 2020).

In breast cancer, mutations in the BRCA1 and BRCA2 genes are linked to a significant proportion of inherited cases. Early diagnosis through genetic screening can markedly increase survival rates (Gancheva & Borovska, 2020). ML techniques have become essential tools for medical researchers, enabling the discovery and identification of models in complex datasets. These techniques allow for the effective prediction of disease outcomes, facilitating personalized treatment plans and improving patient care (Gancheva & Borovska, 2019).

2.3 Genomic Data Analysis Techniques

Genomic data collection involves acquiring raw genetic material through advanced sequencing technologies, producing vast amounts of data that require meticulous preprocessing. This includes sequence alignment against reference genomes using tools like BWA and Bowtie (Li & Durbin, 2009; Langmead & Salzberg, 2012). Subsequent steps involve quality control and variant calling using tools such as GATK and Samtools to detect genetic variations (McKenna et al., 2010; Li et

al., 2009). The preprocessing phase is essential for maintaining the accuracy and reliability of downstream genomic analyses (Liu et al., 2015; Zheng et al., 2016).



Source: Google Cloud

Figure 4: Genomic data analysis flow.

Genomic analysis typically begins with sequence alignment, where DNA sequences are aligned against a reference genome to identify variations (Ham et al., 2023). This process involves aligning read data, which includes chromosome identifiers, base pair sequences, and quality scores. The CIGAR string provides a summary of alignment information, indicating operations such as matched (M), inserted (I), deleted (D), or soft-clipped (S) bases (Ham et al., 2023).

Following sequence alignment, the next stage is data conversion, which transforms raw sequencing data into formats suitable for advanced analysis. This stage includes error correction, data normalization, and variant calling to identify SNPs and other genetic variations (Liu et al., 2015). The advanced analysis stage involves more complex computational techniques to derive meaningful insights from the data.

Statistical techniques are crucial for recognizing genetic variants associated with diseases. GWAS utilize statistical models such as logistic regression and Bayesian analysis to correlate genetic variations with disease phenotypes (Visscher et al., 2017; Evangelou & Ioannidis, 2013). Using tools like DESeq2 and edgeR, differential expression analysis identifies changes in gene expression across conditions, which supports the discovery of potential biomarkers (Anders & Huber, 2010; Love et al., 2014). These methods are pivotal in elucidating the genetic underpinnings of complex diseases (Wu et al., 2018).

ML algorithms enhance predictive modelling in genomics. Supervised learning methods like random forests and support vector machines integrate genomic features for disease classification (Xu et al., 2020; Ching et al., 2018). Deep models such as convolutional neural networks (CNNs) are used to analyze large-scale genomic datasets, extracting intricate patterns for personalized medicine applications (Min et al., 2017; Angermueller et al., 2016). ML models integrate diverse genomic features to enhance predictive accuracy and uncover novel insights into disease mechanisms (DeGroat et al., 2024; Auti et al., 2023).

Deep learning, a branch of ML, employs neural networks with multiple layers to capture intricate patterns in genomic data. Techniques like CNNs and recurrent neural networks (RNNs) are prevalent in healthcare applications, enhancing early disease diagnosis and risk prediction (Deepa et al., 2024). Additionally, clustering algorithms like K-means are used for unsupervised learning, grouping similar genomic sequences to identify novel genetic markers (Dongel and Timar, 2017).

Effective visualization of genomic data is essential for interpreting complex patterns and communicating findings. Tools like Circos plots visualize genomic structural variants and chromosomal rearrangements (Krzywinski et al., 2009). Heatmaps and principal component analysis (PCA) plots provide insights into gene expression profiles and sample clustering (Wang et al., 2019). Interactive platforms such as UCSC Genome Browser and Integrative Genomics Viewer (IGV) enable dynamic exploration of genomic annotations and variants (Thorvaldsdottir et al., 2013; Robinson et al., 2011).

2.4 Current Trends and Challenges

2.4.1 Recent Advances in Genomic Data Analysis

Recent years have witnessed significant advancements in genomic data analysis, driven by technological breakthroughs and methodological innovations. AI and ML techniques are increasingly employed to analyze vast genomic datasets. DL algorithms, among these methods, are highly accurate in extracting complex patterns and predictive biomarkers from high-throughput genomic data (Wu et al., 2018).

There is an increasing trend towards combining data across various omics layers (genomics, transcriptomics, proteomics), leading to a deeper understanding of disease mechanisms and personalized medical approaches (DeGroat et al., 2024).

Advances in single-cell sequencing technologies have enabled detailed profiling of individual cells, uncovering cellular heterogeneity and rare cell types that are pivotal in disease progression (Stuart & Satija, 2019).

Genomic research increasingly relies on big data analytics and cloud computing to manage and analyze large-scale genomic datasets efficiently. These technologies enhance scalability and computational capabilities, crucial for handling the complexities of genomic data (Phogat & Kumar, 2022).

2.4.2 Challenges and Limitations

Despite these advancements, genomic research faces several challenges that impede its full potential. The handling of large-scale genomic data involves substantial computational challenges. High-dimensional datasets and the need for intensive computational resources strain traditional analysis methods, necessitating the use of parallel computation environments and optimized algorithms (Dongel & Timar, 2017).

Analyzing large-scale genomic datasets requires advanced computational techniques and high-performance computing infrastructures. Addressing computational challenges such as scalability, algorithm efficiency, and data storage remains crucial for extracting actionable insights from genomic data (Schadt et al., 2010).

Integrating heterogeneous data from various sources remains a major hurdle. Standardizing data formats and developing interoperable tools are essential to ensure data quality and facilitate meaningful cross-study comparisons (Aledhari et al., 2021)

Genomic research raises ethical concerns regarding data privacy, consent, and equitable data sharing. It is essential to balance data accessibility for research with protecting individual privacy rights (Mohammed et al., 2021).

Overfitting remains a challenge in predictive modelling using genomic data, especially because of limited sample sizes and the complexity of biological variability. Rigorous validation using independent datasets is essential to ensure the robustness and generalizability of predictive models (Widen et al., 2021).

Transmission bottlenecks and slow data transfer speeds hinder the efficient sharing of large genomic datasets between research institutions. Improving data transmission technologies and

enhancing network infrastructures are critical to expediting collaborative genomic research efforts (Aledhari et al., 2021).

Addressing the gap between genomic research findings and their clinical applications involves overcoming challenges such as validation of biomarkers, integration into clinical workflows, and demonstrating clinical utility across diverse populations (DeGroat et al., 2024).

2.5 Gaps in Existing Research

Despite the advancements in genomic data analysis, several gaps and limitations remain evident in current research. Existing literature lacks comprehensive comparative studies between DL approaches and traditional methods in genomic data analysis. Wu et al. (2018) highlight that only a few studies have attempted to assess whether DL significantly outperforms conventional approaches in disease prediction using genomic data. Moreover, the evaluation metrics used, such as accuracy, often do not consider the complexities of imbalanced datasets, thus necessitating more robust comparative analyses (Wu et al., 2018).

The application of DL models like auto-encoders in genomic research suffers from a lack of standardized guidelines for selecting architecture parameters such as hidden layers and nodes. Wu et al. (2018) note that while these parameters are mentioned in studies, there is a dearth of justification and guidance on their selection, which hinders reproducibility and generalizability across different genomic datasets.

The optimization methods like grid search and random search suggested to enhance model performance in genomic data analysis, remain underexplored and inadequately studied. These methods fail to fully mitigate issues like over-fitting and under-fitting, which are critical in ensuring robust and reliable predictive models (Wu et al., 2018).

Genomic data collection is inherently expensive, limiting the availability of large, independent testing datasets essential for validating predictive models. This challenge is compounded by the imbalance in genomic datasets, where training and testing datasets often do not adequately represent the diverse genetic backgrounds required for accurate predictions (Wu et al., 2018).

There exists a significant gap between genomic research advancements and clinical practice. Ahmad et al. (2018) discuss how the underutilization of genetic tests in clinical settings is exacerbated by insufficient training among physicians and inadequate communication within

healthcare teams regarding the availability and utility of genetic tests. This gap highlights the need for enhanced educational programs and accessible genetic consultation services to bridge the divide between genomic research findings and practical clinical application (Ahmad et al., 2018).

2.6 Summary

In summary, this literature review has delved into the complexities and advancements in genomic data analysis for disease prediction. It has explored current trends, highlighting the rise of ML and DL techniques as powerful tools for extracting predictive genetic markers. Challenges such as data dimensionality, overfitting, and the integration of diverse datasets have been identified as critical, emphasizing the critical need for robust methodologies and validation frameworks. The review also identified gaps in research, including the underutilization of advanced computational techniques and the limited application of DL in genomic studies. Overall, this review provides a comprehensive foundation for understanding the complexities and potential of genomic data analysis in predicting disease outcomes.

Moving forward, the methodology chapter will build upon these insights by outlining the approach to be taken in this dissertation. Specifically, it will detail the methods and techniques employed to analyze genomic data, address the identified challenges, and validate predictive models. By leveraging the theoretical foundations and empirical findings synthesized in this literature review, the methodology chapter aims to contribute to advancing our understanding of genetic markers for disease prediction.

Chapter 3: Research Methodology

This chapter outlines the methodology employed to analyze genomic data for predicting various cancer types. The primary objective is to find significant genetic markers and assess the effectiveness of different ML models in disease prediction. To achieve this, a systematic approach was adopted, starting with data collection from TCGA, followed by data preprocessing, feature selection, and the implementation of various ML models.

The methodology is crucial in achieving the research objectives as it provides a structured approach to exploring and analyzing the genetic markers associated with cancer. By employing robust data preprocessing techniques and a variety of ML models, this chapter aims to uncover patterns and insights that can contribute to more accurate and reliable disease prediction. This systematic approach ensures the reliability and validity of the findings, ultimately enhancing our understanding of the complexity of genetic markers in cancer prediction.

This comprehensive methodology aims to provide a transparent and reproducible framework for understanding genetic markers in disease prediction.

3.1 Research Design

The research design for this study focuses on utilizing genomic data analysis to predict cancer diseases by identifying complex genetic markers. This study employs a quantitative research approach, leveraging ML techniques to examine large datasets and uncover patterns and relationships within the data.

Figure 5: Research design

First, we collected a large dataset of genetic marker and disease association information from TCGA, a publicly available and reliable source (Weinstein et al., 2013). This dataset was chosen for its comprehensive coverage of genomic alterations across various cancer types.

Next, we visualized the data using descriptive statistics to identify patterns and trends in genetic markers, such as the prevalence of specific SNPs and insertion/deletion markers. This step was crucial for understanding the distribution and significance of different genetic alterations (Kourou et al., 2015).

The data were then preprocessed, which included cleaning, handling missing values, and converting the data into a format suitable for analysis. We removed columns with a high percentage of null values and performed feature selection to retain only the most relevant variables.

After preprocessing, a range of ML models—such as Random Forest, Extreme Gradient Boosting (XGBoost), Decision Tree, K-Nearest Neighbors (KNN), and Naive Bayes—were trained on the cleaned data. These models were selected for their proficiency in handling complex and high-dimensional data, and their effectiveness was measured using accuracy, precision, F1-score, and confusion matrix.

Finally, the models' results were validated using separate test datasets to ensure their generalizability and robustness. This comprehensive approach ensures that the findings are reliable and applicable to real-world scenarios.

3.2 Research Philosophy

The research philosophy underpinning this study is primarily positivist, as it focuses on empirical evidence and measurable data to understand and predict cancer disease patterns based on genetic markers. Positivism emphasizes observation and experimentation, aligning well with our objective to derive insights from extensive genomic datasets and to build predictive models using statistical and ML techniques.

Our approach is a mixed-methods approach, collecting numerical data from the reliable and well-curated source, TCGA. This approach aligns with the positivist paradigm, which values quantifiable data and objective analysis. This data includes genetic markers, mutation types, and their occurrences across various cancers, providing a solid foundation for our analysis. The utilization of ML models like Random Forest, XGBoost, Decision Tree, KNN, and Naive Bayes

supports this positivist stance, as these models rely on mathematical algorithms and statistical validation to make predictions.

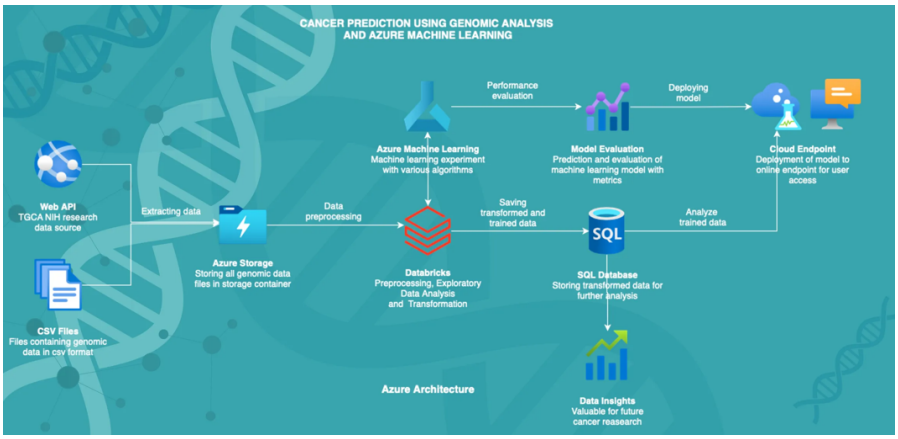
This research philosophy ensures a systematic investigation of genetic markers and their association with cancer. By adhering to the principles of positivism, we ensure that our methodology is rigorous, reproducible, and capable of generating reliable insights that can contribute to the field of genomic data analysis and cancer prediction.

3.3 Research Strategy

The research strategy employed in this dissertation involves a mixed-methods approach, integrating both quantitative and computational methods to analyze genomic data for disease prediction.

- **Quantitative Approach:** The study utilizes a quantitative approach to analyze large-scale genomic data obtained from TCGA. This involves the collection of numerical data on genetic markers, mutations, and associated cancer types across a diverse dataset. Descriptive statistics are employed to summarize and visualize patterns in the data, revealing insights into the prevalence and distribution of genetic markers linked to various cancers.
- **Computational Methods:** ML models play a key role in this research strategy. Models such as Random Forest, XGBoost, Decision Tree, KNN, and Naive Bayes are implemented to predict cancer diseases based on genetic markers. These models are trained with the genomic data collected and assessed using metrics such as accuracy, precision, and F1-score. To ensure robustness and generalizability, cross-validation techniques are employed, which help in addressing potential biases and avoiding overfitting.

By combining quantitative analysis with computational modelling, this research strategy aims to comprehensively unravel the complexity of genetic markers in disease prediction. The mixed-methods approach enables a thorough exploration of genetic variations and their implications for cancer susceptibility and prognosis. This strategy ensures that findings are both statistically rigorous and practically applicable, contributing valuable insights to genomic data analysis and its clinical implications.



