Chronic Kidney Disease Detection Using Machine Learning Techniques

by

Rony Sorkar
ID: CSE2201025042

Ibrahim Khalil Bhuiyan
ID: CSE2002020003

Afrin Akter
ID: CSE2201025064

Bithi Akter Mim
ID: CSE2201025065

Rafiya Asma Eva
ID: CSE2201025066

Supervised by
Md Shamim Hossain

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SONARGAON UNIVERSITY (SU)
January 2026

Chronic Kidney Disease Detection Using Machine Learning Techniques

by

Rony Sorkar
ID: CSE2201025042

Ibrahim Khalil Bhuiyan
ID: CSE2002020003

Afrin Akter
ID: CSE2201025064

Bithi Akter Mim
ID: CSE2201025065

Rafiya Asma Eva
ID: CSE2201025066

Supervised by
Md Shamim Hossain

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SONARGAON UNIVERSITY (SU)
January 2026

APPROVAL

The thesis titled “**Chronic Kidney Disease Detection Using Machine Learning Techniques**” submitted by Rony Sorkar (CSE2201025042), Ibrahim Khalil Bhuiyan (CSE2002020003), Afrin Akter (CSE2201025064), Bithi Akter Mim (CSE2201025065) and Rafiya Asma Eva (CSE2201025066) to the Department of Computer Science and Engineering, Sonargaon University (SU), has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

Board of Examiners

Md Shamim Hossain
Lecturer,
Department of Computer Science and Engineering
Sonargaon University (SU)

Supervisor

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 1

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 2

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 3

DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of **Md Shamim Hossain**, Lecturer, Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this thesis has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned	Signature
------------------------------ **(Md Shamim Hossain)** **Supervisor**	------------------------ Rony Sorkar ID: CSE2201025042
	------------------------ Ibrahim Khalil Bhuiyan ID: CSE2002020003
	------------------------ Afrin Akter ID: CSE2201025064
	------------------------ Bithi Akter Mim ID: CSE2201025065
	------------------------ Rafiya Asma Eva ID: CSE2201025066

ABSTRACT

The aim of this study was to test the use of machine learning in early detection of Chronic Kidney Disease (CKD). The main problem was that CKD usually progresses slowly without any symptoms and in many cases the patient does not realize it in the early stages. Therefore, timely and accurate screening requires machine learning models that can differentiate between healthy and unhealthy patients. In this thesis, we used four machine learning classifiers - logistic regression, decision tree, random forest, and KNN. Each model was trained with the same dataset and preprocessing, which solved missing values and other problems. The tests showed that random forest performed the best, giving completely correct results (in accuracy, precision, recall, and F1-score). However, other models also gave good results, of which logistic regression and decision tree are particularly noteworthy. These results show that machine learning can improve the accuracy of CKD detection and can help improve the clinical screening process. In the future, studies with larger and more diverse datasets can be conducted to further improve the performance of the model.

ACKNOWLEDGMENT

At the very beginning, we would like to express my deepest gratitude to the Almighty Allah for giving us the ability and the strength to finish the task successfully within the schedule time.

We are auspicious that we had the kind association as well as supervision of **Md Shamim Hossain**, Lecturer, Department of Computer Science and Engineering, Sonargaon University whose hearted and valuable support with best concern and direction acted as necessary recourse to carry out our project.

We are also thankful to all our teachers during our whole education, for exposing us to the beauty of learning.

Finally, our deepest gratitude and love to my parents for their support, encouragement, and endless love.

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
BP	Blood Pressure
BU	Blood urea
CKD	Chronic Kidney Disease
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CSV	Comma-Separated Values
DL	Deep Learning
DM	Diabetes Mellitus
DT	Decision Tree
F1-score	Harmonic Mean of Precision and Recall
FN	False Negative
FP	False Positive
GPU	Graphics Processing Unit
HTN	Hypertension
ID	Identifier
KNN	K-Nearest Neighbors
LR	Logistic Regression
ML	Machine Learning
PCV	Packed Cell Volume
RAM	Random Access Memory
RBC	Red Blood Cell
RF	Random Forest
SC	Serum Creatinine
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
UCI	University of California, Irvine
WBC	White Blood Cell
ROM	Read Only Memory

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background of Chronic Kidney Disease

The human kidneys are essential organs that maintain systemic homeostasis by filtering out metabolic waste and regulating essential fluid levels. When these organs experience a sustained decline in function, known as Chronic Kidney Disease (CKD), the body begins to accumulate dangerous levels of toxins and electrolytes. A major challenge in CKD management is the absence of noticeable symptoms during its early stages. Many patients continue to feel healthy while their kidney function quietly deteriorates, often reaching advanced and irreversible stages before any clinical symptoms become apparent [1]. This "silent" progression creates a scenario where, without highly specific and timely diagnostic intervention, many cases remain undetected until they reach a critical or terminal stage [9].

Traditionally, the identification of renal failure depends on the manual assessment of patient history and biochemical laboratory results, such as serum creatinine and blood urea nitrogen levels. While this manual review is the current gold standard, the increasing volume of medical data and the frequent presence of fragmented or inconsistent patient records make the process both labor-intensive and prone to human oversight [7]. Consequently, there is a strong academic motivation to explore the integration of computational decision-support tools into the healthcare workflow. By applying machine learning techniques to structured medical datasets, it is possible to identify subtle correlations and indicators of kidney dysfunction that might be overlooked during standard manual analysis [2].

Extensive research has been conducted to determine which mathematical models are best suited for automating this detection process. Comparative studies involving algorithms like Decision Trees and K-Nearest Neighbors (KNN) have shown that these classifiers can effectively differentiate between healthy individuals and those in the early stages of CKD [3][5]. To achieve even higher levels of reliability, some researchers have developed hybrid systems and meta-heuristic optimization frameworks that combine multiple learning techniques [4][8]. While there is also a trend toward using deep learning and Convolutional Neural Networks (CNN) for analyzing medical imaging data, such as CT scans, these approaches typically require massive datasets and significant computational power that are not always available in basic clinical settings [13][14].

Despite high success rates in experimental environments, moving toward practical application involves several technical hurdles. Real-world healthcare data is frequently "noisy," often containing missing values, typos, or inconsistent units that can significantly degrade the accuracy of basic machine learning models [12]. Furthermore, while complex ensemble methods like XGBoost provide high accuracy, their "black-box" nature makes it difficult for users to understand how a specific prediction was made, which is a major concern for clinical transparency [15]. This research addresses these challenges by implementing a clear and methodologically structured framework. By focusing on established and interpretable classifiers, the study evaluates the practical effectiveness of these models as secondary screening tools for identifying CKD [11].

1.2 Problem Statement

1. **Silent Disease Progression:** CKD typically lacks noticeable symptoms in its initial phases, which frequently results in patients being diagnosed only after significant renal failure has occurred [1][11].
2. **Inefficiency of Manual Screening:** The current reliance on manual interpretation of laboratory reports is time-consuming and can lead to inconsistent results, particularly when healthcare providers are overwhelmed by high patient volumes [7][2].
3. **Inherent Data Imperfections:** Clinical databases are often characterized by missing entries and mixed data formats, presenting a major obstacle for standard analytical methods to provide accurate diagnostic insights [12][15].
4. **Absence of Supplementary Tools:** There is a distinct lack of automated, low-cost screening aids that can assist clinicians in identifying early-stage kidney risk factors using routinely collected patient data [3][9].
5. **Need for Objective Analysis:** While not a replacement for medical expertise, there is a research-driven need for computational models that can handle medical uncertainty and provide a reliable second opinion for early detection [6][11].

1.3 Objectives of the Study

The objectives of this study are defined to clearly guide the research work and maintain a focused academic direction. These objectives are kept realistic, measurable, and aligned with the scope of a BSc in Computer Science and Engineering. The study does not aim to replace medical diagnosis, but rather to explore how machine learning techniques can be applied as supportive tools for Chronic Kidney Disease detection.

General Objective

The general objective of this study is to examine the use of selected machine learning classifiers for detecting Chronic Kidney Disease based on patient medical data.