Source: www.medium.com

Figure 6: Architecture for genomic analysis

3.4 Timeline

This research is projected to span over a period of 6 months, from data collection to final analysis and reporting. The timeline is structured to accommodate sequential stages including data acquisition from TCGA, data preprocessing, model development, evaluation, and validation. Each phase is allotted specific durations to ensure thoroughness and accuracy in analysis. The Gantt chart below illustrates the planned timeline for the research activities:

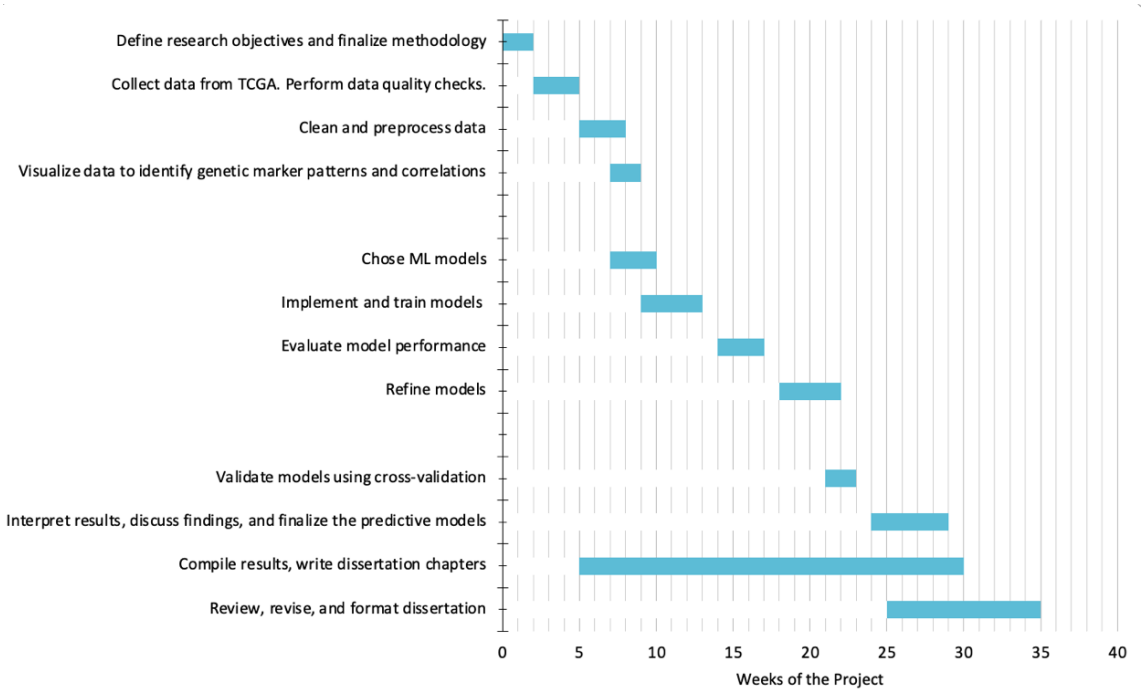


Figure 7: Gantt chart of project from data collection to final analysis and reporting

3.5 Data Collection

3.5.1 Source of Data

The primary source of data for this research is the TCGA database, a comprehensive and publicly accessible resource managed by the U.S. National Institutes of Health. The TCGA project provides extensive pan-cancer somatic mutation data, crucial for understanding genetic markers associated with various cancers (Weinstein et al., 2013). The dataset includes detailed information on genetic marker types, variant types, chromosomal occurrences, start and end positions of chromosomes where markers are noticed, and mutated DNA bases such as A>C and G>T. Additionally, the dataset also provides information about the types of cancer diseases related to these genetic mutations.

3.5.2 Data Extraction

The process of data extraction involved using tools and software to download relevant genetic marker data from the UCSC Genome Browser's Table Browser tool (https://genome.ucsc.edu/cgi-bin/hgTables). The steps for extracting data were as follows:

1. Open the Table Browser page from the UCSC Genome Browser website.

Figure 8: Interface of UCSC genome browser's table browser

2. Select the required data tables for 10 different types of cancer.
3. Download the resulting data files in CSV format for each cancer type.
4. Aggregate and consolidate the data files to create a comprehensive dataset for analysis.

3.5.3 Data Characteristics

The collected data includes diverse genetic marker types such as SNPs, insertions, and deletions. It provides detailed information on genetic marker types, variant types, chromosomal occurrences, start and end positions of the chromosomes, mutated DNA bases, and related cancer diseases.

In total, the dataset consisted of 446,582 records and 37 data fields, offering a rich source of information for analyzing the genetic complexity associated with cancer.

3.6 Data Preprocessing

3.6.1 Data Cleaning

To maintain data quality and consistency, the dataset was carefully cleaned:

- Missing or incorrect values were handled by converting placeholders (e.g., "--") to null values.
- Columns with substantial null data were removed to enhance dataset integrity. The following table summarizes the columns removed and the corresponding number of null records:

Column Name	Number of null records
dbsnp_rs	81074
dbsnp_val_status	392721
days_to_death	303162
cigarettes_per_day	268477
weight	180568
alcohol_history	303371
alcohol_intensity	446582
bmi	192071
years_smoked	417133
height	191255
ethnicity	12

Table 1: Number of null records of each column

- Duplicate records were checked but not found in the dataset, ensuring data quality, and preventing potential redundancy and bias.

- All column names were converted to lowercase for uniformity and ease of reference. Some Column names were adjusted to improve clarity and alignment.

Column Name	Renamed Column Name
#"chrom"	chrom
project_id'	cancer_type
'variant_classification	variant
matched_norm_sample_barcode	barcode

Table 2: Renamed column name

3.6.2 Feature Selection

Feature selection is an essential part of data preprocessing, focusing on identifying and keeping the most relevant variables that contribute to accurate disease prediction while removing those that are redundant or insignificant (Guyon & Elisseeff, 2003; Chandrashekar & Sahin, 2014). To focus on relevant information for analysis, the following steps were taken:

- Insignificant columns that did not contribute significantly to the analysis were removed. These included: 'case_id', 'reserved', 'blockcount', 'score', 'strand', 'chromstarts', 'samplecount', 'tumor_sample_barcode', 'entrez_gene_id'.
- Only columns deemed essential for genetic marker analysis were retained. The selected columns are: chrom, chromstart, chromend, name, thickstart, thickend, blocksizes, hugo_symbol, freq, variant, variant_type, tumor_seq_allele1, tumor_seq_allele2, reference_allele, gender, cancer_type, barcode.
- After conducting descriptive statistical analysis, the barcode and reference_allele column were removed before model training. The barcode column, which served as a unique identifier for patients and reference_allele column was removed to streamline the dataset further and eliminate any non-essential data.

3.6.3 Data Transformation

Data transformation methods were applied to prepare the dataset for ML algorithms:

- Label encoding was applied to categorical variables to convert them into numeric format.
- Data scaling using StandardScaler normalized the numeric features, ensuring consistent ranges and mitigating the impact of varying scales on model performance.

3.7 Data Visualization

The study relied heavily on data visualization to discover patterns and extract meaningful insights from the genomic data. Descriptive statistics were visualized using various graphical techniques to elucidate relationships and trends among genetic markers and cancer types. Bar graphs were employed to illustrate the distribution of different types of genetic markers across chromosomes, highlighting the prevalence of SNPs, insertions, and deletions. Heatmaps provided a visual representation of correlations between genetic markers and disease types, aiding in the identification of potential biomarkers.

3.8 Machine Learning Models

ML models were employed to predict disease outcomes based on genetic markers. This section details the selection, implementation, and training of various models for the analysis.

3.8.1 Model Selection

Five ML models were chosen for this study. The choice of ML models was based on their suitability for handling genomic data and predicting disease outcomes:

- **Random Forest Classifier:** Known for its ensemble approach and proficiency in handling complex datasets.
- **Extreme Gradient Boosting:** Optimized for high performance and efficiency, suitable for large-scale genomic datasets.
- **Decision Tree Classifier:** Provides interpretable results and insights into feature importance.
- **Naïve Bayes Classifier:** Efficient for probabilistic classification tasks.
- **K-Nearest Neighbors Classifier:** Utilized for its simplicity and effectiveness in pattern recognition tasks.

3.8.2 Model Implementation

The implementation of the selected models was carried out using Python programming language and relevant libraries (e.g., scikit-learn, XGBoost):

- Each model was instantiated with default or optimized hyperparameters, such as n_neighbors=5, metric='minkowski', and p=2 for the KNN classifier, and

objective='multi:softmax', num_class=20, n_estimators=10, and random_state=42 for the XGBoost classifier, to maximize performance.

- Data preprocessing steps, including cleaning, feature selection, and transformation, were integrated into the implementation pipeline to ensure data compatibility.

3.8.3 Model Training

The training of the ML models comprised the following steps:

- The dataset, after preprocessing, was split into 70% training and 30% testing sets, using stratified sampling to retain the class distributions.
- Models were trained on the training set using appropriate algorithms and hyperparameters.
- Cross-validation methods, like k-fold cross-validation, were applied to evaluate model generalization and prevent overfitting.
- Performance metrics like accuracy, precision, and F1-score were utilized to evaluate and compare the effectiveness of each model in predicting diseases.

3.9 Evaluation Metrics

To evaluate the predictive performance of the ML models in disease prediction, several key metrics were utilized:

- **Accuracy:** Represents the overall prediction correctness, determined by the proportion of correctly predicted cases out of the total instances.
- **Precision:** Represents the percentage of true positives out of all positive predictions, indicating the model's exactness in identifying positive instances.
- **F1-score:** Represents the harmonic mean of precision and recall, providing a balanced metric that accounts for both false positives and false negatives.
- **Confusion Matrix:** Provides a detailed summary of classification model performance, including counts of true positives, true negatives, false positives, and false negatives.

3.10 Validation and Testing

To ensure the robustness and generalizability of our ML models for disease prediction, rigorous validation and testing procedures were employed.

3.10.1 Cross-Validation

To evaluate the model's performance, k-fold cross-validation was used, which involves dividing the dataset into k subsets (folds), training the model on k-1 folds, and validating it on the remaining fold (James et al., 2013). This process is repeated k times, with each fold serving as the validation set once, which helps estimate the model's performance on new data and mitigates the risk of overfitting.

3.10.2 Hyperparameter Tuning

Hyperparameters of the ML models were carefully selected and optimized to enhance their predictive performance. For the KNN classifier, the hyperparameters n_neighbors=5, metric='minkowski', and p=2 were chosen to define the model's behavior with respect to its nearest neighbors and distance metric. The XGBoost classifier was configured with objective='multi:softmax', num_class=20, n_estimators=10, and random_state=42, ensuring it operates optimally for multi-class classification tasks with 20 classes. Additionally, cross-validation techniques were applied to assess and validate the models' performance robustly across different subsets of the training data. This iterative process involved partitioning the dataset into k subsets and rotating through each fold to estimate the model's effectiveness in predicting unseen data, thereby reducing the risk of overfitting, and ensuring reliable model generalization.

3.10.3 Testing

The final evaluation of model performance was conducted on an independent test set. This approach simulates real-world conditions where the model deals with new, unseen data. Performance metrics like accuracy, precision, and F1-score were computed on this test set to evaluate the model’s generalization capabilities.

3.11 Software and Tools

3.11.1 Programming Languages and Libraries

The analysis and modelling for this study were conducted using Python, a versatile programming language ideal for data analysis and ML tasks (Perez & Granger, 2007). The work was carried out using Jupyter Notebook. Key libraries utilized include:

- **Pandas:** To handle genomic data efficiently, Pandas was used, providing powerful DataFrames for data manipulation and analysis (McKinney, 2010).

- **NumPy:** Essential for numerical operations and array manipulations, supporting the underlying computations required for data preprocessing and model training.
- **Matplotlib and Seaborn:** Employed for data visualization, facilitating the creation of plots and graphs to highlight patterns and connections within the dataset.
- **Scikit-learn:** A robust ML library that offers resources for data preprocessing, model selection, evaluation metrics, and cross-validation (Pedregosa et al., 2011).
- **XGBoost:** A powerful gradient-boosting library renowned for its efficiency and performance in managing complex datasets (Chen & Guestrin, 2016).
- **Other utilities:** Includes tools for handling warnings, shuffling data, and importing specific functionalities required for model evaluation and interpretation.

3.11.2 Data Processing Tools

Various data processing techniques like label encoding, standard scaling, and stratified sampling were implemented using Scikit-learn's preprocessing and utility functions. These tools were instrumental in preparing the genomic data for model training and evaluation.

3.11.3 Computational Resources

The computational resources for this study primarily involved a personal laptop, which was adequate for running the Python scripts and conducting the analyses. The use of cloud-based platforms like Google Colab provided additional computational power when required, particularly for more intensive tasks such as hyperparameter tuning and cross-validation.

3.12 Ethical Considerations

Ethical considerations are paramount in genomic data analysis, particularly when dealing with sensitive information and potential implications for individuals and populations (Knoppers & Chadwick, 2005). This study adheres to the ethical guidelines and principles outlined below:

3.12.1 Data Privacy and Confidentiality

- **Informed Consent:** Data sourced from public repositories like TCGA were anonymized and obtained with proper consent from participants.
- **Confidentiality:** Strict measures were employed to protect the identity of individuals in the dataset. Personally identifiable information was not included or disclosed in any analysis or publication.

- **Data Handling:** All data handling procedures followed ethical standards, ensuring that data were used solely for research purposes and in compliance with relevant regulations.

3.13 Limitation

While every effort has been made to ensure the rigor and validity of this study, several limitations should be acknowledged. Firstly, it exclusively utilized data from TCGA, potentially limiting the generalizability of results to broader populations not represented in this dataset. Secondly, the analysis focused on a selection of 10 specific types of cancer, potentially overlooking variations in genetic markers across other cancer types. Data preprocessing involved the exclusion of fields with significant missing values, which may have impacted the completeness and comprehensiveness of the dataset used for modelling. Furthermore, the study evaluated a limited set of five ML models, omitting DL approaches that could potentially capture more intricate genomic relationships. Additionally, computational resources were limited to a personal laptop, which may have constrained the scalability and complexity of the analyses conducted. Although these limitations exist, measures have been taken to address biases and uphold the rigor and validity of the research results.

3.14 Summary

This chapter elucidated the methodology employed to analyze genomic data for disease prediction through the exploration of genetic markers. The study utilized the TCGA database to extract genetic data across ten types of cancer, focusing on SNP variants, insertions, and deletions. Data preprocessing ensured quality and relevance through cleaning, feature selection, and transformation. Five ML models were implemented and assessed based on accuracy, precision, and F1-score metrics. The study adhered to ethical guidelines, considered methodological limitations such as dataset constraints and computational resources, and utilized Python-based tools for data analysis and visualization. This comprehensive approach enabled the extraction of valuable insights into genetic marker complexities in disease prediction, laying the groundwork for the subsequent Result Analysis and Discussion chapter where we present and interpret the findings derived from our genomic data analysis.

Chapter 4: Result Analysis and Discussion

This chapter offers an in-depth analysis of the findings from our study on genomic data analysis focused on predicting cancer disease outcomes. The primary objective of this study is to evaluate the effectiveness of various ML models in predicting diverse cancer outcomes based on genomic data and to uncover patterns within genetic markers through advanced visualization techniques. The analysis is structured into several key sections: Descriptive Statistics, Model Comparison, Visualization of Results, Result Analysis, Limitations, and Conclusion.

The ML models implemented in this study comprise Random Forest, KNN, Decision Trees, Naive Bayes, and XGBoost. These algorithms were chosen due to their diverse approaches to classification and their proven effectiveness in handling complex datasets. As an ensemble method, Random Forest improves accuracy in predictions by aggregating the results from multiple decision trees (Breiman, 2001). KNN is renowned for its straightforward approach and effectiveness in non-linear data patterns (Altman, 1992). Decision Trees provide clear interpretability and are capable of processing handle both numerical and categorical data (Quinlan, 1986). Naive Bayes is valued for its efficiency and performance in high-dimensional datasets (Hand & Yu, 2001). Finally, XGBoost, a robust boosting algorithm, is renowned for its high performance and scalability in large datasets (Chen & Guestrin, 2016).

The chapter begins with Descriptive Statistics, offering a summary of the dataset, including the distribution of genetic markers. The Model Comparison section will showcase the performance metrics of each ML model, emphasizing their strengths and limitations. The Visualization of Data and Results section will offer graphical representations to facilitate the understanding of patterns within the genetic markers. Result Analysis will interpret the findings from the model performances and visualizations. The Limitations section will address the constraints encountered during the study. Finally, the Conclusion will provide a summary of the key findings and their significance.

4.1 Descriptive Statistics

We explore the genomic dataset utilized in our study, sourced from the TCGA project conducted by the U.S. National Institutes of Health, focusing on its descriptive statistics. The goal is to present

a detailed analysis of the dataset's composition, highlighting the distribution and frequency of various genetic markers, such as SNPs, insertions, and deletions. Understanding these distributions is crucial for identifying patterns and anomalies that may contribute to disease prediction. This analysis covers the total number of patients and cancer records, the breakdown of training and test datasets, and the chromosomal distribution of genetic markers. By understanding the detailed characteristics of these genetic markers, we uncover critical patterns and trends within the data that inform the subsequent analysis and discussions in this chapter.

4.1.1 Dataset Overview

The dataset used in this study comprises a total of 2,417 patients, providing a substantial pool of 446,582 records. This dataset includes genomic data from 10 distinct types of cancer, sourced from TCGA. The cancers represented in this dataset are:

- ACC: Adrenocortical Carcinoma
- BLCA: Bladder Urothelial Carcinoma
- CESC: Cervical and Endocervical Cancer
- DLBC: Diffuse Large B Cell Lymphoma
- ESCA: Esophageal Carcinoma
- HNSC: Head and Neck Squamous Cell Carcinoma
- KICH: Kidney Chromophobe
- LAML: Acute Myeloid Leukemia
- LGG: Brain Lower Grade Glioma
- PCPG: Pheochromocytoma and Paraganglioma

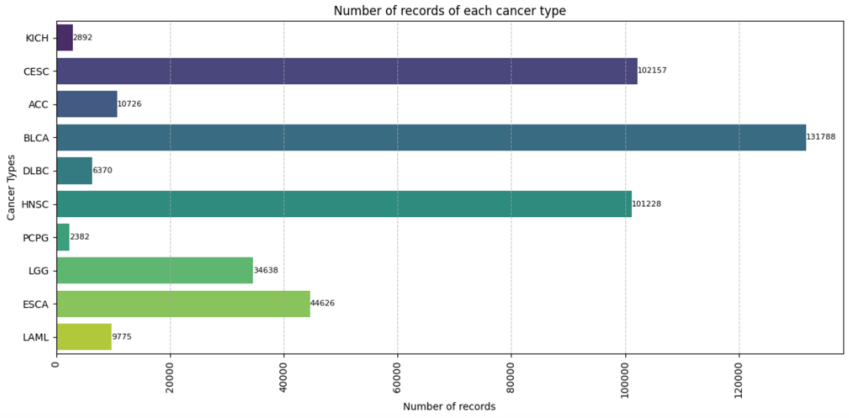


Figure 9: Bar graph of number of records of each cancer type

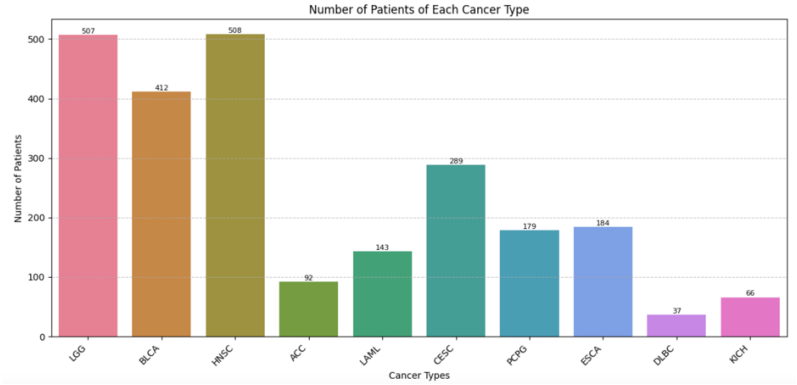


Figure 10: Bar graph of number of patients of each cancer type

To ensure a robust analysis and reliable model performance, the data was divided into two sets: 70% for training and 30% for testing. This split allows the models to learn from a significant portion of the data while still having a separate, unseen dataset to validate their predictions.

4.1.2 Distribution of Genetic Markers

This study's dataset offers insights into the distribution of three major types of genetic markers across various cancer types.

- **Single Nucleotide Polymorphisms:** SNPs constitute the majority of the genetic markers in this dataset, with a total of 422,702 records. SNPs are the most frequent type of genetic variation, characterized by single nucleotide changes in the DNA sequence. These variations are pivotal for understanding how genetic differences can influence disease susceptibility and progression.

- **Deletions (DELs):** The dataset includes 14,073 deletion records. DELs entail the removal of nucleotide sequences from the genome, potentially influencing gene function and disease progression mechanisms.
- **Insertions (INSs):** There are 9,807 insertion records in the dataset. Insertions occur when extra nucleotides are appended to the DNA sequence. Like deletions, insertions can disrupt normal gene function and are important for identifying novel genetic variations that may be implicated in cancer.

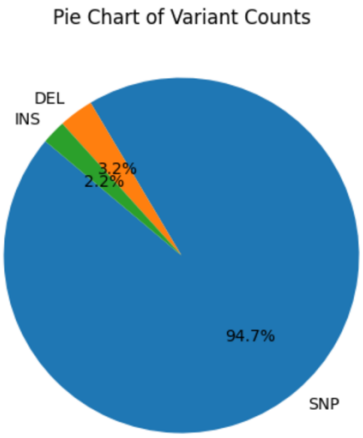


Figure 11: Pie chart of genetic markers type

The pie chart illustrates the composition of genetic markers in the dataset used for cancer analysis. SNP data dominates with 94.7%, indicating the prevalence of single nucleotide variations across the genome. DEL data accounts for 3.2%, representing deletions of nucleotide sequences, while INS data constitutes 2.2%, indicating insertions within the genome. This distribution underscores the significance of SNP analysis in genetic studies, with deletions and insertions also playing essential roles in understanding genomic mutations and their potential implications in disease prediction.

4.1.3 Variant Distribution

In examining the distribution of genetic marker variants across the dataset, a total of 18 distinct variants were identified. Among these, the top three variants by count include Missense_Mutation with 232,872 occurrences, Silent mutations totaling 90,862, and variants located in the 3'UTR region amounting to 31,678. Missense_Mutation involves alterations that change the encoded amino acid, potentially influencing protein function. Silent mutations do not affect the amino acid

sequence due to redundancy in the genetic code, while variants in the 3'UTR region impact mRNA stability and translational regulation.

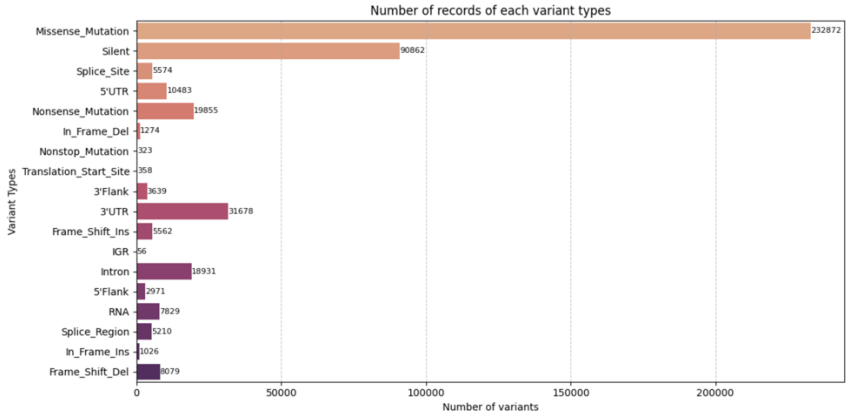


Figure 12: Bar graph of number of records of each variant types

4.1.4 Chromosomal Distribution

The dataset comprises genetic marker records spanning 24 different chromosomes. Chromosome 1 (chr1) exhibits the highest occurrence of genetic markers, totaling 45,966 records, followed by chromosome 2 (chr2) with 31,915 records and chromosome 19 (chr19) with 30,478 records. In contrast, chromosome 21 (chr21) and the Y chromosome (chrY) have the fewest genetic markers, with 4,404 and 161 records, respectively. This distribution underscores varying levels of genomic data availability across chromosomes, which may influence the interpretation and prediction accuracy of disease-related genetic markers.

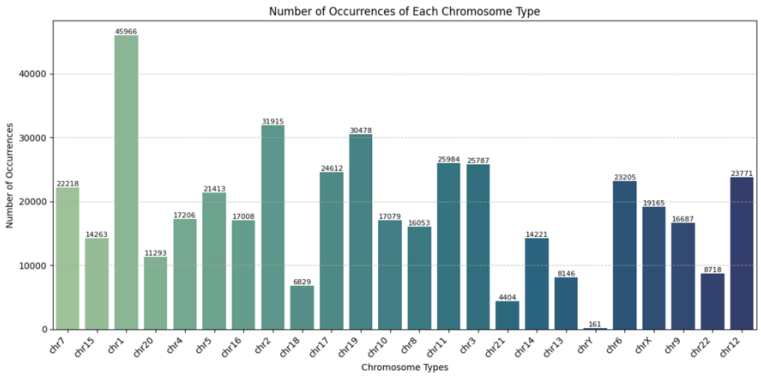


Figure 13: Bar graph of number of occurrences of each chromosome types

4.1.5 Gender Distribution

The dataset analyzed for this study comprises genetic marker records from a total of 446,570 individuals, categorized by gender. Among these records, 238,376 are from male individuals, while 208,194 are from female individuals.

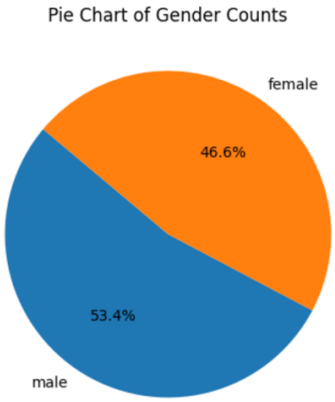


Figure 14: Pie chart of gender distribution

4.1.6 Single Nucleotide Polymorphisms

The dataset includes a total of 422,702 records of SNPs. These SNPs exhibit various variant classifications, with the most prevalent being Missense_Mutation (232,828), followed by Silent mutations (90,862), and variants located in the 3'UTR region (28,210). Other significant classifications include Nonsense_Mutation (19,213), Intron (17,467), 5'UTR (9,813), RNA (7,477), Splice_Region (5,033), Splice_Site (4,965), 3'Flank (3,296), and 5'Flank (2,806). Less frequently observed variants include Translation_Start_Site (358), Nonstop_Mutation (318), and intergenic regions (IGR) (56).

Variant	**Number of records**
Missense_Mutation	232828
Silent	90862
3'UTR	28210
Nonsense_Mutation	19213
Intron	17467
5'UTR	9813
RNA	7477

Splice_Region	5033
Splice_Site	4965
3'Flank	3296
5'Flank	2806
Translation_Start_Site	358
Nonstop_Mutation	318
IGR	56

Table 3: Number of records of each variant of SNPs

The distribution of SNPs by nucleotide changes reveals that the most common mutations are C>T (108,998), indicating a substitution of thymine (T) for cytosine (C), and G>A (107,800), indicating a substitution of adenine (A) for guanine (G). These are followed by C>G (40,608) and G>C (39,418) mutations. These variations highlight the prevalence and diversity of genetic mutations within the dataset, which are crucial for understanding their potential implications in disease prediction and genetic predisposition studies.

Mutation type	**Number of records**
C>T	108998
G>A	107800
C>G	40608
G>C	39418
G>T	28662
C>A	28363
T>C	19407
A>G	18516
T>A	8599
A>T	8207
T>G	7124
A>C	7000

Table 4: Number of records of each mutation type of SNPs

The distribution of SNPs markers across chromosomes shows that chromosome 1 has the highest count with 43,514 occurrences, followed by chromosome 2 with 30,231, and chromosome 19 with

29,034. Chromosome Y has the lowest count with just 145 occurrences. Notably, chromosome 11 and chromosome 3 also have significant occurrences with 24,719 and 24,403 respectively. This distribution shows that SNP markers are not uniformly distributed across chromosomes, with some chromosomes like 1, 2, and 19 having a much higher density of SNP markers compared to others like Y, 21, and 18.

Chromosome	Number of records
chr1	43514
chr2	30231
chr19	29034
chr11	24719
chr3	24403
chr22	8189
chr13	7641
chr18	6472
chr21	4179
chrY	145

Table 5: Number of top 5 and bottom 5 chromosomes occurrence in SNPs

4.1.7 Insertion Markers

Insertion markers in the dataset encompass a total of 9,807 records, revealing a diverse array of sequence variations. Among the identified insertions, insA (adenine insertion) is the most frequent with 1,588 instances, followed closely by insT (thymine insertion) with 1,527 occurrences, insC with 726 occurrences, and insG (guanine insertion) with 697 occurrences. The dataset also includes less frequent but notable insertions such as insTT (78 occurrences) and complex sequences like insAGCCTCAGCTGTTGTCATTGGGGCTCACTCAAACGTATGAGGCA (1 occurrence).

Mutation type	Number of records
insA	1588
insT	1527
insC	726

insG	697
insTT	78
insAGAG	1
insAATAATAAATATTCA	1
insGCATCAAAGAACATCTTCA	1
insAGCCTCAGCTGTTGTCATTGGGGCTCACTCAAACGTATGAGGCA	1
insAAGATGTATT	1

Table 6: Number of records of top 5 and bottom 5 insertion mutation type

The classification of insertion variants underscores their functional impact within the genome. Frame_Shift_Ins is the predominant classification, accounting for 5,562 instances, which indicates the insertion or deletion of nucleotides that alter the reading frame. Other classifications include variants in 3'UTR (1,157 occurrences), In_Frame_Ins (1,026 occurrences), Intron (711 occurrences), and Nonsense_Mutation (622 occurrences), among others. These classifications provide insights into the structural and regulatory consequences of insertions across genomic regions.

Variant	**Number of records**
Frame_Shift_Ins	5562
3'UTR	1157
In_Frame_Ins	1026
Intron	711
Nonsense_Mutation	622
5'UTR	264
RNA	139
3'Flank	112
Splice_Region	105
Splice_Site	63
5'Flank	45
Nonstop_Mutation	1

Table 7: Number of records of each variant of insertion

The distribution of Insertion markers across chromosomes shows varying occurrences, with chromosome 1 having the highest count at 1,009 markers, followed by chromosome 2 with 731 and chromosome 6 with 578. Chromosome Y has the fewest Insertion markers, totaling only 7. This data indicates a disparate distribution of Insertion markers across different chromosomes, highlighting variations in genomic regions prone to insertional events.

Chromosome	Number of records
chr1	1009
chr2	731
chr6	578
chr3	570
chr17	568
chr13	217
chr20	211
chr18	168
chr21	109
chrY	7

Table 8: Number of records of top 5 and bottom 5 chromosomes occurrence in insertion marker

4.1.8 Deletion Markers

Deletion markers within the dataset comprise a total of 14,073 records, showcasing a range of genomic alterations. Among the identified deletions, delC leads with 2,107 occurrences, closely followed by delT (2,076), delA (2,049), and delG (2,032). Additionally, there are less frequent deletions such as delAA (230 occurrences) and more complex deletions like delGGCCCAGCAGCCGCCTGCGGCTGGACGTCTCCA (1 occurrence).

Mutation type	Number of records
delC	2107
delT	2076
delA	2049
delG	2032
delAA	230

delGCAGGCTCCCCGCGGGCCCCATG	1
delGGCTTCACCGAGGAGTCC	1
delGTTTTCTGCGCAAGTTAG	1
delGGAACACCCAAACTAAATTTGT	1
delGGCCCAGCAGCCGCCTGCGGCTGGACGTCTCCA	1

Table 9: Number of records of top 5 and bottom 5 deletion mutation type

The classification of deletion variants highlights their impact on genomic structure and function. Frame_Shift_Del is the most prevalent classification, accounting for 8,079 instances, indicating deletions that alter the reading frame of genes. Other classifications include deletions in 3'UTR (2,311 occurrences), In_Frame_Del (1,274 occurrences), Intron (753 occurrences), and Splice_Site (546 occurrences), among others. These classifications provide insights into the regulatory and structural consequences of deletions across different genomic regions.