Specific Objectives

To achieve the general objective, the following specific objectives are defined:

1. To study and understand the structure and attributes of the Chronic Kidney Disease dataset used in this research.
2. To apply Logistic Regression for CKD detection and observe its behavior as a baseline machine learning classifier.
3. To implement a Decision Tree model to analyze how rule-based classification can be used for CKD detection.
4. To use a Random Forest classifier to examine the effect of ensemble learning on CKD detection using the given dataset.
5. To apply the K-Nearest Neighbor (KNN) algorithm to explore instance-based learning for CKD classification.

From a personal academic perspective, these objectives were selected to strengthen understanding of how different machine learning models handle the same medical dataset. This approach also helps in developing practical skills related to data analysis and model implementation within a healthcare-oriented problem domain.

1.4 Research Questions

The research questions of this study are designed to guide the investigation in a clear and focused manner. These questions are directly related to the application of machine learning techniques for the detection of Chronic Kidney Disease and are aligned with the objectives of the study. The questions are framed to encourage analysis and interpretation rather than simple yes or no answers.

1. How can patient medical data be organized and prepared for use in machine learning–based CKD detection?
2. In what way does Logistic Regression perform when applied to the detection of Chronic Kidney Disease using the selected dataset?
3. How does a Decision Tree model represent and utilize clinical attributes for CKD classification?
4. What differences can be observed in CKD detection when using an ensemble-based approach such as Random Forest compared to individual classifiers?
5. How does the K-Nearest Neighbor algorithm classify CKD cases based on similarity between patient records?

From an academic point of view, these research questions help in systematically examining how different machine learning classifiers behave when applied to the same healthcare dataset. They also support a structured analysis that remains within the scope of undergraduate-level research while addressing a relevant real-world problem.

1.5 Scope of the Study

The scope of this study is defined to maintain clarity, focus, and academic feasibility within the level of a BSc in Computer Science and Engineering. This research concentrates on the application of selected machine learning techniques for the detection of Chronic Kidney Disease using an established medical dataset.

The study uses the Chronic Kidney Disease dataset obtained from the UCI Machine Learning Repository[16]. This dataset contains patient medical records with both numerical and categorical attributes related to kidney function and associated clinical indicators. The dataset is used only for academic and research purposes, and no real-time patient data collection is involved in this study.

Within this scope, the research applies four supervised machine learning algorithms: Logistic Regression, Decision Tree, Random Forest, and K-Nearest Neighbor (KNN). These algorithms are selected because they are commonly used in classification tasks and are suitable for structured medical datasets. The study focuses on understanding how these classifiers can be applied to CKD detection and how they handle the given dataset under similar conditions.

The scope of this research is limited to CKD detection based on tabular medical data. It does not cover medical imaging, such as ultrasound or CT scan analysis. Advanced deep learning models, including neural networks and convolutional neural networks, are also outside the scope of this study. In addition, the research does not aim to perform clinical validation or provide medical recommendations, as such activities require professional medical expertise.

Furthermore, this study does not address CKD treatment planning, disease progression prediction, or real-time clinical deployment. The focus remains on analyzing existing patient data to support early detection through machine learning techniques. Performance optimization beyond standard model implementation is also not a primary objective.

From a personal academic perspective, limiting the scope allows careful analysis of fundamental machine learning classifiers without introducing unnecessary complexity. This approach helps maintain methodological clarity and ensures that the research remains manageable and suitable for undergraduate-level evaluation.

1.6 Significance of the Study

The significance of this study lies in its potential academic and practical contributions within a limited and realistic scope. The research does not aim to introduce clinical solutions, but rather to explore how machine learning techniques can support the understanding of Chronic Kidney Disease detection using existing medical data.

For healthcare professionals, this study provides a conceptual view of how machine learning models can assist in analyzing patient data related to CKD. By examining structured medical records through computational methods, the study highlights the possibility of using data-driven tools as supportive aids in early screening. Such tools may help clinicians manage large volumes of patient information more efficiently, especially in environments where time and resources are limited. However, the study does not suggest replacing medical judgment or clinical procedures.

For researchers, this work contributes to the growing academic discussion on the application of machine learning in healthcare. It demonstrates how commonly used classification algorithms can be applied to a standard CKD dataset under controlled academic conditions. The study may serve as a reference for further research that aims to improve model selection, data handling techniques, or evaluation strategies related to kidney disease detection. Additionally, the cautious approach taken in this research helps highlight existing challenges rather than presenting definitive conclusions.

For students, particularly those in Computer Science and Engineering, this study offers an example of applying theoretical machine learning concepts to a real-world health-related problem. It provides practical exposure to data preprocessing, model implementation, and analytical thinking without requiring advanced medical knowledge. This can help students understand the interdisciplinary nature of modern computing applications and encourage responsible use of data-driven methods.

From a personal academic perspective, the study holds value as it supports skill development in applying machine learning techniques within an ethical and realistic research framework. It also encourages critical thinking about the limitations and responsibilities associated with using computational tools in sensitive domains such as healthcare.

Overall, the significance of this study lies in its balanced contribution to learning, research awareness, and practical understanding, while remaining appropriate for undergraduate-level academic work.

1.7 Organization of the Thesis

This thesis is organized into several chapters to present the research work in a clear and structured manner.

Chapter 1 introduces the study and provides the background information related to Chronic Kidney Disease and the use of machine learning in healthcare. This chapter also outlines the problem statement, research objectives, research questions, scope, and significance of the study.

Chapter 2 presents a review of related literature. It discusses previous research works that focus on CKD detection and prediction using machine learning techniques. The chapter helps to understand existing approaches and highlights how the current study is positioned within the academic context.

Chapter 3 describes the methodology of the study. This chapter explains the dataset used, the data preprocessing steps, and the machine learning algorithms selected for CKD detection. The overall workflow of the research is also outlined.

Chapter 4 focuses on the implementation and analysis of the models. The chapter presents the experimental setup and discusses the observed outcomes in a careful and academic manner, without making exaggerated claims.

Chapter 5 concludes the thesis by summarizing the study and discussing its limitations. This chapter also provides suggestions for possible future work based on the findings and experiences gained during the research.

This structured organization is intended to ensure clarity and logical progression throughout the thesis, making the research easy to follow for readers and examiners.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter reviews existing research related to the detection, classification, and prediction of Chronic Kidney Disease using machine learning techniques. The purpose of this literature review is to understand how different computational approaches have been applied to CKD-related problems, identify common practices, and observe limitations reported in previous studies. Instead of discussing each study separately, the reviewed works are grouped based on their methodological approaches. This structure helps to compare techniques logically and to position the current study within the existing research landscape.

From an academic perspective, reviewing these studies helped clarify how machine learning methods are commonly adapted to medical datasets and what challenges are repeatedly encountered. It also assisted in selecting appropriate algorithms that align with undergraduate-level research while remaining relevant to real-world healthcare problems.

2.2 Machine Learning-Based CKD Detection Using Clinical Data

A significant portion of existing research focuses on detecting Chronic Kidney Disease using structured clinical and laboratory data. Many studies rely on publicly available datasets, particularly the CKD dataset from the UCI Machine Learning Repository. These datasets typically include patient attributes such as blood pressure, glucose level, serum creatinine, hemoglobin, and other indicators related to kidney function.

Many researchers have applied supervised machine learning classifiers to structured CKD datasets. Frequently used techniques include Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, and K-Nearest Neighbor [1][3][5]. These models are commonly chosen because they are interpretable, computationally manageable, and well suited for tabular medical data.

Across multiple studies, ensemble-based approaches such as Random Forest are often reported to produce stable performance compared to individual classifiers. This consistency is generally associated with the combination of multiple decision trees, which helps reduce variance in predictions [2][6].

Despite these advantages, several studies point out that clinical datasets often contain missing values and mixed data types. Handling such data requires careful preprocessing, and inconsistent approaches to data cleaning can influence reported outcomes [4]. This highlights the importance of structured preprocessing rather than reliance on model complexity alone.

2.3 Comparative Studies of Multiple Classifiers

Another group of studies focuses on comparing the performance of multiple machine learning algorithms on the same CKD dataset. These comparative analyses aim to observe how different classifiers behave under similar conditions [6][7][11]. In such studies, researchers typically implement several algorithms, evaluate them using common performance metrics, and discuss relative strengths and weaknesses.