Variant	**Number of records**
Frame_Shift_Del	8079
3'UTR	2311
In_Frame_Del	1274
Intron	753
Splice_Site	546
5'UTR	406
3'Flank	231
RNA	213
5'Flank	120
Splice_Region	72
Missense_Mutation	44
Nonsense_Mutation	20
Nonstop_Mutation	4

Table 10: Number of records of each variant of deletion

The distribution of Deletion markers across chromosomes exhibits varying frequencies, with chromosome 1 containing the highest count at 1,443 markers, followed by chromosome 2 with 953 and chromosome 19 with 897. Chromosome Y shows the lowest count with only 9 Deletion

markers. This data underscores the heterogeneous distribution of Deletion events across different chromosomes, reflecting genomic regions susceptible to deletional events.

Chromosome	Number of records
chr1	1443
chr2	953
chr19	897
chr17	888
chr3	814
chr22	304
chr13	288
chr18	189
chr21	116
chrY	9

Table 11: Number of records of top 5 and bottom 5 chromosomes occurrence in deletion marker

4.1.9 Feature Correlation

The correlation table, Table 12, reveals the correlation coefficients between various genomic features and cancer types, with values ranging from -1 to +1. Most features exhibit very weak correlations with cancer type, such as 'chrom' (0.0125) and 'freq' (-0.0513), indicating minimal linear relationships. 'Name' shows the highest positive correlation (0.0436), while 'variant_type' has a notable negative correlation (-0.0425). These weak correlations suggest that cancer type is influenced by complex interactions beyond simple linear relationships.

Feature	Correlation to Cancer Type
chrom	0.012499
chromstart	0.007513
chromend	0.007513
name	0.043602
thickstart	0.007513
thickend	0.007513
blocksizes	0.001565

freq	-0.051308
hugo_symbol	0.001477
variant	0.018553
variant_type	-0.042541
reference_allele	0.004841
tumor_seq_allele1	0.004841
tumor_seq_allele2	-0.013433
gender	0.030347
cancer_type	1.000000

Table 12: Correlation value of each feature to cancer types

The correlation matrix heat map is shown below to provide a visual representation of these relationships.

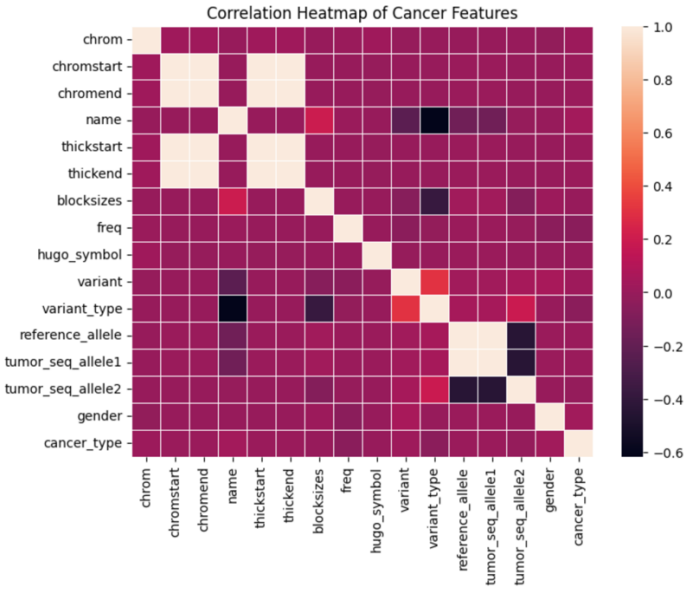


Figure 15: Correlation heatmap of cancer features.

4.1.10 Comparative Analysis

The descriptive statistics reveal notable differences among the three primary types of genetic markers: SNPs, Insertion Markers (INS), and Deletion Markers (DEL). SNPs dominate the dataset with 422,702 records, constituting approximately 94.7% of the total genetic markers. This abundance underscores their significance in genomic studies, highlighting their role in genetic variability and disease susceptibility.

In contrast, INS account for 9,807 records (2.2%) and DEL for 14,073 records (3.2%). While less prevalent than SNPs, INS and DEL markers demonstrate distinctive patterns. INS markers, for instance, exhibit a diverse range of insertion types, with insA and insT being the most frequent. Meanwhile, DEL markers display variations such as delC, delT, delA, and delG, with frame_shift_del as the predominant variant.

Furthermore, the distribution of SNP variants reveals significant insights into genetic mutations. Missense_Mutation (232,828 occurrences) and Silent (90,862 occurrences) are the most prevalent SNP variants, indicating amino acid changes and silent mutations, respectively.

4.1.11 Summary

The descriptive statistics analysis reveals a comprehensive view of genetic markers, prominently featuring Single Nucleotide Polymorphisms (SNPs), Insertions, and Deletions. SNPs constitute the majority of markers, underscoring their prevalence in the dataset. Insights into chromosomal distribution and variant classifications provide valuable context for understanding genetic diversity and mutation patterns implicated in cancer disease prediction.

4.2 Models comparison

This section presents a comparative analysis of five ML models employed to predict cancer outcomes based on genomic data: Random Forest Classifier, XGBoost, Decision Tree, Naïve Bayes, and KNN. These models were chosen for their proven efficacy in handling complex classification tasks and their varying approaches to data processing and prediction. The primary objective is to find the most accurate model for cancer prediction through the analysis of genetic markers.

Performance metrics like accuracy, precision, F1-score, and confusion matrix are employed to provide a detailed evaluation of each model. By examining these metrics, we seek to identify the model that provides the best balance of sensitivity and specificity, ultimately contributing to more reliable disease prediction and advancing the field of genomic data analysis.

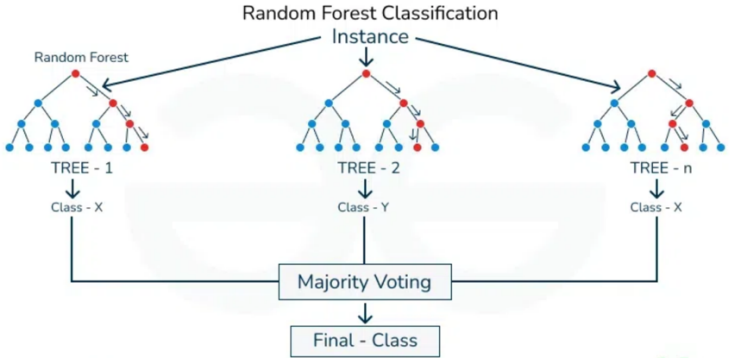
4.2.1 Model Descriptions

This section offers a comprehensive description of the ML models implemented for predicting diverse cancer outcomes through the analysis of genomic data. Each model was chosen for its

unique strengths and capabilities in handling complex datasets and various types of genetic markers.

4.2.1.1 Random Forest Classifier

Random Forest, an ensemble method, creates a multitude of decision trees and synthesizes their results to boost accuracy and lower the likelihood of overfitting. This model is particularly effective in dealing with high-dimensional data, such as genetic markers, by reducing variance and improving generalization (Breiman, 2001). Random Forest's inherent feature selection capability also helps in identifying the most significant genetic markers for disease prediction.

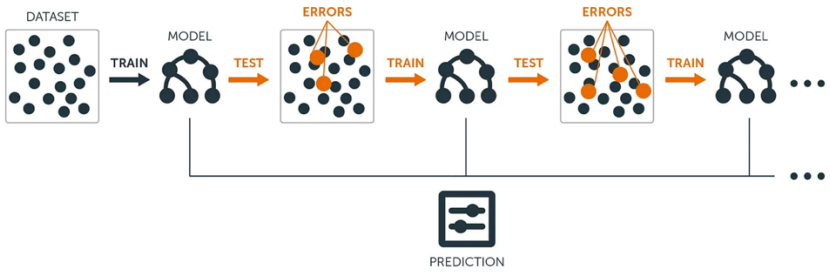


Source: www.geeksforgeeks.org

Figure 16: Random forest classifier diagram

4.2.1.2 Extreme Gradient Boosting

XGBoost is a robust gradient-boosting framework renowned for its efficiency and performance. It constructs trees in sequence, where each tree aims to correct the mistakes of its predecessor, making it adept at identifying intricate data patterns, which is valuable for genomic studies (Chen & Guestrin, 2016). Its ability to manage missing data and incorporate regularization techniques helps prevent overfitting, leading to superior predictive performance.

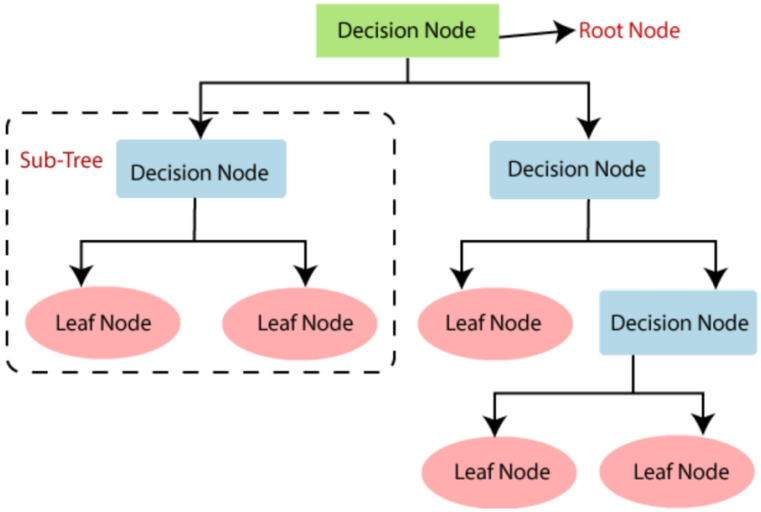


Source: www.medium.com

Figure 17: XGBoost classifier diagram

4.2.1.3 Decision Tree

Decision Trees are a straightforward yet powerful model that segments data into subsets according to the most important features (Quinlan, 1986). The algorithm creates a tree structure where nodes denote features, branches reflect decision rules, and leaves signify results. Despite its simplicity, the Decision Tree is valuable for its interpretability and capacity to handle both numerical and categorical data.

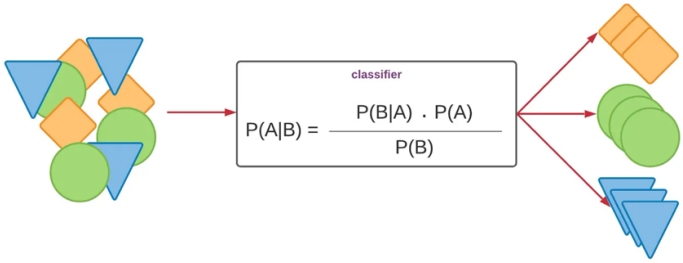


Source: www.javatpoint.com

Figure 18: Decision tree classifier diagram

4.2.1.4 Naïve Bayes

Based on Bayes' theorem, the Naïve Bayes classifier operates under the assumption that predictors are independent of each other. It is particularly effective for high-dimensional datasets and maintains strong performance with limited training data. Known for its simplicity and efficiency, Naïve Bayes is adept at solving both binary and multiclass classification problems (Lewis, 1998).

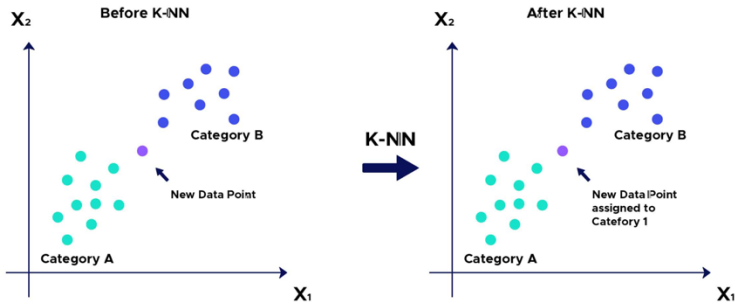


Source: www.medium.com

Figure 19: Naive Bayes classifier diagram

4.2.1.5 K-Nearest Neighbors

KNN is a non-parametric, instance-based learning approach that determines the classification of data points based on their distance from neighboring points (Altman, 1992). KNN assigns a class to a data point by majority vote from its k-nearest neighbors. This model is advantageous for its simplicity and effectiveness in scenarios where the decision boundary is irregular. However, it can become computationally expensive when applied to large datasets.



Source: www.datascientest.com

Figure 20: KNN classifier diagram

4.2.2 Performance Metrics

This section examines the performance metrics employed to evaluate and compare the efficacy of ML models in predicting cancer using genomic data. These metrics—accuracy, precision, F1-score, and the confusion matrix—offer a detailed insight into how well each model performs.

4.2.2.1 Accuracy

Accuracy measures the ratio of true results (true positives and true negatives) to the total number of cases analyzed (Powers, 2011). It offers a simple evaluation of model performance, but for imbalanced datasets, accuracy alone may not provide a complete picture of how well the model performs.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Population}}$$

Figure 21: Formula to calculate accuracy.

4.2.2.2 Precision

Precision, often referred to as positive predictive value, represents the ratio of correctly identified positive cases to the total number of predicted positive instances (Powers, 2011). High precision

reflects a minimal occurrence of false positives, making it vital for scenarios where false positives are detrimental.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Figure 22: Formula to calculate precision.

4.2.2.3 F1-Score

The F1-score combines precision and recall into a single metric by calculating their harmonic mean, offering a balanced measure of model performance. This metric is particularly valuable in situations with imbalanced datasets, where it is crucial to account for both false positives and false negatives (Powers, 2011).

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Figure 23: Formula to calculate F1-score.

4.2.2.4 Confusion Matrix

A confusion matrix is used to summarize the performance of a classification model by showing how many instances were correctly and incorrectly classified. It includes counts for true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), helping to identify error patterns.

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Figure 24: Confusion matrix table for binary classification

The confusion matrix provides insights into the performance of a classification model, highlighting how accurately the model distinguishes between different classes and where it might be failing.

4.2.3 Model Performances

The effectiveness of each model was assessed by measuring the classification accuracy on the test dataset.

4.2.3.1 Accuracy

The accuracy of the models varied significantly in our analysis. The XGBoost Classifier demonstrated the highest accuracy at 92.01%, indicating its strong predictive capability for cancer

prediction based on genetic markers. The Naive Bayes Classifier followed closely with an accuracy of 91.67%, and the Random Forest Classifier achieved 91.16%. The Decision Tree Classifier also achieved a commendable accuracy of 90.47%. However, the KNN Classifier showed a significantly lower accuracy at 73.90%. These results highlight the superior performance of ensemble methods like XGBoost and Random Forest in handling complex genomic data for disease prediction.

4.2.3.2 Precision

Precision is crucial for evaluating the accuracy of positive predictions. Among the models, the Decision Tree Classifier achieved the highest precision at 91.80%, indicating its superior capability in correctly identifying true positives. The Naive Bayes Classifier and XGBoost Classifier also performed well, with precision scores of 90.82% and 90.60%, respectively. The Random Forest Classifier had a precision of 83.88%, while the KNN Classifier showed the lowest precision at 67.48%. These results highlight that while complex models generally offer high precision, simpler models like Decision Tree can sometimes outperform them, providing critical insights into the reliability of these models in predicting genetic markers for disease.