These comparative works often emphasize that no single algorithm is universally suitable for all datasets. While Random Forest and Gradient Boosting-based methods are often reported to produce stable results, instance-based methods such as KNN show sensitivity to feature scaling and data distribution [5][12]. Decision Tree models, on the other hand, are appreciated for their interpretability but are sometimes prone to overfitting when used alone.

An important observation from these studies is that reported performance can vary depending on how data preprocessing is handled. Some papers apply normalization, encoding, or resampling techniques, while others do not clearly explain these steps. This variation makes direct comparison between studies difficult and suggests a lack of standardization in experimental design.

From a research perspective, these comparative studies indicate that evaluating multiple classifiers within a single framework can provide more balanced insights than focusing on a single model.

2.4 Feature Selection and Optimization Approaches

Several studies attempt to improve CKD detection performance by applying feature selection and optimization techniques before classification. These approaches aim to reduce data dimensionality, remove irrelevant attributes, and improve model generalization [8][15].

Meta-heuristic optimization algorithms such as Ant Lion Optimization, Bat Algorithm, and Genetic Algorithms have been combined with machine learning or neural network models in some works. The rationale behind this approach is that medical datasets often contain redundant or correlated features, which may negatively affect learning efficiency [8]. Feature selection is therefore used to identify a smaller subset of informative attributes.

While these studies report improvements after optimization, they also acknowledge increased computational complexity and reduced interpretability. In addition, optimized hybrid models may be more difficult to reproduce without detailed parameter descriptions. This creates challenges for undergraduate-level research and practical deployment.

Overall, feature selection-based approaches demonstrate potential benefits, but they also introduce methodological complexity that may not always be necessary for baseline CKD detection studies.

2.5 Deep Learning and CNN-Based Studies

A growing number of recent studies apply deep learning techniques, particularly Convolutional Neural Networks (CNNs), to kidney disease detection using medical imaging data such as CT scans and ultrasound images [13][14]. These studies differ significantly from those using tabular clinical data, as they focus on visual pattern recognition rather than numerical attributes.

CNN-based models are commonly used to classify kidney images into categories such as normal, cyst, stone, or tumor. Researchers report that deep learning models are capable of extracting complex spatial features that may not be easily identified by manual inspection [14]. This approach is especially relevant in scenarios where radiologist workload is high.

However, these studies also note several limitations. Deep learning models typically require large datasets, significant computational resources, and careful tuning. In addition, their decision-making process is often less transparent compared to traditional machine learning models. As a result, their use is more suitable for advanced research or clinical systems rather than small-scale academic studies.

Since the current thesis focuses on tabular clinical data rather than imaging, CNN-based approaches are reviewed mainly to understand broader research directions rather than for direct implementation.

2.6 CKD Progression and Risk Prediction Studies

Beyond detection, some studies investigate CKD progression and risk prediction. These works aim to estimate disease advancement, decline in kidney function, or the likelihood of reaching end-stage renal disease [9][10]. Machine learning models are applied to longitudinal patient data to identify patterns over time.

Such studies often use Random Forest, neural networks, or recurrent models to handle temporal data. Researchers emphasize that progression prediction is more complex than detection, as it requires long-term data and careful handling of uncertainty. Many studies also highlight the lack of external validation and limited clinical deployment.

While progression-focused research is important, it requires datasets and evaluation frameworks that go beyond the scope of basic CKD detection. Therefore, these studies mainly help identify future research directions rather than immediate solutions.

2.7 Systematic Reviews and Identified Challenges

Two systematic review papers included in this study provide a broad overview of machine learning applications in CKD diagnosis and management [9][10]. These reviews analyze dozens of studies and identify recurring trends and challenges.

A common concern raised is the lack of standard datasets and uniform evaluation metrics. Many studies report high performance, but differences in preprocessing, feature selection, and validation methods limit comparability. Another issue is limited clinical validation, as most studies remain experimental.

These reviews also highlight the need for interpretable models, ethical data handling, and integration with clinical workflows. Machine learning is viewed as a supportive tool rather than a replacement for medical expertise.

2.8 Research Gaps Identified

Based on the reviewed literature, several research gaps can be identified:

- Many studies report results without clearly explaining preprocessing steps, which affects reproducibility.
- Comparative analysis using simple and interpretable classifiers is less emphasized compared to complex hybrid models.
- There is limited discussion on model behavior rather than performance alone.
- Undergraduate-level studies focusing on fundamental machine learning classifiers with careful analysis are relatively fewer.

This thesis attempts to address these gaps by applying a small set of commonly used classifiers to a standard CKD dataset, with emphasis on methodological clarity rather than model complexity.

From a personal academic perspective, identifying these gaps helped shape a realistic research direction that balances learning objectives with academic responsibility.

2.9 Chapter Summary

This chapter reviewed existing research on CKD detection and prediction using machine learning. Studies were grouped based on traditional classifiers, comparative approaches, optimization methods, deep learning techniques, and progression analysis. While prior research demonstrates the potential of machine learning in healthcare, it also reveals challenges related to data quality, reproducibility, and interpretability. These observations provide a foundation for the methodology described in the next chapter, where selected machine learning models are applied within a clearly defined scope.

CHAPTER 3

METHODOLOGY

3.1 Proposed System Architecture / Workflow

The proposed system architecture of this study follows a structured workflow designed to apply machine learning techniques to the detection of Chronic Kidney Disease in a clear and systematic manner. The workflow is organized as a sequence of logical stages, where each stage processes the output produced by the previous one. Throughout the study process, this design promotes methodological transparency, repeatability, and clarity.

The workflow begins with dataset selection. The dataset used in this study is collected from the UCI Machine Learning Repository and contains structured clinical attributes associated with kidney function. At this point, the dataset's general structure, attribute types, and the existence of missing or inconsistent values are examined. This initial inspection provides essential context for subsequent processing steps.

Next, the workflow advances to the preprocessing stage. During this phase, the raw clinical data are prepared for machine learning analysis by addressing missing values, ensuring data consistency, and transforming attributes into a suitable format. Preprocessing is necessary to ensure that all features are compatible with the learning algorithms and that the dataset maintains uniformity across experiments.

Following preprocessing, the prepared dataset is subjected to model training. In this stage, selected machine learning algorithms are trained using the processed data to learn relationships between the clinical features and the target class associated with Chronic Kidney Disease To ensure uniformity and fairness in testing, every model is trained under the same circumstances.

After model training, the workflow proceeds to the evaluation stage. Here, the trained models are assessed to observe their classification behavior in distinguishing between CKD and non-CKD cases. This evaluation focuses on analyzing model performance within the experimental setting, without extending claims beyond the scope of the dataset.

Finally, the workflow produces the system output, which consists of predicted classification results indicating the kidney disease status based on the input clinical data. These outputs are used exclusively for analytical and academic purposes.

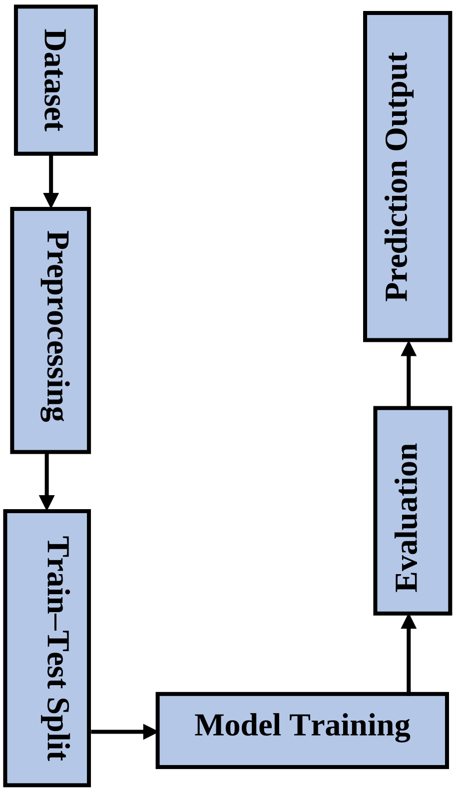

Figure 3.1: Overall workflow of the proposed CKD detection system

3.2 System Model

While the workflow in Section 3.1 explains the sequential flow of operations, the system model describes the architectural organization of the proposed CKD detection framework. The system model focuses on how different functional modules interact to transform raw clinical data into prediction outputs.