4.2.3.3 F1-Score

The F1-score offers a comprehensive evaluation of both precision and recall for each classifier. The Decision Tree Classifier achieved the highest performance, with an F1-score of 91.79%, showcasing its effectiveness in genetic marker prediction. Following closely, Naive Bayes achieved an F1-score of 90.68%, emphasizing its robust performance despite its simpler nature. XGBoost also performed well with an F1-score of 86.09%, indicating its ability to handle complex relationships in the data. The KNN Classifier, on the other hand, recorded a lower F1-score of 69.19%, suggesting limitations in capturing the intricacies of the dataset. Overall, these results underscore the varied strengths and limitations of each model in genomic data analysis for disease prediction.

4.2.4 Comparative Analysis

In this section, the performance of five ML models—Random Forest, XGBoost, Decision Tree, KNN, and Naïve Bayes—is evaluated and compared based on several key metrics such as accuracy, precision, F1-score, and confusion matrix.

The accuracy, precision, and F1 score for each model are as follows:

Model	Accuracy (%)	Precision (%)	F1 Score (%)
Random Forest	91.16	83.88	85.68
XGBoost	92.01	90.60	86.09
Decision Tree	90.47	91.79	91.78
KNN	73.90	67.48	69.19
Naïve Bayes Classifier	91.67	90.82	90.68

Table 13: Model performance

The above tables show the prediction, precision, and F1 score of five different models- Random Forest, XGBoost, Decision Tree, KNN, and Naïve Bayes; used for cancer disease prediction. These models were evaluated for their accuracy and precision, key metrics for determining their effectiveness in predicting disease presence.

The XGBoost model outperformed the others, achieving the highest accuracy of 92.01% and a strong precision of 90.60%. This indicates XGBoost's superior ability to correctly classify cancer cases while maintaining a low rate of false positives, supported by its F1 score of 86.09%. The Naïve Bayes Classifier also demonstrated excellent performance, achieving an accuracy of 91.67%, a precision of 90.82%, and an F1 score of 90.68%, highlighting its robustness in handling the complexity of genomic data. The Random Forest Classifier showed reliable performance achieving an accuracy of 91.16%, a precision of 83.88% and an F1 score of 85.68%, though its precision was slightly lower than that of XGBoost and Naïve Bayes.

The Decision Tree Classifier achieved an accuracy of 90.47%, the highest precision at 91.79%, and the highest F1 score of 91.78%, indicating its strong ability to correctly identify true positives, though it lagged slightly behind XGBoost and Naïve Bayes in overall accuracy. In contrast, the KNN had the lowest performance, with an accuracy of 73.90%, a precision of 67.48%, and an F1 score of 69.19%, suggesting it is less effective for this specific genomic data analysis task.

Overall, XGBoost and Naïve Bayes emerged as the most effective models for cancer disease prediction in our study, balancing high accuracy, precision, and F1 score, making them particularly suitable for this application.

4.2.5 Visualization of Results

This section presents visual representations that illustrate the performance and outcomes of the five ML models—KNN, XGBoost, Random Forest, Naïve Bayes, and Decision Tree—used for predicting diseases based on genetic markers. These visualizations offer a clear and intuitive comparison of each model's effectiveness.

4.2.5.1 Performance Metrics Visualization

- **Accuracy Comparison:** The following bar graph of accuracy reveals the overall prediction correctness of each model, with XGBoost showing the highest accuracy at 92.01%, closely followed by Naïve Bayes at 91.67%.

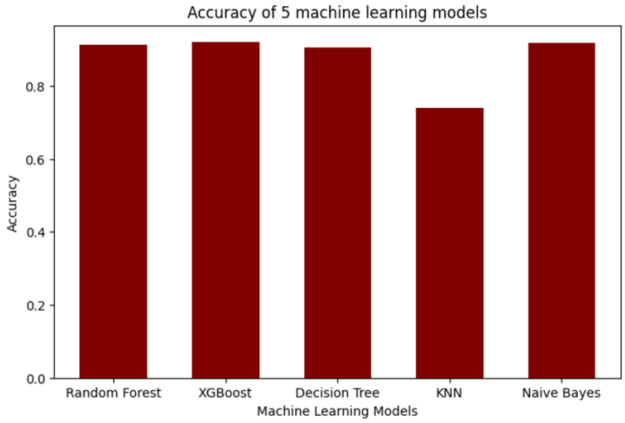


Figure 25: Bar graph comparing the accuracy of 5 ML models.

- **Precision Comparison:** The following bar graph of precision illustrates the effectiveness of each model in correctly identifying true positives out of all positive predictions. The Decision Tree Classifier demonstrates the highest precision at 91.79%, indicating its ability to minimize false positives.

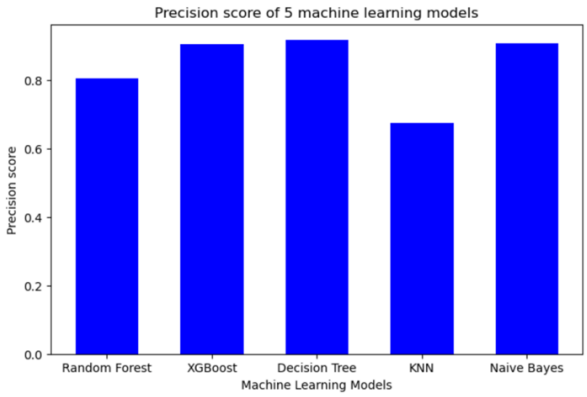


Figure 26: Bar graph comparing the precision score of 5 ML models.

- **F1-Score Comparison:** The F1 score bar graph integrates both precision and recall into a unified metric, offering a comprehensive view of model effectiveness. Models with higher F1 scores demonstrate superior performance by reducing both false positives and false negatives. The Decision Tree Classifier also leads in F1 score with 91.78%, reflecting its effectiveness in disease prediction tasks.

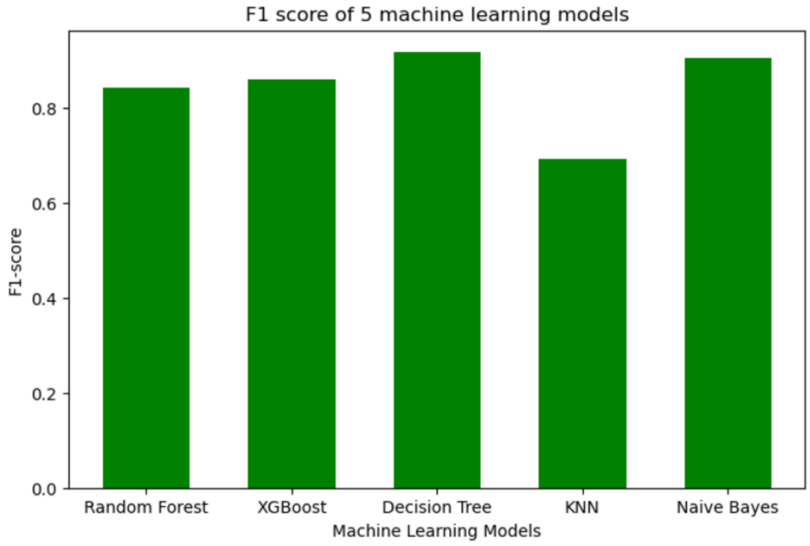


Figure 27: Bar graph of F1-score comparisons of 5 ML models.

- **Confusion Matrix:** Each model's confusion matrix visually illustrates the distribution of predicted versus actual outcomes across different classes. It allows for a clear assessment of both correct and incorrect prediction, aiding in the evaluation of the model's accuracy and potential areas of misclassification.

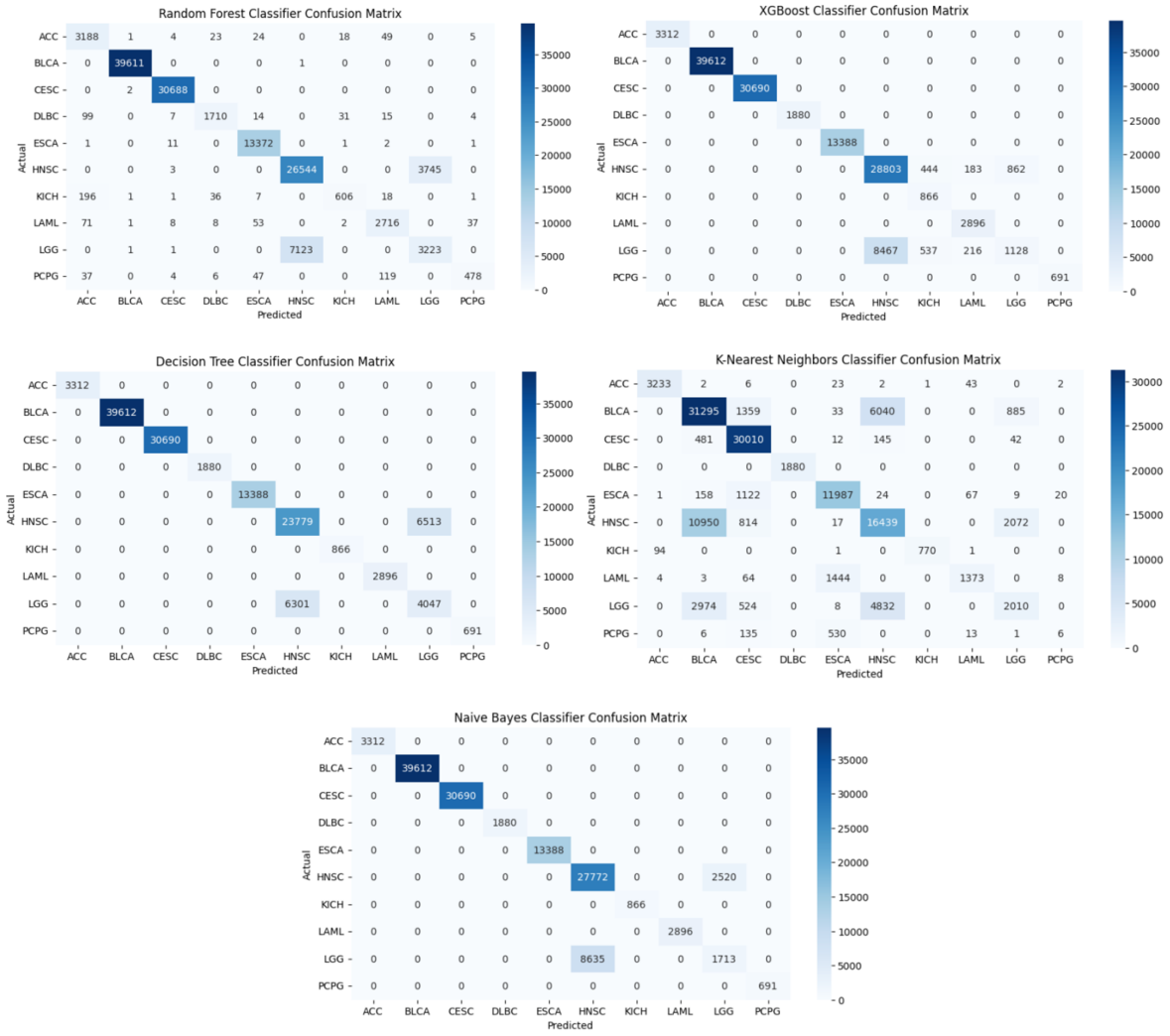
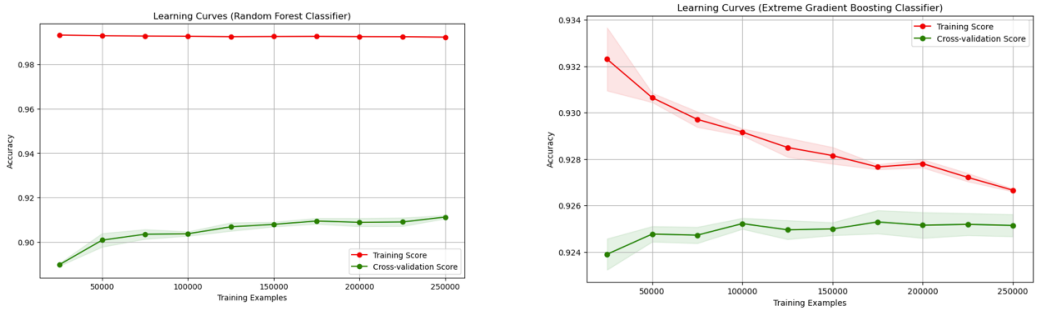


Figure 28: Confusion matrices of 5 ML models.

4.2.5.2 Learning Curve

Learning curves depict how model performance improves with increasing amounts of training data. They provide insights into whether the models would benefit from additional data or if they are at risk of overfitting with the current dataset size.



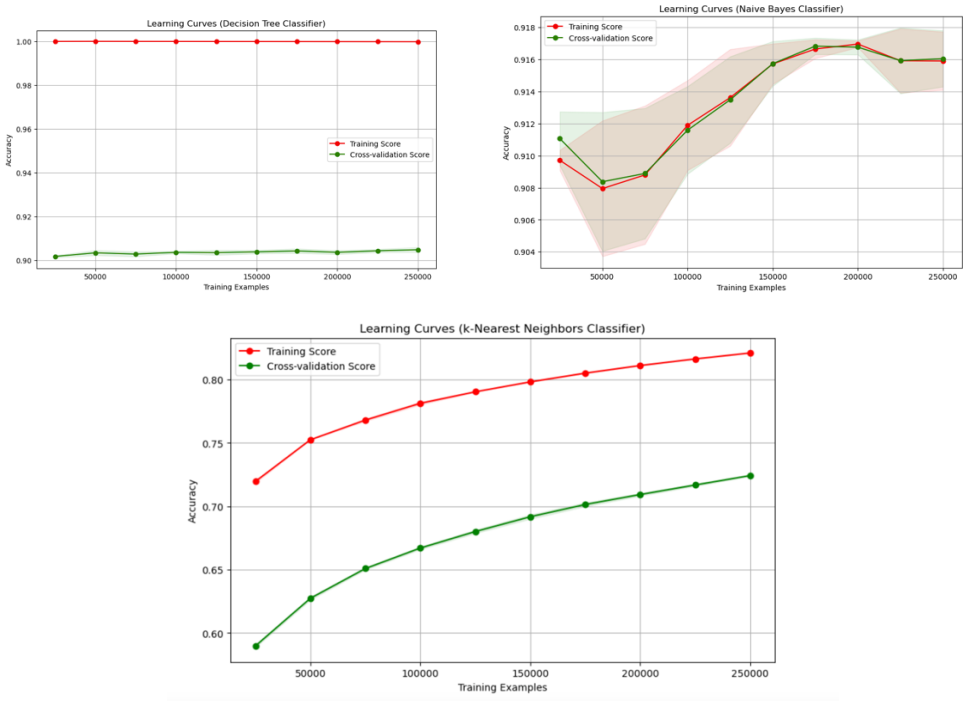


Figure 29: Learning curves of 5 ML models.

The learning curves indicate that all five models perform well as more training data is provided. Notably, XGBoost's learning curve shows a slight decrease in training scores over time, but this decrease is minimal (0.5%) and does not indicate overfitting. The convergence of training and validation scores across all models suggests they are effectively learning from the available data without overfitting, demonstrating their robustness in capturing patterns and generalizing to unseen data.

4.2.5.3 Summary of Visualizations

The visualizations collectively underscore the robustness and effectiveness of XGBoost and Random Forest in predicting disease from genomic data. XGBoost's superior performance across various metrics and its ability to manage complex data interactions are depicted in the accuracy, precision, F1-score, and Learning curves. Random Forest also demonstrates strong performance, particularly in handling high-dimensional data and identifying significant genetic markers.

These visualizations not only validate the quantitative results discussed earlier but also provide a visual affirmation of the models' predictive capabilities and the importance of specific genetic markers in disease prediction. The insights gained from these visualizations are crucial for

advancing the field of genomic data analysis and improving predictive modelling in genetic epidemiology.