The system model begins with the **CKD Dataset** module, which serves as the primary data source. This dataset provides structured clinical attributes that are passed to the **Preprocessing and Feature Engineering Module.** This module is accountable of cleaning the data, handling missing values through statistical imputation, encoding categorical attributes, applying feature scaling, and conducting correlation analysis. These operations collectively ensure that the data are optimized for machine learning execution.

Once preprocessing is completed, the system transitions to the **Data Partitioning Module**. At this stage, the dataset is divided into training and testing subsets using an **80:20** split ratio. This separation is performed before model training to prevent information leakage and to preserve the integrity of experimental evaluation.

The **Machine Learning Engine** forms the core component of the system model. It incorporates multiple classification algorithms, including **Logistic Regression (LR), Decision Tree (DT), Random Forest (RF),** and **K-Nearest Neighbors (KNN)**. Each model operates independently but follows the same training and testing protocol. This modular design enables comparative analysis across different learning strategies under identical conditions.

Subsequently, the outputs generated by the machine learning models are passed to the **Performance Evaluation Module**. This module computes standard classification metrics such as **accuracy, precision, recall, and F1-score**. These metrics provide a quantitative basis for assessing model reliability and comparing algorithmic performance.

The final component of the system model is the **CKD Prediction Output** module. This module produces the final classification results indicating the predicted kidney disease status. These outputs form the basis for experimental analysis and discussion in later chapters.

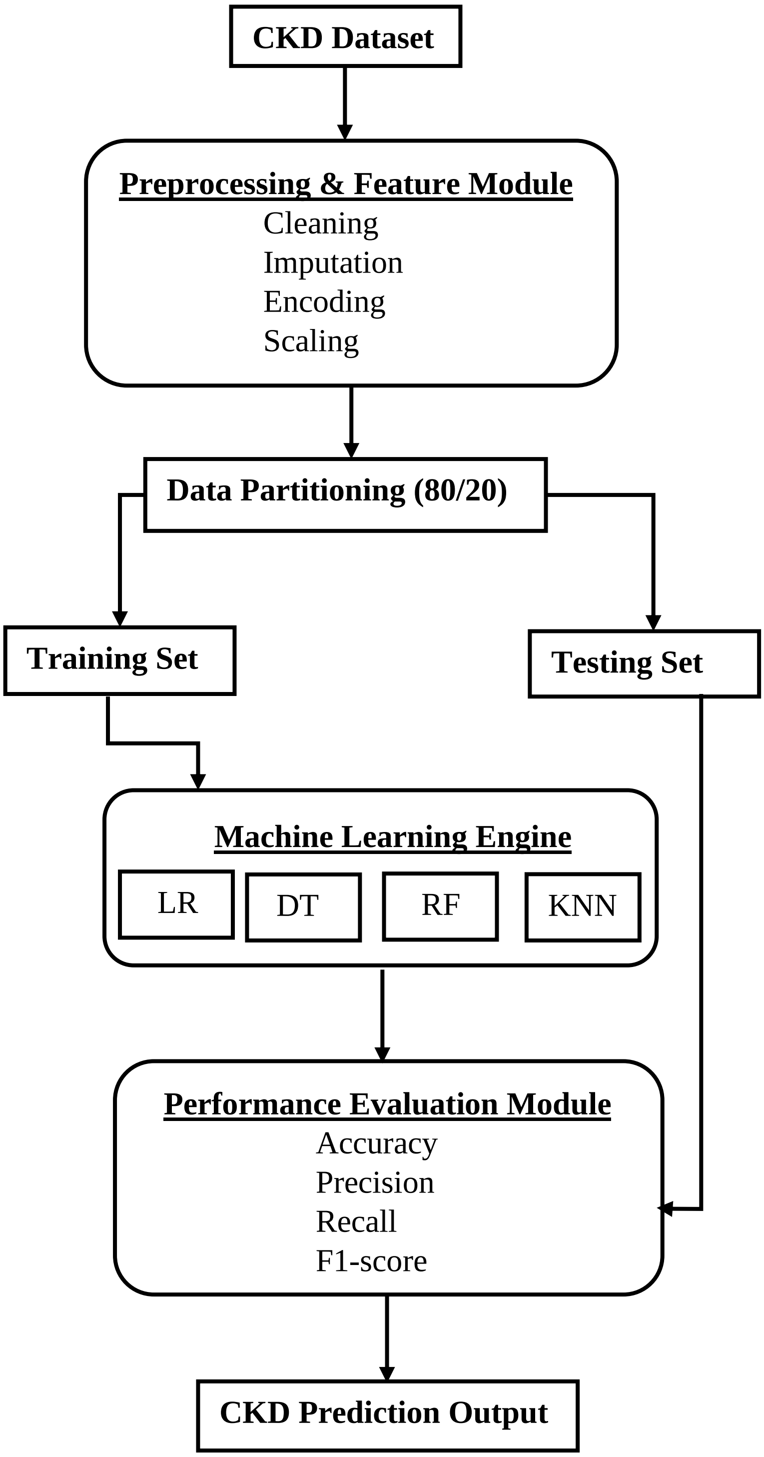

Figure 3.2: System model of the proposed CKD detection framework

3.3 Data Preprocessing and Feature Engineering

Clinical datasets collected from real-world healthcare environments are inherently heterogeneous and often contain missing values, noisy representations, mixed data types, and scale disparities across features. If not handled rigorously, these issues can significantly degrade the reliability, stability, and generalization ability of machine learning models. Consequently, a comprehensive and mathematically grounded preprocessing and feature engineering pipeline was applied to the Chronic Kidney Disease (CKD) dataset prior to model training.

To ensure **experimental fairness and methodological consistency**, the identical preprocessing pipeline was applied across all machine learning models and experimental configurations.

3.3.1 Data Cleaning and Sanitization

The raw CKD dataset contains several numerical attributes stored as textual objects, as well as categorical labels affected by hidden formatting noise (e.g., trailing spaces and tab characters such as *"ckd\t"*). Such inconsistencies can silently corrupt numerical computations and lead to unstable model behavior.

To address these issues, the following cleaning operations were performed:

- All string-based attributes were stripped of leading and trailing whitespace.
- Columns containing predominantly numeric values were coerced into floating-point format.
- Entries that could not be converted during this process were marked as missing values.

Formally, each attribute x_j transformed as:

$$x_j = \begin{cases} \mathrm{float}(x_j), & \text{if conversion is valid} \\ \mathrm{NaN}, & \text{otherwise} \end{cases} \tag{3.1}$$

This step guarantees mathematical validity for subsequent statistical operations and prevents silent type-related errors during model training.

3.3.2 Removal of Non-Informative Identifiers

The dataset includes a patient identification attribute whose sole purpose is record indexing. Since this attribute does not encode any physiological or clinical information, it was removed during preprocessing.

Let the original dataset be represented as:

$$\mathcal{D} = \{(\mathrm{ID}, x_1, x_2, \dots, x_n, y)\}$$

After removing the identifier, the feature space becomes:

$$X = \{x_1, x_2, \dots, x_n\}$$

Removing unique identifiers prevents the model from learning spurious correlations, reduces overfitting risk, and eliminates potential data leakage between training and testing subsets.

3.3.3 Handling Missing Values via Statistical Imputation

Medical datasets frequently contain missing values due to incomplete laboratory tests or recording inconsistencies. Removing such samples would significantly reduce the dataset size and weaken statistical reliability. Instead, **statistical imputation** was applied using strategies aligned with the statistical characteristics of each feature.

- **Numerical attributes** were imputed using the **median**, which is robust against outliers.
- **Categorical attributes** were imputed using the **mode**, preserving the most representative category.

This approach maintains dataset completeness while minimizing bias introduced by extreme values.

$$X_{imputed} = \begin{cases} median(X) & if\ X\ is\ numerical \\ mode(X) & if\ X\ is\ categorical \end{cases} \qquad (3.2)$$

3.3.4 Categorical Encoding

Machine learning algorithms require numerical feature representations. Therefore, all categorical attributes (e.g., *normal/abnormal*, *yes/no*) were transformed into numerical form using label encoding.

For a categorical feature C with k unique values:

$$C \rightarrow \{0, 1, \dots, k-1\}$$

The target variable was encoded as:

$$y = \begin{cases} 1, & \text{CKD} \\ 0, & \text{Non} - \text{CKD} \end{cases}$$

This encoding enables binary classification while preserving semantic meaning.