4.3 Result Analysis

Our genomic data analysis aimed to decode the intricate relationships between genetic markers and cancer disease prediction, leveraging advanced ML models. From a comprehensive evaluation of five ML models using data from TCGA, several key insights emerged.

4.3.1 Descriptive Analysis of Genomic Data

The descriptive analysis of genomic data uncovered several significant patterns in genetic markers associated with cancer:

- **Prevalence of Genetic Markers:** SNPs were predominant compared to insertion and deletion markers. The most frequent SNP mutation was C>T, indicating a thymine substitution for cytosine, prevalent across various chromosomes.
- **Chromosome Distribution:** Chromosome 1 exhibited the highest number of genetic markers, followed by chromosome 2 and chromosome 19. This distribution highlights specific genomic regions potentially crucial for cancer susceptibility.
- **Variant Classifications:** Among SNP markers, the Missense Mutations variant was most prevalent, followed by Silent Mutations. In insertion markers, insA and insT were predominant, while deletion markers showed a high occurrence of delC, followed by delT, delA, and delG. Frame_Shift variants were the most recorded type in both insertion and deletion markers.

4.3.2 Model Performance

The study evaluated five ML models for cancer disease prediction based on genomic data:

- **XGBoost:** Achieved the highest accuracy of 92.01% and precision of 90.60%, indicating robust performance in classifying cancer types.
- **Naïve Bayes:** Demonstrated strong performance with an accuracy of 91.67% and precision of 90.82%, leveraging its probabilistic approach for effective classification.
- **Random Forest:** Showed reliable accuracy at 91.16% but had a lower precision of 83.88%, suggesting potential for further optimization.

- **Decision Tree:** Achieved an accuracy of 90.47% and a high precision of 91.79%, indicating its capability to correctly identify positive cases.
- **K-Nearest Neighbors:** Exhibited lower effectiveness with an accuracy of 73.90% and precision of 67.48%, indicating challenges in handling the complexity of genomic data.

Overall, XGBoost and Naïve Bayes emerged as the most effective models for cancer disease prediction in our study, balancing high accuracy and precision, making them particularly suitable for this application.

4.4 Limitations

While our study on genomic data analysis for cancer disease prediction provides valuable insights and significant findings, it is important to consider several limitations to fully understand the scope and applicability of our results. Trained models like XGBoost, Random Forest, Decision Tree, Naive Bayes, and KNN demonstrated robust performance, but acknowledging these limitations is crucial for contextualizing our findings.

Firstly, the dataset used in this study was sourced exclusively from TCGA. Although TCGA is a comprehensive and well-curated resource, it may not fully represent the genetic diversity and various environmental factors influencing cancer across different populations. This limitation could affect the generalizability of our models to broader demographic groups and different cancer types not included in TCGA. This bias in dataset composition restricts the applicability of our predictive models beyond cancer-specific scenarios.

Secondly, our study focused solely on evaluating the performance of the models within the confines of the TCGA dataset. These models have not been tested on data from other sources, which could provide different insights or reveal additional patterns. The reliance on a single data source limits our ability to validate the models' robustness and their applicability in other clinical scenarios.

The use of advanced models such as XGBoost, Naive Bayes, Random Forests and Naive Bayes for detecting various cancers, while effective, presents challenges. These models often require significant computational resources for training and deployment, which were not accounted for in

our study. This oversight highlights the potential practical constraints when implementing these models in real-world clinical environments.

Furthermore, while advanced ML models like XGBoost, Naïve Bayes and Random Forests offer superior predictive performance, their complex nature often sacrifices interpretability. Understanding the biological significance of specific genetic markers identified by these models remains challenging, limiting our ability to translate findings into actionable clinical insights or biomarkers.

Additionally, the process of collecting, preparing, and processing the dataset itself poses inherent limitations. Issues such as data quality, missing values, and inconsistencies in sample collection can introduce biases and affect the robustness and reliability of the analysis. Moreover, the dataset does not cover the full spectrum of all cancer cases, potentially limiting the comprehensive applicability of our findings.

Another significant limitation is that our study primarily considered prediction accuracy and precision for evaluating model performance. While these metrics are essential, other important factors such as the interpretability and explainability of the models were not addressed. Understanding the decision-making process of complex models like XGBoost is crucial for their acceptance and integration into clinical practice.

Despite these constraints, the results from the study offer important perspectives on the effectiveness of using cutting-edge ML models for cancer prediction with genetic markers. Addressing these limitations through additional research with larger and more diverse datasets, exploring alternative model architectures, and employing comprehensive evaluation strategies can substantially improve the effectiveness and relevance of these predictive models. Future studies should focus on validating these models across various datasets, considering the computational resources required for training and deployment, and prioritizing model interpretability to ensure the practical applicability and acceptance of these predictive tools in clinical settings.

4.5 Conclusion

This chapter explored genomic data to uncover genetic markers vital for disease prediction across various cancers. Through detailed descriptive statistics, we identified a diverse array of genetic

variants, including SNPs, insertions, and deletions, underscoring the complexity of genetic factors influencing disease susceptibility.

Evaluation of ML models—Random Forest, XGBoost, Decision Tree, Naïve Bayes, and KNN—revealed XGBoost as the most effective, achieving an accuracy of 92.01% and an F1-score of 90.60%. These findings contribute significantly to understanding genetic markers' roles in disease prediction, aiding in the development of targeted therapies.

Several limitations were discussed in this chapter, including the use of TCGA data, which may not fully represent genetic diversity, and the computational demands and interpretability challenges of advanced models like XGBoost and Random Forest.

This research contributes to the field by enhancing our understanding of genetic markers' roles in disease prediction and underscores the potential for personalized healthcare interventions based on individual genetic profiles.

The next chapter concludes this dissertation by summarizing the main findings, exploring their wider implications, and suggesting directions for future research.

Chapter 5: Conclusion

The field of genomic data analysis has the potential to revolutionize cancer disease prediction by uncovering the intricate patterns of genetic markers associated with various types of cancer. This dissertation has focused on exploring and evaluating the effectiveness of different ML models in predicting diverse cancer disease outcomes based on genomic data and uncovering patterns within genetic markers through advanced visualization techniques. The study has provided significant insights into the performance of models such as Random Forest, XGBoost, Decision Tree, KNN, and Naïve Bayes, highlighting their strengths and limitations. By addressing the complexities of genetic data and leveraging advanced analytical techniques, this research aims to decipher how genetic markers contribute to disease susceptibility and progression.

The following sections summarize key findings, discuss challenges, and propose directions for future research to enhance the practical utility of genomic data analysis in cancer disease prediction.

5.1 Key Findings

The descriptive analysis of the genomic data revealed several important patterns and trends in genetic markers. SNPs were more prevalent compared to insertion and deletion markers. Chromosome 1 had the highest number of genetic markers, followed by chromosome 2 in second and chromosome 19 in third. Among SNP markers, the C>T substitution was the most frequent, indicating a thymine (T) substitution for cytosine (C) across various chromosomes. The highest number of SNP variants recorded were Missense Mutations, followed by Silent. In insertion markers, insA (adenine insertion) and insT (thymine insertion) were predominant, with Frame_Shift_Ins being the most frequent variant, followed by 3'UTR and In_Frame_Ins. Deletion markers showed a high occurrence of delC, with delT, delA, and delG also frequently observed. The most recorded deletion variant was Frame_Shift_Del, followed by 3'UTR.

The performance evaluation of ML models for disease prediction demonstrated varying degrees of accuracy and precision. XGBoost achieved the highest accuracy at 92.01% and a strong precision of 90.60%, making it the most effective model in this study. The Naïve Bayes classifier also performed well with an accuracy of 91.67% and a precision of 90.82%. The Random Forest classifier showed a reliable accuracy of 91.16% but had a lower precision of 83.88%. The Decision Tree classifier had an accuracy of 90.47% and a precision of 91.79%, indicating a high capability

for true positive identification. In contrast, the KNN classifier was less effective, with an accuracy of 73.90% and a precision of 67.48%, suggesting it was less effective for this type of genomic data analysis.

5.2 Implications of the Study

This research emphasizes the substantial promise of analyzing genomic data for predicting cancer, especially with the application of sophisticated ML models. The findings demonstrate that models like XGBoost and Naïve Bayes can achieve high accuracy and precision, suggesting their applicability in clinical settings for early disease detection and personalized treatment planning. Additionally, the study underscores the importance of integrating diverse genomic markers, such as SNPs, insertions, and deletions, to improve predictive accuracy. The identification of pivotal genetic variants connected to disease risk yields important insights that can drive further research and the design of targeted therapies. However, the limitations identified emphasize the need for ongoing refinement of models and the incorporation of diverse datasets to ensure broad applicability and robustness in real-world scenarios.

5.3 Challenges and Limitations

Despite the promising results, this study encountered several challenges and limitations in its exploration of genomic data for cancer disease prediction. Firstly, the reliance on data primarily sourced from TCGA restricted the diversity and general applicability of findings to broader populations and other diseases beyond cancer. The dataset predominantly represents specific demographics, potentially overlooking genetic variations in other ethnic groups or geographical regions.

The evaluation focused predominantly on metrics like accuracy and precision, overlooking crucial factors such as model interpretability and computational efficiency. The inherent complexity of genomic data posed challenges in preprocessing and feature selection, which could impact the models' performance and reliability. Furthermore, the computational resources required for training and deploying these models were not comprehensively addressed, which could impede scalability in real-world clinical applications.

Another critical consideration is the interpretability of the models, particularly advanced techniques like XGBoost, Naïve Bayes and Random Forest. While these models excel in predictive accuracy, understanding the underlying biological mechanisms driving their predictions remains

challenging. This lack of interpretability poses a barrier to their adoption in clinical decision-making processes, where transparency and explanatory power are paramount.

Addressing these limitations in future research by incorporating more diverse datasets and enhancing computational resources will be pivotal in advancing the robustness and practical application of genomic-based disease prediction models. Future research endeavors should adopt comprehensive evaluation frameworks that encompass these dimensions, ensuring holistic assessments of model efficacy and practical utility in clinical practice.

5.4 Future Research Directions

Looking ahead, advancing the field of genomic data analysis for disease prediction requires concerted efforts in several key areas. First and foremost, expanding the scope of genomic datasets to encompass a wider spectrum of diseases and diverse population groups is imperative. This approach will improve the applicability of predictive models and aid in uncovering new genetic markers and disease mechanisms.

Secondly, exploring cutting-edge ML methods, including DL and ensemble approaches, could significantly enhance model performance and interpretability. These approaches have the potential to capture complex interactions among genetic markers and other biological factors more effectively, leading to more reliable predictions.

Additionally, integrating genomic data analysis into personalized medicine frameworks holds immense potential for revolutionizing patient care. By leveraging predictive insights from genetic markers, healthcare providers can tailor treatment strategies, optimize therapeutic outcomes, and ultimately improve patient outcomes and quality of life.

By focusing on these recommendations, future research can advance genomic data analysis towards more robust, interpretable, and ethically sound approaches in disease prediction, ultimately benefiting public health and personalized medicine initiatives.

5.5 Conclusion

In conclusion, this dissertation contributes to advancing our understanding of genomic data analysis for disease prediction, particularly in the context of cancer. While significant strides have been made in predictive accuracy and model performance, continued innovation, collaboration across disciplines, and a commitment to ethical practices are essential for realizing the full

transformative potential of genomic insights in healthcare. By overcoming current limitations and leveraging the strengths of advanced analytics, the field can advance towards a more precise, personalized, and effective approach to disease prevention, diagnosis, and treatment based on genomic insights.

References

1. Collins, F.S., Green, E.D., Guttmacher, A.E. and Guyer, M.S., 2003. A vision for the future of genomics research. Nature, 422(6934), pp.835-847.
2. Manolio, T.A., Brooks, L.D. and Collins, F.S., 2009. A HapMap harvest of insights into the genetics of common disease. Journal of Clinical Investigation, 119(8), pp.2108-2116.
3. Ritchie, M.D., Holzinger, E.R., Li, R., Pendergrass, S.A. and Kim, D., 2015. Methods of integrating data to uncover genotype-phenotype interactions. Nature Reviews Genetics, 16(2), pp.85-97.
4. Ham, T.J., Bruns-Smith, D., Sweeney, B., Lee, Y., Seo, S.H., Song, U.G., Oh, Y.H., Asanovic, K., Lee, J.W., and Wills, L.W., 2020. Genesis: A Hardware Acceleration Framework for Genomic Data Analysis. 2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA), 30 May-3 June 2020, pp.254-267. DOI: 10.1109/ISCA45697.2020.00031.
5. Ki, Y. and Yoon, J.W., 2017. An Efficient Method for Securely Storing and Handling of Genomic Data. 2017 International Conference on Software Security and Assurance (ICSSA), 24-25 July 2017, pp.121-125. DOI: 10.1109/ICSSA.2017.13.
6. Senadheera, S.P.B.M. and Weerasinghe, A.R., 2020. Genomic Data Analyzing Workflow for Single Nucleotide Polymorphisms in Human Nervous System Cancers. 2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer), 4-7 Nov. 2020, pp.107-112. DOI: 10.1109/ICTer51097.2020.9325461.
7. Sinecen, M., 2019. Comparison of Genomic Best Linear Unbiased Prediction and Bayesian Regularization Neural Networks for Genomic Selection. IEEE Access, 7, pp.79199-79210. DOI: 10.1109/ACCESS.2019.2922006.
8. Zheng, Y., Lu, R., Shao, J., Zhang, Y., and Zhu, H., 2019. Efficient and Privacy-Preserving Edit Distance Query Over Encrypted Genomic Data. 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP), 23-25 Oct. 2019, pp.1-6. DOI: 10.1109/WCSP.2019.8927885.
9. Al Kawam, A., Sen, A., Datta, A. and Dickey, N., 2018. Understanding the bioinformatics challenges of integrating genomics into healthcare. IEEE Journal of Biomedical and Health Informatics, 22(5), pp. 1672-1683. doi: 10.1109/JBHI.2017.2778263.

10. DeGroat, W., Abdelhalim, H., Patel, K., Mendhe, D., Zeeshan, S. and Ahmed, Z., 2024. Discovering biomarkers associated and predicting cardiovascular disease with high accuracy using a novel nexus of machine learning techniques for precision medicine. Scientific Reports, 14(1), p. 1. doi: 10.1038/s41598-023-50600-8.
11. Gancheva, V. and Borovska, P., 2019. SOA based system for big genomic data analytics and knowledge discovery. In: 2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 18-21 September 2019, Bucharest, Romania. IEEE, pp. 536-541. doi: 10.1109/IDAACS.2019.8924370.
12. Jones, T., 2024. Unveiling the power of Genomic Data Analysis and how it's Transforming Genomics Research! Unveiling the Power of Genomic Data Analysis and How it's Transforming Genomics Research!. Available at: https://www.kinetica.co.uk/resources/blog/unveiling-the-power-of-genomic-data-analysis-and-how-it-s-transforming-genomics-research-/
13. Liu, W.Y., Hsiao, H.-I. and Dai, S.Y., 2015. Genomic analysis with MapReduce. In: 2015 IEEE International Conference on Big Data (Big Data), 29 October - 1 November 2015, Santa Clara, CA. IEEE, pp. 1330-1335. doi: 10.1109/BigData.2015.7363891.
14. Shao, W., Han, Z., Cheng, J., Cheng, L., Wang, T., Sun, L., Lu, Z., Zhang, J., Zhang, D. and Huang, K., 2020. Integrative analysis of pathological images and multi-dimensional genomic data for early-stage cancer prognosis. IEEE Transactions on Medical Imaging, 39(1), pp. 99-110. doi: 10.1109/TMI.2019.2920608.
15. Silva, P.P., Gaudillo, J.D., Vilela, J.A., et al., 2022. A machine learning-based SNP-set analysis approach for identifying disease-associated susceptibility loci. Scientific Reports, 12, 15817. DOI: 10.1038/s41598-022-19708-1.
16. Widen, E., Raben, T.G., Lello, L. and Hsu, S.D.H., 2021. Machine learning prediction of biomarkers from SNPs and of disease risk from biomarkers in the UK Biobank. Genes (Basel), 12(7), p. 991. doi: 10.3390/genes12070991.
17. Xu, J., Yang, P., Xue, S., and Sharma, B., 2018. AI approaches in cancer genomics. Cancer Informatics, 24(1), pp. 95-108.
18. Anders, S. and Huber, W., 2010. Differential expression analysis for sequence count data. Nature Precedings. doi: 10.1038/npre.2010.4282.2

19. Auti, R., Bhatt, A. and Tidake, S., 2023. Comparative analysis of machine learning algorithms for genomic data. In: 2023 1st DMIHER International Conference on Artificial Intelligence in Education and Industry 4.0 (IDICAIEI), 27-28 November 2023. IEEE, pp. 1-6. doi: 10.1109/IDICAIEI58380.2023.10406455.
20. Evangelou, E. and Ioannidis, J.P.A., 2013. Meta-analysis methods for genome-wide association studies and beyond. Nature Reviews Genetics, 14, pp. 379-389. doi: 10.1038/nrg3472.
21. Krzywinski, M., Schein, J., Birol, I., Connors, J., Gascoyne, R., Horsman, D., Jones, S.J. and Marra, M.A., 2009. Circos: an information aesthetic for comparative genomics. Genome Research, 19(9), pp. 1639-1645. doi: 10.1101/gr.092759.109.
22. Langmead, B. and Salzberg, S.L., 2012. Fast gapped-read alignment with Bowtie 2. Nature Methods, 9(4), pp. 357-359. doi: 10.1038/nmeth.1923.
23. Li, H. and Durbin, R., 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics, 25(14), pp. 1754-1760. doi: 10.1093/bioinformatics/btp324.
24. McKenna, A., Hanna, M., Banks, E., Sivachenko, A., Cibulskis, K., Kernytsky, A., Garimella, K., Altshuler, D., Gabriel, S., Daly, M. and DePristo, M.A., 2010. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome Research, 20(9), pp. 1297-1303. doi: 10.1101/gr.107524.110.
25. Min, S., Lee, B. and Yoon, S., 2017. Deep learning in bioinformatics. Briefings in Bioinformatics, 18(5), pp. 851-869. doi: 10.1093/bib/bbw068.
26. Robinson, J.T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E.S., Getz, G. and Mesirov, J.P., 2011. Integrative genomics viewer. Nature Biotechnology, 29(1), pp. 24-26. doi: 10.1038/nbt.1754.
27. Visscher, P.M., Wray, N.R., Zhang, Q., Sklar, P., McCarthy, M.I., Brown, M.A. and Yang, J., 2017. 10 years of GWAS discovery: biology, function, and translation. American Journal of Human Genetics, 101(1), pp. 5-22. doi: 10.1016/j.ajhg.2017.06.005.
28. Wang, K., Li, M. and Hakonarson, H., 2010. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. Nucleic Acids Research, 38(16), p. e164. doi: 10.1093/nar/gkq603.

29. Wu, Q., Boueiz, A., Bozkurt, A., Masoomi, A., Wang, A., DeMeo, D.L., Weiss, S.T. and Qiu, W., 2018. Deep learning methods for predicting disease status using genomic data. Journal of Biometrics & Biostatistics, 9(5), p. 417.
30. Love, M.I., Huber, W. and Anders, S., 2014. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology, 15(12), p. 550. doi: 10.1186/s13059-014-0550-8.
31. Schadt, E., Linderman, M., Sorenson, J. et al., 2010. Computational solutions to large-scale data management and analysis. Nature Reviews Genetics, 11, pp. 647-657. doi: 10.1038/nrg2857.
32. Stuart, T. and Satija, R., 2019. Integrative single-cell analysis. Nature Reviews Genetics, 20(5), pp. 257-272. doi: 10.1038/s41576-019-0093-7.
33. Phogat, M. and Kumar, D., 2022. Feature selection techniques for genomic data. In: 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), 26-27 May 2022. IEEE, pp. 785-790. doi: 10.1109/COM-IT-CON54601.2022.9850466
34. Döngel, T. and Timar, Y., 2017. B3SafirBiyo: Genomic variant analysis with big data technologies. In: 2017 IEEE International Conference on Big Data (Big Data), 11-14 December 2017, Boston, MA. IEEE, pp. 1-9. doi: 10.1109/BigData.2017.8258137
35. Aledhari, M., Pierro, M.D., Hefeida, M. and Saeed, F., 2021. A deep learning-based data minimization algorithm for fast and secure transfer of big genomic datasets. IEEE Transactions on Big Data, 7(2), pp. 271-284. doi: 10.1109/TBDATA.2018.2805687
36. Manolio, T.A., 2010. Genomewide association studies and assessment of the risk of disease. New England Journal of Medicine, 363(2), pp. 166-176. doi: 10.1056/NEJMra0905980
37. The International Cancer Genome Consortium, 2010. International network of cancer genome projects. Nature, 464, pp. 993-998. doi: 10.1038/nature08987
38. Kourou, K., Exarchos, T.P., Exarchos, K.P., Karamouzis, M.V. and Fotiadis, D.I., 2015. Machine learning applications in cancer prognosis and prediction. Computational Structure Biotechnology Journal, 13, pp. 8-17. doi: 10.1016/j.csbj.2014.11.005.
39. Libbrecht, M.W. and Noble, W.S., 2015. Machine learning applications in genetics and genomics. Nature Reviews Genetics, 16(6), pp. 321-332. doi: 10.1038/nrg3920.

40. Sudlow, C., Gallacher, J., Allen, N., Beral, V., Burton, P., Danesh, J., Downey, P., Elliott, P., Green, J., Landray, M., Liu, B., Matthews, P., Ong, G., Pell, J., Silman, A., Young, A., Sprosen, T., Peakman, T. and Collins, R., 2015. UK Biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. PLoS Medicine, 12(3), p. e1001779. doi: 10.1371/journal.pmed.1001779
41. The Cancer Genome Atlas Research Network, Weinstein, J.N., Collisson, E.A. et al., 2013. The Cancer Genome Atlas Pan-Cancer analysis project. Nature Genetics, 45, pp. 1113-1120. doi: 10.1038/ng.2764.
42. Chandrashekar, G. and Sahin, F., 2014. A survey on feature selection methods. Computers & Electrical Engineering, 40(1), pp. 16-28. doi: 10.1016.
43. Guyon, I. and Elisseeff, A., 2003. An introduction to variable and feature selection. Journal of Machine Learning Research, 3, pp. 1157-1182.
44. James, G., Witten, D., Hastie, T. and Tibshirani, R., 2013. An introduction to statistical learning: with applications in R. Springer.
45. Chen, T. and Guestrin, C., 2016. XGBoost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16), 13-17 August 2016, San Francisco, CA. Association for Computing Machinery, New York, NY, pp. 785-794. doi: 10.1145/2939672.2939785
46. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, É., 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, pp. 2825-2830.
47. Breiman, L., 2001. Random forests. Machine Learning, 45(1), pp. 5-32. doi:10.1023/A:1010933404324
48. Altman, N.S., 1992. An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician, 46(3), pp. 175-185. doi: 10.1080/00031305.1992.10475879
49. Powers, D.M.W., 2011. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. Journal of Machine Learning Technologies, 2(1), pp. 37-63.

Appendix

```
# Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from google.colab import drive
import seaborn as sns
from sklearn.utils import shuffle
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split, StratifiedKFold
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report, precision_recall_curve, confusion_matrix, f1_score
from sklearn.preprocessing import label_binarize
from xgboost import XGBClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import precision_score, roc_curve, auc
import warnings
warnings.filterwarnings('ignore')
from sklearn.model_selection import learning_curve
```

```
cancerTypes = ['cesc','acc','laml','esca','dlbc','blca','hnsc','pcpg','kich','lgg']
fileName = lambda x: x + '_cancer.csv'
# Create an empty dataframe to store the combined data
cancerData_df = pd.DataFrame()

# Loop through each CSV file and append its contents to the combined dataframe
for cancerType in cancerTypes:
    #print("data/" + fileName(cancerType))
    df = pd.read_csv("data/" + fileName(cancerType))
    df = df[df['project_id' ].str.contains(',')==False]
    df['project_id'] = df['project_id'].apply(lambda x: x.replace('TCGA-', ''))
    cancerData_df = pd.concat([cancerData_df, df])

cancerData_df = shuffle(cancerData_df, random_state=42)
cancerData_df.to_csv("data/project_cancer_data.csv", index=False)
```

```
cancerData_df.head()
```

```
cancerData_df = pd.read_csv('data/project_cancer_data.csv')
```

```
cancerData_df.head()
```

```
cancerData_df.tail()
```

```
cancerData_df.shape
```

```
cancerData_df.info()
```

```
cancerData_df.describe()
```

```
cancerData_df.describe(include='O')
```

```
cancerData_df.isnull().sum()
```

```
cancerData_df.columns
```

```
cancerData_df['Variant_Classification'].unique()
```

```
cancerData_df['Variant_Type'].unique()
```

```
cancerData_df.dtypes
```

```
#To know, How many values available in object('categorical') type of features
#And Return Categorical values with Count.
def explore_object_type(df ,feature_name):
       if df[feature_name].dtype ==  'object':
        print(df[feature_name].value_counts())
```

```
# Now, Test and Call a function for gender only
explore_object_type(cancerData_df, 'gender')
```

```
explore_object_type(cancerData_df, 'Variant_Type')
```

```
explore_object_type(cancerData_df, 'Variant_Classification')
```

```
explore_object_type(cancerData_df, '#"chrom"')
```

```
cancerData_snp_df = cancerData_df[cancerData_df["Variant_Type"] == "SNP"]
cancerData_snp_df.head()
```

```
explore_object_type(cancerData_snp_df, 'name')
```

```
explore_object_type(cancerData_snp_df, 'Variant_Classification')
```

```
explore_object_type(cancerData_snp_df, '#"chrom"')
```

```
cancerData_ins_df = cancerData_df[cancerData_df["Variant_Type"] == "INS"]
cancerData_ins_df.head()
```

```
explore_object_type(cancerData_ins_df, 'name')
```

```
explore_object_type(cancerData_ins_df, 'Variant_Classification')
```

```
explore_object_type(cancerData_ins_df, '#"chrom"')
```

```
cancerData_del_df = cancerData_df[cancerData_df["Variant_Type"] == "DEL"]
cancerData_del_df.head()
```

```
explore_object_type(cancerData_del_df, 'name')
```

```
explore_object_type(cancerData_del_df, 'Variant_Classification')
```

```
explore_object_type(cancerData_del_df, '#"chrom"')
```

```
cancerData_blca_df = cancerData_df[cancerData_df['project_id' ]=='BLCA']
cancerData_hnsc_df = cancerData_df[cancerData_df['project_id' ]=='HNSC']
cancerData_cesc_df = cancerData_df[cancerData_df['project_id' ]=='CESC']
```

```
explore_object_type(cancerData_blca_df, 'name')
```

```
explore_object_type(cancerData_blca_df, 'Variant_Classification')
```

```
explore_object_type(cancerData_hnsc_df, 'name')
```

```
explore_object_type(cancerData_hnsc_df, 'Variant_Classification')
```

```
explore_object_type(cancerData_cesc_df, 'name')
```

```
explore_object_type(cancerData_cesc_df, 'Variant_Classification')
```

```
# Getting number of unique values in each column
result_df = pd.DataFrame(cancerData_df.nunique(), columns=['No. of Unique Values']).sort_values(by='No. of Unique Values')
print(result_df.to_string())
```

```
cancerData_df.select_dtypes(include=['int64']).columns
```

```
cancerData_df.info()
```

```
# Columns to lowercase
cancerData_df.columns = cancerData_df.columns.str.lower()

# Rename columns
cancerData_df.rename(columns={'#"chrom"':'chrom', 'project_id':'cancer_type', 'variant_classification':'variant',
                              'matched_norm_sample_barcode':'barcode'}, inplace=True)
```

```
# Specify the columns to split
columns_to_split = ['days_to_death', 'cigarettes_per_day', 'weight', 'alcohol_history', 'alcohol_intensity',
                    'bmi', 'years_smoked', 'height', 'gender', 'ethnicity', 'tumor_sample_barcode', 'barcode', 'case_id']

# Convert columns to strings and then split values
cancerData_df.loc[:, columns_to_split] = cancerData_df[columns_to_split].astype(str).apply(lambda x: x.str.split(','))

# Explode the specified columns and reset the index
cancerData_df = cancerData_df.explode(columns_to_split).reset_index(drop=True)
```

```
cancerData_df.shape
```

```
# Raw data has null values with dashes '--'
cancerData_df.head(2)
```

```
# Replace '--' with nan
cancerData_df.replace('--', np.nan, inplace=True)
cancerData_df.head(2)
```

```
# Check missing values
cancerData_df.isnull().sum()
```

```
# Drop based on null values
cancerData_df.drop(columns=['dbsnp_rs', 'dbsnp_val_status', 'days_to_death', 'cigarettes_per_day', 'weight',
                            'alcohol_history', 'alcohol_intensity', 'years_smoked', 'height', 'ethnicity', 'bmi'], inplace=True)

# Drop based on insignificance
cancerData_df.drop(columns=['case_id', 'reserved', 'blockcount', 'score', 'strand', 'chromstarts', 'samplecount',
                            'tumor_sample_barcode', 'entrez_gene_id'], inplace=True)
```

```
len(cancerData_df[cancerData_df.duplicated()])
```

```
# Getting number of unique values in each column
result_df = pd.DataFrame(cancerData_df.nunique(), columns=['No. of Unique Values']).sort_values(by='No. of Unique Values')
print(result_df.to_string())
```

```
# Save a copy of pre-processed dataset
cancerData_df.to_csv('data/project_cancerData_preprocessed.csv', index=False)
```

```
# Import pre-processed data
cancerData_df = pd.read_csv('data/project_cancerData_preprocessed.csv')
cancerData_df.describe()
```

```
# Total patients
cancerData_df['barcode'].nunique()
```

```
cancerData_df = cancerData_df[cancerData_df['cancer_type' ].str.contains(',')==False]
cancerData_df['cancer_type'] = cancerData_df['cancer_type'].apply(lambda x: x.replace('TCGA-', ''))
```

```
# Data of each cancer type
cancer_counts = cancerData_df['cancer_type'].value_counts().sample(frac=1, random_state=42)

# Graph of number of records of each cancer types
plt.figure(figsize=(12, 6))
ax = sns.barplot(x=cancer_counts.values, y=cancer_counts.index, orient='h', palette='viridis')

# Label the bars with values
for i, v in enumerate(cancer_counts.values):
  ax.text(v + 0.2, i, str(v), color='black', va='center', fontsize=8)

# Plot graph
ax.grid(True, axis='x', linestyle='--', alpha=0.7)
plt.xlabel('Number of records')
plt.ylabel('Cancer Types')
plt.title('Number of records of each cancer type')
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```

```
# Patients of each cancer type
cancer_patient_counts = cancerData_df.groupby('cancer_type')['barcode'].nunique().sample(frac=1, random_state=42)

# Graph of number of patients of each cancer types
plt.figure(figsize=(12, 6))
sns.barplot(x=cancer_patient_counts.index, y=cancer_patient_counts.values, palette='husl')


# Label each bar with its value
for i, value in enumerate(cancer_patient_counts.values):
  plt.text(i, value + 0.1, str(value), ha='center', va='bottom', fontsize=8)