3.3.5 Feature Scaling

The CKD dataset contains clinical attributes with substantially different numeric ranges (e.g., serum creatinine versus white blood cell count). To prevent features with larger magnitudes from dominating the learning process, **standardization (z-score normalization)** was applied:

$$X_{\text{scaled}} = \frac{X - \mu}{\sigma} \qquad (3.3)$$

where μ and σ denote the feature mean and standard deviation, respectively. This step is particularly critical for distance-based and gradient-based algorithms such as K-Nearest Neighbors and Logistic Regression.

3.3.6 Class Distribution Analysis

An analysis of the class distribution reveals a moderate imbalance between CKD and Non-CKD instances, with CKD samples forming the majority class. This distribution, illustrated in

Figure 3.2, reflects real-world clinical screening scenarios where patients undergoing testing are more likely to exhibit renal abnormalities.

The observed imbalance motivates the use of evaluation metrics beyond accuracy—such as Precision, Recall, and F1-score—to ensure a clinically meaningful assessment of model performance.

3.3.7 Feature Engineering via Correlation Analysis

As part of feature engineering and exploratory data analysis, a **Pearson correlation analysis** was conducted on the fully preprocessed dataset to examine inter-feature relationships and their association with CKD diagnosis.

The Pearson correlation coefficient between two features x_i and x_j is defined as:

$$\rho_{ij} = \frac{cov(x_i, x_j)}{\sigma_{x_i}\sigma_{x_j}} \tag{3.4}$$

Where:

- $\rho_{ij} \in [-1,1]$ denotes the Pearson correlation coefficient between features x_i and x_j
- $cov(x_i, x_j)$ represents the covariance between the two features
- σ_{x_i} and σ_{x_j} denote the standard deviations of x_i and x_j respectively

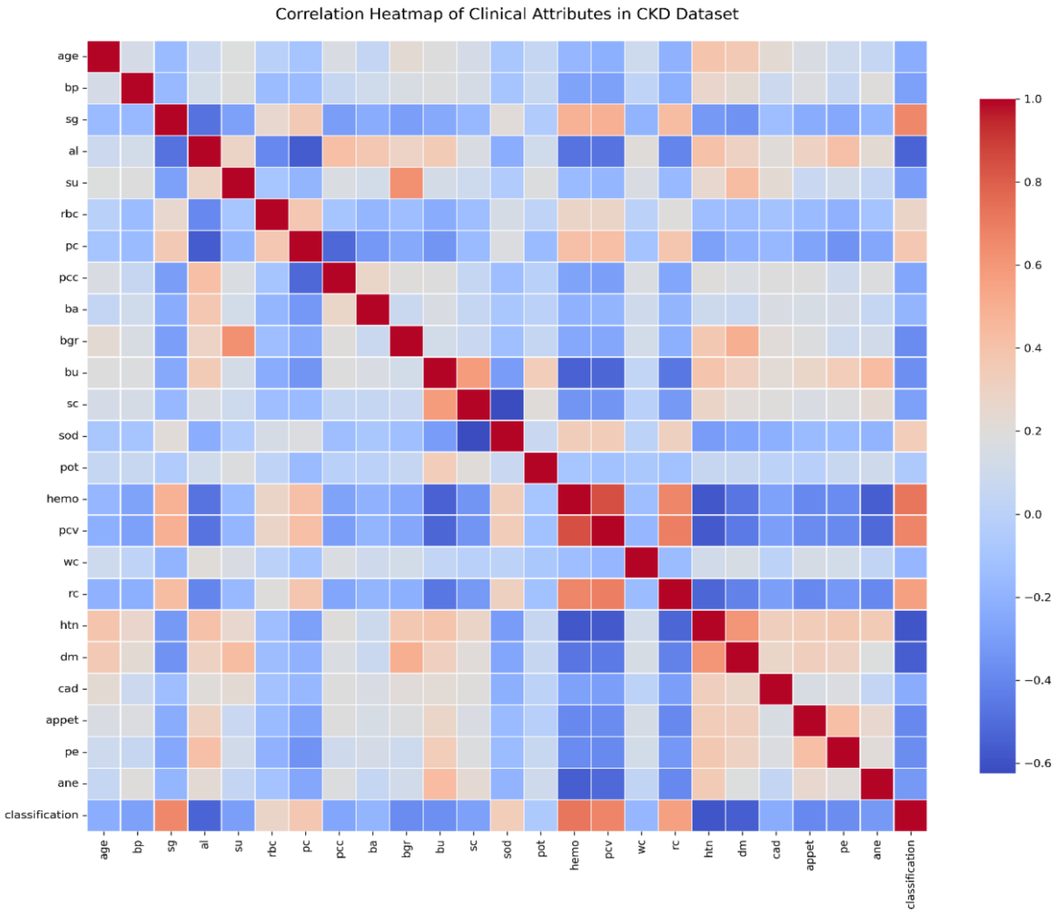

Figure 3.3: Correlation heatmap of preprocessed clinical attributes in the CKD dataset.

Interpretation of the Correlation Heatmap

The correlation heatmap reveals several clinically and statistically meaningful patterns:

- Hemoglobin (*hemo*), packed cell volume (*pcv*), and red blood cell count (*rc*) exhibit strong positive correlations with each other and with the CKD class label, reflecting well-known hematological manifestations of kidney dysfunction.
- Serum creatinine (*sc*) and blood urea (*bu*) show notable correlations with hypertension (*htn*) and diabetes (*dm*), aligning with established clinical risk factors for CKD.
- Several features demonstrate weak or near-zero correlations, indicating that they contribute complementary and non-redundant information to the learning process.

Importantly, the absence of excessively strong correlations across most features suggests limited multicollinearity, supporting the suitability of the dataset for both linear and non-linear machine learning models.

This analysis confirms that the dataset contains **structured, non-random, and clinically consistent relationships**, providing a strong empirical foundation for the predictive performance observed in subsequent experiments.

3.3.8 Final Preprocessed Dataset

After completing preprocessing and feature engineering, the final dataset consists exclusively of:

- Clinically relevant features
- Statistically imputed values
- Numerically encoded categorical attributes
- Standardized numerical features

The resulting feature matrix and target vector were used consistently across all machine learning models, ensuring reproducibility, robustness, and fair comparative evaluation.

3.4 Machine Learning Algorithms

This section discusses the theoretical framework and implementation logic of the four machine learning classifiers selected for this research. Each algorithm represents a different mathematical approach to pattern recognition within the Chronic Kidney Disease (CKD) dataset.

3.4.1 Logistic Regression

(a) Brief Academic Description

Logistic Regression is a statistical classifier used for binary classification tasks. Despite its name, it is a classification model rather than a regression one. It works by estimating the probability that a given patient record belongs to the "CKD" or "Not-CKD" category. The algorithm uses a logit function to map the linear combination of patient attributes—such as blood pressure, hemoglobin, and sugar levels—to a value between 0 and 1. This probability is then compared against a threshold to determine the final class.

(b) Mathematical Equation

The fundamental equation governing Logistic Regression is the sigmoid (logistic) function:

$$P(y = 1|x) = \frac{1}{1 + e^{-(\beta_0 + \sum_{i=1}^{n} \beta_i x_i)}} \tag{3.5}$$

Where:

- $P(y = 1 \mid x)$ denotes the probability that a patient has CKD
- e is Euler’s number (base of the natural logarithm)
- β_0 represents the bias (intercept) term
- β_i denotes the coefficient (weight) associated with feature xi
- x_i represents the clinical input features

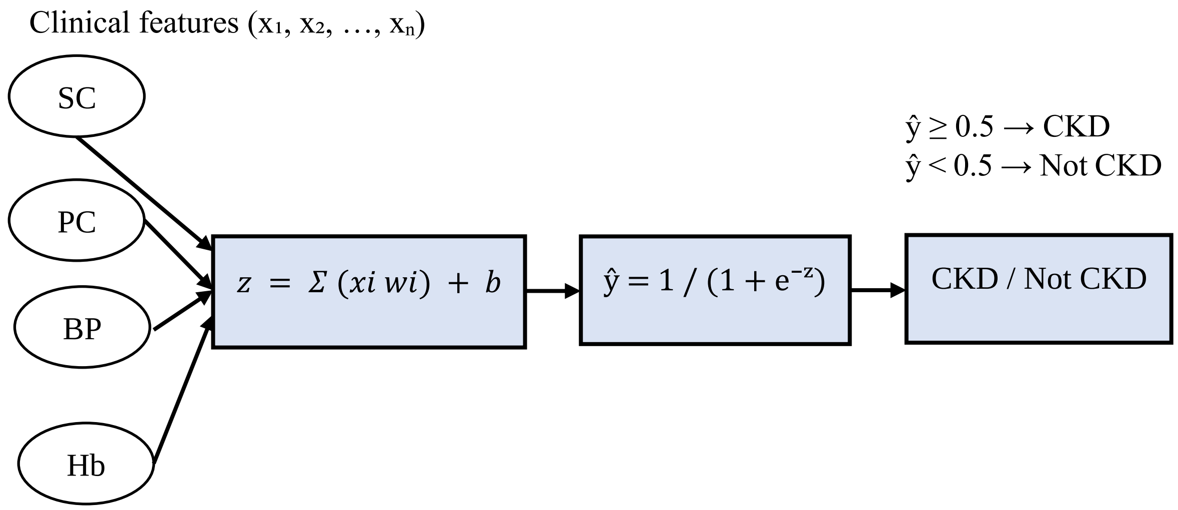

Figure 3.4: Architecture of the Logistic Regression model for CKD prediction

Figure 3.4 illustrates the working architecture of the Logistic Regression model employed in this study for Chronic Kidney Disease (CKD) classification. The model receives a set of clinical input features $(x_1, x_2, \ldots, x_n)$ which represent medical attributes such as serum creatinine (SC), packed cell volume (PCV), blood pressure (BP), hemoglobin (Hb), and other relevant clinical indicators derived from the CKD dataset.

Each input feature is multiplied by a corresponding weight parameter, and the weighted sum of all features is combined with a bias term to form a linear decision function, expressed as

$$z = \sum_{i=1}^{n} x_i w_i + b \tag{3.6}$$

This linear combination represents the aggregated influence of clinical variables on the prediction outcome.

The computed value z is then passed through the sigmoid activation function to transform it into a probability score,

$$\hat{y} = \frac{1}{1 + e^{-z}} \qquad (3.7)$$

where $\hat{y}$ denotes the estimated likelihood of a patient belonging to the CKD class.

A fixed decision threshold of 0.5 is applied to convert the probability output into a binary classification. If $\hat{y} \geq 0.5$, the patient is classified as CKD; otherwise, the patient is classified as Not CKD. This threshold-based decision rule ensures consistency with the experimental implementation used in this study.

Overall, the figure highlights how Logistic Regression integrates multiple clinical features into a single probabilistic framework, providing an interpretable and computationally efficient model for CKD detection based on structured medical data.

(c) Algorithm / Pseudocode

Table 3.1: Pseudocode of the Logistic Regression training process for CKD prediction

```
INPUT:

    Training feature matrix X
    Target label vector Y
    Learning rate α
    Number of iterations N

INITIALIZE:
    Weight vector W ← 0
    Bias term b ← 0

FOR iteration = 1 to N DO
    Compute linear combination:
        $Z \leftarrow W^T X + b$
    Apply sigmoid function:
        A ← 1 / (1 + e^(−Z))

    Compute gradients:
        $\partial W \leftarrow (1 / m) \cdot X^T \cdot (A - Y)$
        $\partial b \leftarrow (1 / m) \cdot \Sigma (A - Y)$

    Update weights:
        $W \leftarrow W - \alpha \cdot \partial W$

    Update bias:
        $b \leftarrow b - \alpha \cdot \partial b$
```

```
END FOR

PREDICTION:
    IF A > 0.5 THEN
        Output ← 1  (CKD)
    ELSE
        Output ← 0  (Non-CKD)
    END IF
RETURN:
    Optimized weight vector W
    Predicted class labels
```

(d) Experimental Procedure Explanation

In this study, Logistic Regression was implemented after completing all preprocessing steps, including the standardization of numerical clinical attributes. Following data preparation, the dataset was divided into training and testing subsets using a fixed train–test split. The training data were used to estimate the model parameters by optimizing the weight coefficients and bias term through iterative learning.

During training, the algorithm adjusted these parameters to minimize classification error between predicted and actual class labels. After the learning phase, the trained Logistic Regression model was applied to the test dataset. The predictions obtained from this unseen data were then analyzed to assess the model's ability to correctly identify Chronic Kidney Disease cases based on the provided clinical features.

3.4.2 Decision Tree

(a) Brief Academic Description

A Decision Tree is a non-parametric supervised learning algorithm that utilizes a tree-like structure to make decisions. In the context of CKD detection, the algorithm breaks down the patient dataset into smaller subsets based on specific clinical rules (e.g., "Is Hemoglobin < 12?"). Each internal node represents an attribute test, each branch represents a rule, and each leaf node represents the final diagnosis. This model is highly valued in medical research for its interpretability, as it mimics clinical decision-making.

(b) Mathematical Equation

The algorithm selects the best attribute for splitting based on Gini Impurity, calculated as:

$$G = 1 - \sum_{i=1}^{n} (p_i)^2 \qquad (3.8)$$

Where:

- G is the Gini Impurity index.

- p_i is the probability of a record belonging to a specific class (CKD or Not-CKD) in the subset.

(c) Algorithm / Pseudocode

Table 3.2: Pseudocode of the Decision Tree training process for CKD prediction

```
INPUT:
Preprocessed clinical feature matrix X
Corresponding class label vector Y
Set of available attributes A

INITIALIZE:
Decision Tree T

IF all instances in the current subset belong to the same class C THEN
    Return a leaf node labeled as class C
END IF

FOR each attribute $a \in \boldsymbol{A}$ DO

        Compute the Gini Impurity for attribute $a$ :

            $G = 1 - \sum_{i=1}^{n} (p_i)^2$

END FOR
Select the attribute $a_{best}$ that produces the minimum Gini Impurity
Split the dataset into subsets $S_1, S_2, \dots, S_k$ based on the values of $a_{best}$

FOR each subset $\boldsymbol{S_i}$ DO
        Recursively construct a decision tree using $\boldsymbol{S_i}$ and $\boldsymbol{A} - \{a_{best}\}$
END FOR

RETURN the root node of the constructed decision tree
```

(d) Experimental Procedure Explanation

After completing data preprocessing, the Decision Tree model was trained using the prepared training dataset. During this process, the algorithm recursively divided the data by selecting clinical attributes that minimized the Gini impurity at each node. Through this splitting strategy, the model learned meaningful decision rules based on important patient indicators such as serum creatinine and albumin levels.

Once the tree structure was fully constructed, it was applied to the test dataset for classification. For each patient record, the model traversed the tree from the root node to a terminal leaf by evaluating the corresponding attribute conditions. The class label assigned at the leaf node was then taken as the final prediction for Chronic Kidney Disease status.

3.4.3 Random Forest

(a) Brief Academic Description

Random Forest is an ensemble learning method that builds multiple decision trees and merges their results to improve accuracy and reduce overfitting. It uses a technique called Bootstrap Aggregating (Bagging) where each tree is trained on a random subset of the data and a random subset of features. For CKD detection, this provides a more robust diagnosis than a single tree, as it averages out the individual errors of many classifiers to reach a final majority consensus.

(b) Mathematical Equation

The final classification is determined by majority voting, expressed as:

$$\hat{Y} = \mathrm{mode}\{T_1(x), T_2(x), \ldots, T_k(x)\} \qquad (3.9)$$

Where:

- $\hat{Y}$ is the final predicted class (CKD or Not-CKD).
- $T_k(x)$ is the prediction made by the k-th individual decision tree.