# Plot graph
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.xlabel('Cancer Types')
plt.ylabel('Number of Patients')
plt.title('Number of Patients of Each Cancer Type')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

```
# Total variants
variant_counts = cancerData_df['variant'].value_counts().sample(frac=1, random_state=42)

# Graph of variant counts
plt.figure(figsize=(12, 6))
sns.barplot(x=variant_counts.values, y=variant_counts.index, orient='h', palette='flare')

# Label each bar with its value
for i, value in enumerate(variant_counts.values):
  plt.text(value + 0.1, i, str(value), ha='left', va='center', fontsize=8)

# Plot graph
plt.grid(axis='x', linestyle='--', alpha=0.7)
plt.xlabel('Number of variants')
plt.ylabel('Variant Types')
plt.title('Number of records of each variant types')
plt.tight_layout()
plt.show()
```

```
# Chrom counts
chrom_counts = cancerData_df['chrom'].value_counts().sample(frac=1, random_state=42)

# Graph of number of patients of each cancer types
plt.figure(figsize=(12, 6))
sns.barplot(x=chrom_counts.index, y=chrom_counts.values, palette='crest')

# Label each bar with its value
for i, value in enumerate(chrom_counts.values):
  plt.text(i, value + 0.1, str(value), ha='center', va='bottom', fontsize=8)

plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.xlabel('Chromosome Types')
plt.ylabel('Number of Occurrences')
plt.title('Number of Occurrences of Each Chromosome Type')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

```
# Variant type counts
variant_type_counts = cancerData_df['variant_type'].value_counts()

# Plot graph
plt.figure(figsize=(5, 5))
plt.pie(variant_type_counts, labels=variant_type_counts.index, autopct='%1.1f%%', startangle=140)
plt.title('Pie Chart of Variant Counts')
plt.show()
```

```
# Gender distribution
gender_counts = cancerData_df['gender'].value_counts()

# Plot graph
plt.figure(figsize=(5,5))
plt.pie(gender_counts, labels=gender_counts.index, autopct='%1.1f%%', startangle=140)
plt.title('Pie Chart of Gender Counts')
plt.show()
```

```
cancerData_df.drop(columns=['barcode'], inplace=True)
```

```
# Label encoding
le = LabelEncoder()

# Iterate through columns and apply label encoding
for column in cancerData_df.columns:
    if cancerData_df[column].dtype == 'object':
        cancerData_df[column] = le.fit_transform(cancerData_df[column])

class_labels = {i: label for i, label in enumerate(le.classes_)}
cancerData_df.head()
```

```
cancerData_df.info()
```

```
# Correlation of 'cancer_type' with all features
cancerData_df.corr()['cancer_type']
```

```
# Create the heatmap using the correlation matrix
plt.figure(figsize=(8, 6))
sns.heatmap(cancerData_df.corr(), linewidths=0.5)
plt.title('Correlation Heatmap of Cancer Features')
plt.show()
```

```
# Select required features
cancerData_df.drop(columns=['reference_allele'], inplace=True)
cancerData_df.head()
```

```
# Drop all duplicate rows
print(cancerData_df.shape)
cancerData_df.drop_duplicates(inplace=True)
print(cancerData_df.shape)
```

```
cancerData_df.to_csv('data/project_cancer_analysis.csv', index=False)
```

```
X = cancerData_df.drop(['cancer_type'],axis=1)
y = cancerData_df['cancer_type']
X.shape,y.shape

X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.3,
                                                    random_state=42)
X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

```
# Scale dataset
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

```
# Calculate the confusion matrix
def display_confusion_matrix(predicted, title):
    cm = confusion_matrix(y_test, predicted)

    # Create a Seaborn heatmap for the confusion matrix
    plt.figure(figsize=(10, 5))
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=class_labels.values(), yticklabels=class_labels.values())
    plt.yticks(rotation=0)
    plt.title(title)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.show()
```

```
# Cross validation
def cross_val(model, X_train, X_test, y_train, n_splits=5):
  oofs = np.zeros(len(X_train))
  preds = np.zeros(len(X_test))

  target_col = pd.DataFrame(data=y_train)

  folds = StratifiedKFold(n_splits=n_splits, shuffle=True, random_state=42)
  stratified_target = pd.qcut(y_train, 10, labels=False, duplicates='drop')

  for index, (trn_idx, val_idx) in enumerate(folds.split(X_train, stratified_target)):
    print(f'\n================================ Fold {index + 1} ================================')

    cv_X_train, cv_y_train = X_train[trn_idx], target_col.iloc[trn_idx]
    cv_X_val, cv_y_val = X_train[val_idx], target_col.iloc[val_idx]

    model.fit(cv_X_train, cv_y_train)

    val_preds = model.predict(cv_X_val)
    test_preds = model.predict(X_test)

    error = precision_score(cv_y_val, val_preds, average='macro')
    print(f'Precision is : {error}')

    oofs[val_idx] = val_preds
    preds += test_preds/n_splits

  total_error = precision_score(target_col, oofs, average='macro')
  print(f'\n Precision is {total_error}')

  return oofs, preds
```

```
def display_learning_curve(model, title):
    # Plot learning curves for the Logistic Regression model
    train_sizes, train_scores, test_scores = learning_curve(
        estimator=model,
        X=X_train,
        y=y_train,
        cv=5,
        train_sizes=np.linspace(0.1, 1.0, 10),
        scoring='accuracy'
    )

    # Calculate mean and standard deviation for training and testing scores
    train_scores_mean = np.mean(train_scores, axis=1)
    train_scores_std = np.std(train_scores, axis=1)
    test_scores_mean = np.mean(test_scores, axis=1)
    test_scores_std = np.std(test_scores, axis=1)

    # Plot the learning curves
    plt.figure(figsize=(10, 6))
    plt.title(title)
    plt.xlabel("Training Examples")
    plt.ylabel("Accuracy")
    plt.grid()

    plt.fill_between(train_sizes, train_scores_mean - train_scores_std,
                     train_scores_mean + train_scores_std, alpha=0.1, color="r")
    plt.fill_between(train_sizes, test_scores_mean - test_scores_std,
                     test_scores_mean + test_scores_std, alpha=0.1, color="g")
    plt.plot(train_sizes, train_scores_mean, 'o-', color="r", label="Training Score")
    plt.plot(train_sizes, test_scores_mean, 'o-', color="g", label="Cross-validation Score")

    plt.legend(loc="best")
    plt.show()
```

```
# Random Forest Classifier
rf_classifier = RandomForestClassifier(n_estimators=10, random_state=42)
rf_classifier.fit(X_train, y_train)
```

```
# Random Forest
rf_pred = rf_classifier.predict(X_test)
```

```
# Random Forest
rf_accuracy = accuracy_score(y_test, rf_pred)
print(f"Random Forest Classifier Accuracy: {rf_accuracy:.4f}")
```

```
# Random Forest
rf_precision = precision_score(rf_pred, y_test, average='macro')
print(f"Random Forest Classifier Precision: {rf_precision:.4f}")
```

```
# Calculate F1 score
rf_f1 = f1_score(y_test, rf_pred, average='macro')
print(f"Random Forest Classifier F1 Score: {rf_f1:.4f}")
```

```
print("\n Random Forest Classifier Classification Report:\n")
print(classification_report(y_test, rf_pred))
```

```
display_confusion_matrix(rf_pred, "Random Forest Classifier Confusion Matrix")
```

```
display_learning_curve(rf_classifier, "Learning Curves (Random Forest Classifier)")
```

```
# XGBoost Classifier
xgb_classifier = XGBClassifier(objective="multi:softmax", num_class=20, n_estimators=10, random_state=42)

# Cross validation
xgb_oofs, xgb_pred = cross_val(xgb_classifier, X_train, X_test, y_train, 5)
```

```
# XGBoost
xgb_pred = list(map(int, xgb_pred))
```

```
# XGBoost
xgb_accuracy = accuracy_score(y_test, xgb_pred)
print(f"XGBoost Classifier Accuracy: {xgb_accuracy:.4f}")
```

```
# XGBoost
xgb_precision = precision_score(xgb_pred, y_test, average='macro')
print(f"XGBoost Classifier Precision: {xgb_precision:.4f}")
```

```
# XGBoost Classifier
xgb_report = classification_report(y_test, xgb_pred)
print("\nXGBoost Classifier Classification Report:\n")
print(xgb_report)
```

```
# Calculate F1 score
xgb_f1 = f1_score(y_test, xgb_pred, average='macro')
print(f"XGBoost Classifier F1 Score: {xgb_f1:.4f}")
```

```
display_confusion_matrix(xgb_pred, "XGBoost Classifier Confusion Matrix")
```

```
num = len(cancerTypes)
# Binarize labels
y_test_bin = label_binarize(y_test, classes=np.arange(num))

# Classes
class_names = list(class_labels.values())

# Plot curve
fig, axs = plt.subplots(2, 5, figsize=(12, 5))
fig.suptitle("Precision-Recall Curves", fontsize=16)
axs = axs.flatten()

for class_label in range(num):
  # Calculate precision/recall for current class
  xgb_precision, xgb_recall, _ = precision_recall_curve(y_test_bin[:, class_label], xgb_classifier.predict_proba(X_test)[:, class_lab

  # Plot current curve
  axs[class_label].plot(xgb_recall, xgb_precision) #, label=f"{class_names[class_label]}", color='r')
  axs[class_label].set_xlabel("Recall")
  axs[class_label].set_ylabel("Precision")
  axs[class_label].set_title(f"{class_names[class_label]}")
  #axs[class_label].legend(loc='best')
  axs[class_label].grid()

plt.tight_layout(rect=[0, 0, 1, 0.97])
plt.show()
```

```
# Plot curve
fig, axs = plt.subplots(2, 5, figsize=(18, 7))
fig.suptitle("Precision-Recall and AUC-ROC Curves", fontsize=16)
axs = axs.flatten()

for class_label in range(num):
    # Calculate ROC curve for current class
    fpr, tpr, _ = roc_curve(y_test_bin[:, class_label], xgb_classifier.predict_proba(X_test)[:, class_label])

    # Calculate AUC for ROC curve
    roc_auc = auc(fpr, tpr)

    # Plot Precision-Recall curve for the current class
    axs[class_label].plot(xgb_recall, xgb_precision) #, label=f"PR Curve ({class_names[class_label]})", color='r')
    axs[class_label].plot(fpr, tpr, label=f"AUC-ROC ({class_names[class_label]}) = {roc_auc:.2f}", color='b', linestyle='--')
    axs[class_label].set_xlabel("False Positive Rate")
    axs[class_label].set_ylabel("True Positive Rate")
    axs[class_label].set_title(f"{class_names[class_label]}")
    #axs[class_label].legend(loc='best')
    axs[class_label].grid()

plt.tight_layout(rect=[0, 0, 1, 0.97])
plt.show()
```

```
display_learning_curve(xgb_classifier, "Learning Curves (Extreme Gradient Boosting Classifier)")
```

```
dtree = DecisionTreeClassifier()
%timeit dtree.fit(X_train, y_train)
```

```
dtree_pred = dtree.predict(X_test)
```

```
dtree_accuracy = accuracy_score(y_test, dtree_pred)
print(f"Decision Tree Classifier Accuracy: {dtree_accuracy:.4f}")
```

```
dtree_precision = precision_score(dtree_pred, y_test, average='macro')
print(f"Decision Tree Classifier Precision: {dtree_precision:.4f}")
```

```
# Calculate F1 score
dtree_f1 = f1_score(y_test, dtree_pred, average='macro')
print(f"Decision Tree Classifier F1 Score: {dtree_f1:.4f}")
```

```
display_confusion_matrix(dtree_pred, "Decision Tree Classifier Confusion Matrix")
```

```
dtree_report = classification_report(y_test, dtree_pred)
print("\n Decision Tree Classifier Classification Report:\n")
print(dtree_report)
```

```
display_learning_curve(dtree, "Learning Curves (Decision Tree Classifier)")
```

```
knn= KNeighborsClassifier(n_neighbors=5, metric='minkowski', p=2 )
%timeit knn.fit(X_train, y_train)
```

```
knn_pred= knn.predict(X_test)
```

```
knn_accuracy = accuracy_score(y_test, knn_pred)
print(f"K-Nearest Neighbors Classifier Accuracy: {knn_accuracy:.4f}")
```

```
knn_precision = precision_score(knn_pred, y_test, average='macro')
print(f"K-Nearest Neighbors Classifier Precision: {knn_precision:.4f}")
```

```
# Calculate F1 score
knn_f1 = f1_score(y_test, knn_pred, average='macro')
print(f"K-Nearest Neighbors Classifier F1 Score: {knn_f1:.4f}")
```

```
display_confusion_matrix(knn_pred, "K-Nearest Neighbors Classifier Confusion Matrix ")
```

```
knn_report = classification_report(y_test, knn_pred)
print("\n K-Nearest Neighbors Classifier Classification Report:\n")
print(knn_report)
```

```
display_learning_curve(knn, "Learning Curves (k-Nearest Neighbors Classifier)")
```

```
# Fitting Naive Bayes to the Training set
gnb = GaussianNB()
%timeit gnb.fit(X_train, y_train)
```

```
gnb_pred= gnb.predict(X_test)
```

```
gnb_accuracy = accuracy_score(y_test, gnb_pred)
print(f"Naive Bayes Classifier Accuracy: {gnb_accuracy:.4f}")
```

```
gnb_precision = precision_score(gnb_pred, y_test, average='macro')
print(f"Naive Bayes Classifier Precision: {gnb_precision:.4f}")
```

```
# Calculate F1 score
gnb_f1 = f1_score(y_test, gnb_pred, average='macro')
print(f"Naive Bayes Classifier F1 Score: {gnb_f1:.4f}")
```

```
display_confusion_matrix(gnb_pred, "Naive Bayes Classifier Confusion Matrix")
```

```
gnb_report = classification_report(y_test, gnb_pred)
print("\n Naive Bayes Classifier Classification Report:\n")
print(gnb_report)
```

```
display_learning_curve(gnb, "Learning Curves (Naive Bayes Classifier)")
```

```
accuracy_model = {'Random Forest':0.9116, 'XGBoost':0.9201, 'Decision Tree':0.9047,
        'KNN':0.7390,'Naive Bayes':0.9167}
models = list(accuracy_model.keys())
accuracy = list(accuracy_model.values())

fig = plt.figure(figsize = (8, 5))

# creating the bar plot
plt.bar(models, accuracy, color ='maroon',
        width = 0.6)

plt.xlabel("Machine Learning Models")
plt.ylabel("Accuracy")
plt.title("Accuracy of 5 machine learning models")
plt.show()
```

```
precisions_classifier = {'Random Forest':0.8388, 'XGBoost':0.9060, 'Decision Tree':0.9179,
        'KNN':0.6748,'Naive Bayes':0.9082}
classifiers = list(precisions_classifier.keys())
precisions = list(precisions_classifier.values())

fig = plt.figure(figsize = (8, 5))

# creating the bar plot
plt.bar(classifiers, precisions, color ='blue',
        width = 0.6)

plt.xlabel("Machine Learning Models")
plt.ylabel("Precision score")
plt.title("Precision score of 5 machine learning models")
plt.show()
```

```
f1_score_classifier = {'Random Forest':0.8568, 'XGBoost':0.8609, 'Decision Tree':0.9178,
        'KNN':0.6919,'Naive Bayes':0.9068}
classifiers = list(f1_score_classifier.keys())
f1_scores = list(f1_score_classifier.values())

fig = plt.figure(figsize = (8, 5))

# creating the bar plot
plt.bar(classifiers, f1_scores, color ='green',
        width = 0.6)

plt.xlabel("Machine Learning Models")
plt.ylabel("F1-score")
plt.title("F1 score of 5 machine learning models")
plt.show()
```