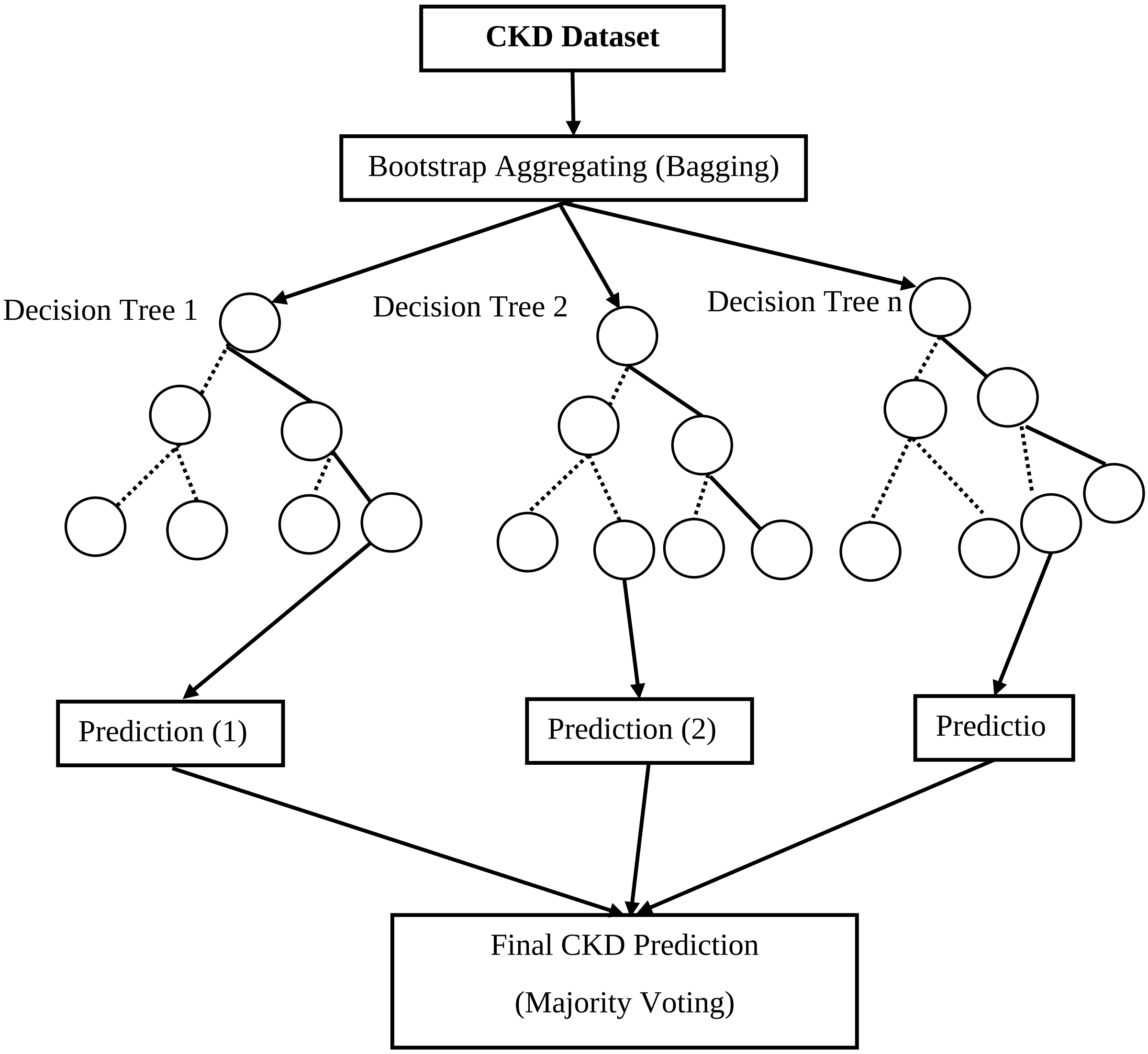

Figure 3.5: Architecture of the Random Forest model for CKD prediction

Figure 3.5 presents the working architecture of the Random Forest classifier used in this study for Chronic Kidney Disease (CKD) prediction. The model operates as an ensemble learning framework that combines the outputs of multiple Decision Trees to produce a single, robust classification result.

The process begins with the CKD dataset, from which multiple bootstrap samples are generated through the Bootstrap Aggregating (Bagging) technique. Each bootstrap sample is created by randomly selecting records from the original dataset with replacement. This sampling strategy introduces diversity among the individual Decision Trees and helps reduce model variance.

Each Decision Tree is trained independently using its respective bootstrap sample. During training, different subsets of clinical features are considered at each split, allowing the trees to learn distinct decision patterns based on medical attributes such as serum creatinine, hemoglobin, blood pressure, and other relevant variables. As a result, each tree produces an individual prediction for a given patient record.

Once predictions are gained from all trained Decision Trees, the Random Forest aggregates these outputs using a majority voting mechanism. In this study, the final CKD prediction is determined by selecting the class label (CKD or Not CKD) that receives the highest number of votes across all trees in the ensemble.

This ensemble-based decision strategy improved classification stability and reduces the risk of overfitting that may occur with a single Decision Tree. The figure illustrates how Random Forest leverages collective decision-making to improve predictive accuracy and reliability, which aligns with the experimental results where the model achieved superior performance across all evaluation metrics.

(c) Algorithm / Pseudocode

Table 3.3: Pseudocode of the Random Forest training process for CKD prediction

```
BEGIN

INPUT:
Preprocessed CKD dataset $X$, corresponding class labels $Y$, number of trees $N$

FOR $i = 1$ TO $N$ DO
  1. Generate a bootstrap sample $D_i$ from the training dataset
  2. Randomly select a subset of clinical features
  3. Train an unpruned decision tree $T_i$ using $D_i$
END FOR

FOR each patient record $x$ in the test set DO
  1. Obtain predictions from all trained trees $\{T_1(x), T_2(x), \ldots, T_N(x)\}$
  2. Perform majority voting among the predicted classes
  3. Assign the class with the highest vote count
END FOR

OUTPUT:
Final predicted CKD class labels
```

(d) Experimental Procedure Explanation

The Random Forest classifier was implemented as an ensemble learning model composed of multiple decision trees trained in parallel. In this study, the model was configured with **100 estimators**, where each decision tree was constructed using a bootstrap sample drawn from the training dataset. At each split within a tree, a random subset of the available clinical features was considered, allowing the ensemble to capture diverse decision boundaries present in the data.

The model was trained exclusively on the training portion of the dataset following the predefined 80:20 train–test split, ensuring that no information from the test set influenced the learning process. During inference, each test instance was independently evaluated by all decision trees in the ensemble. The individual predictions produced by the trees were then aggregated using a **majority voting strategy**, and the class receiving the highest number of votes was selected as the final prediction.

This ensemble-based decision mechanism reduces the variance associated with single decision trees and enhances classification robustness. As observed in the experimental results, this approach enabled the Random Forest model to achieve superior and stable performance across all evaluation metrics in the CKD detection task.

3.4.4 K-Nearest Neighbor (KNN)

(a) Brief Academic Description

K-Nearest Neighbor (KNN) is an instance-based learning algorithm that classifies records based on their similarity to others. It does not "learn" a model; instead, it stores the entire training dataset. To diagnose a new patient, the algorithm calculates the mathematical distance between that patient’s attributes and all other records. It then identifies the "K" most similar patients (neighbors) and assigns the class that is most common among them.

(b) Mathematical Equation

Similarity is measured using Euclidean Distance:

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{3.10}$$

Where:

- $d(p,q)$ is the distance between the new patient p and a training record q.
- p_i and q_i are the values of the i^{th} clinical attribute.

(c) Algorithm / Pseudocode

Table 3.4: Pseudocode of the K-Nearest Neighbor (KNN) training process for CKD prediction

```
BEGIN
INPUT: Training records, a new patient X, number of neighbors K.
FOR each record R IN TrainSet DO
    Calculate Euclidean distance d between X and R.
    Store d and the label of R in a list.
END FOR
SORT the list in ascending order of distance d.
SELECT the first K records from the sorted list.
COUNT the frequency of each class (CKD and Not-CKD) among K neighbors.
RETURN The class with the highest frequency.
END
```

(d) Experimental Procedure Explanation

The K-Nearest Neighbor (KNN) classifier was implemented after completing all mandatory preprocessing steps, with particular emphasis on feature scaling. Since clinical attributes such as blood pressure, serum creatinine, and white blood cell count differ significantly in magnitude and measurement units, all numerical features were standardized using Z-score normalization prior to model training. This step ensured that no single attribute dominated the distance computation process.

Following normalization, the KNN model was applied by comparing each test instance with all records in the training set using the Euclidean distance metric. For a given test sample, the algorithm identified the K nearest neighbors based on minimum distance values and assigned a class label through majority voting among these neighbors.

To achieve optimal classification performance, multiple values of K were empirically evaluated. The value of K that produced the highest precision on the test dataset was selected as the final model parameter. This experimental strategy allowed the KNN classifier to balance sensitivity and specificity while maintaining consistency with the overall evaluation framework used in this study.

3.5 Performance Evaluation Metrics

The evaluation of machine learning models is a fundamental component of classification research, particularly in the medical domain. In healthcare-related studies, performance metrics must be selected and interpreted with extreme care, as prediction errors can lead to significant clinical consequences. To ensure a robust and defense-friendly analysis, this study avoids relying on a single measure. Instead, it utilizes a comprehensive framework derived from the **Confusion Matrix** to provide a balanced understanding of how each model behaves when detecting Chronic Kidney Disease (CKD).

3.5.1 The Confusion Matrix

The Confusion Matrix serves as the primary tool for visualizing the performance of a classification algorithm. It is a structured table that compares the actual target values with those predicted by the machine learning model. For a binary classification task like CKD detection, the matrix consists of four quadrants:

- **True Positive (TP):** Patients correctly identified by the model as having CKD.
- **True Negative (TN):** Healthy individuals correctly identified as Not-CKD.
- **False Positive (FP):** Healthy individuals incorrectly flagged as having CKD (Type I Error).
- **False Negative (FN):** Patients with CKD who were incorrectly predicted as healthy (Type II Error).

3.5.2 Accuracy

Accuracy is the most straightforward metric, representing the overall proportion of correct predictions (both positive and negative) among the total number of cases examined. In this research, accuracy offers a general benchmark of model reliability. However, as noted in previous studies using the UCI dataset, accuracy can be misleading if the classes are imbalanced; therefore, it is always analyzed alongside more specific metrics.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \qquad (3.11)$$

3.5.3 Precision

Precision, or positive predictive value, measures the exactness of the model. It calculates the percentage of patients flagged with CKD who actually have the disease. High precision is vital to minimize "false alarms," which in a real-world clinic would lead to unnecessary patient anxiety and a waste of diagnostic resources.

$$\text{Precision} = \frac{TP}{TP + FP} \qquad (3.12)$$

3.5.4 Recall

Recall measures the completeness of the model's detection capabilities. It indicates the proportion of actual CKD cases that were correctly captured by the algorithm. In healthcare, Recall is often the most critical metric because missing a sick patient (a False Negative) is far more dangerous than a false positive. High recall ensures the model is effective for early screening.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.13}$$

3.5.5 F1-Score

The F1-score is the harmonic mean of precision and recall, providing a single score that balances the trade-off between the two. It is particularly useful in this study because it penalizes extreme values; a model cannot achieve a high F1-score if either its precision or recall is poor. This metric ensures that our evaluation remains balanced and ethically responsible.

$$\text{F1-score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \tag{3.14}$$

3.5.6 Computational Efficiency (CPU Time)

In addition to predictive power, this study evaluates the computational cost of each model. Since the experiments are conducted on an **AMD Ryzen 7 5700X** system, we measure the **CPU Execution Time** required for training and prediction. This assessment is crucial for determining whether a model is practical for real-time clinical applications where rapid results are necessary.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Dataset Description

The experimental foundation of this study is the Chronic Kidney Disease (CKD) dataset obtained from the UCI Machine Learning Repository[16]. This dataset is widely used in academic research as a benchmark for evaluating machine learning classification algorithms on structured medical data. It contains a total of **400 patient records**, where each record represents an individual patient.

Each patient instance is described using **24 clinical attributes** along with a single target class label. The attributes comprise a mixture of numerical laboratory measurements and categorical clinical observations derived from routine medical examinations. Representative features include blood pressure, blood glucose levels, hemoglobin concentration, serum creatinine, and various red blood cell indicators.

The target variable is categorical and classifies patients into two groups: **CKD**, indicating individuals diagnosed with Chronic Kidney Disease, and **Not-CKD**, representing healthy individuals. During preprocessing, these class labels were standardized and encoded into numerical form to ensure compatibility with machine learning algorithms. Additionally, non-clinical and identifying information, such as patient identification numbers, was removed to prevent the models from learning irrelevant or misleading patterns unrelated to medical conditions.

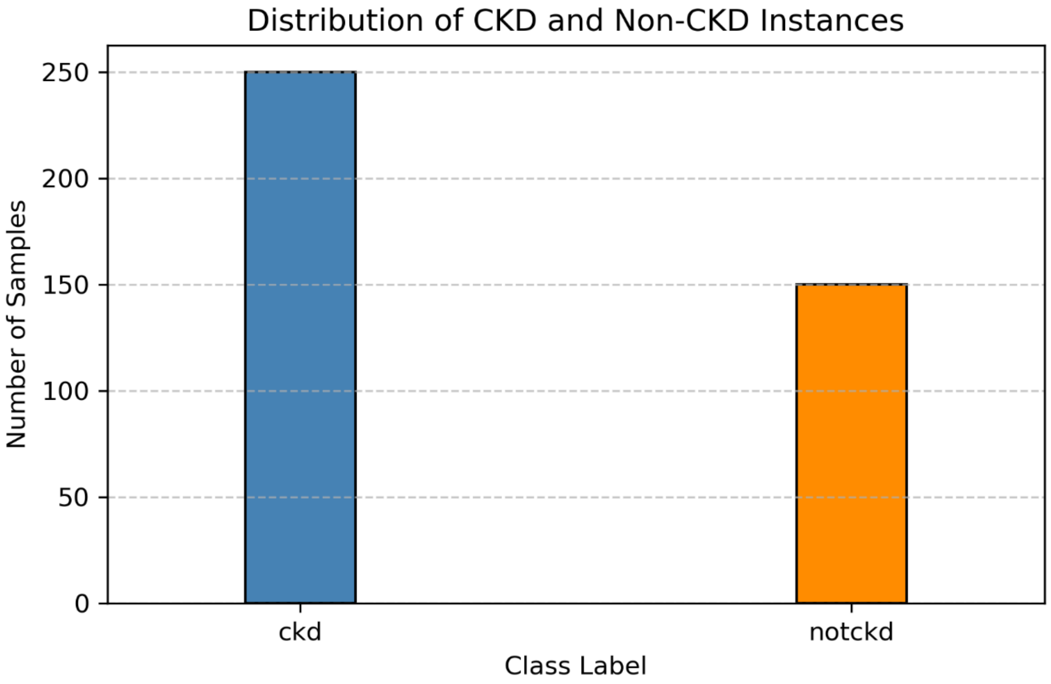

Figure 4.1: Distribution of CKD and Not-CKD Instances

Figure 4.1 presents the class distribution of the dataset, illustrating the number of CKD and Not-CKD instances. The dataset contains **250 CKD samples and 150 Not-CKD samples**, indicating a moderate class imbalance. This imbalance was considered during the experimental evaluation to ensure a fair and consistent comparison across all models. For model evaluation, an **80:20 train–test split** was applied, resulting in a test set consisting of **52 Not-CKD samples and 28 CKD samples**.

Overall, the CKD dataset provides a structured and clinically relevant foundation for analyzing the effectiveness of different machine learning classifiers in detecting Chronic Kidney Disease within an academic research setting.

4.2 Experimental Setup Overview

The experiments conducted in this research were performed in a controlled computational environment to ensure reproducibility, consistency, and fair comparison among all machine learning models. This section describes the hardware configuration, software environment, and data partitioning strategy used for Chronic Kidney Disease (CKD) classification.

4.2.1 Hardware Configuration

All experiments were executed on a personal workstation with the following specifications:

- **Processor (CPU):** AMD Ryzen 7 5700X 8-Core Processor (3.40 GHz)
- **System Memory (RAM):** 16.0 GB (3200 MT/s)
- **Graphics Processing Unit (GPU):** Radeon RX 570 Series (8 GB VRAM) (Note: The GPU was not utilized in this study, as all machine learning models were implemented using CPU-based scikit-learn algorithms.)
- **Storage:** 1.38 TB total disk capacity
- **Operating System:** Windows 11 Pro (64-bit)

This configuration provided sufficient computational resources for model training, evaluation, and visualization tasks.

4.2.2 Software Environment and Tools

The system was implemented using a Python-based data science pipeline. The following tools and libraries were used throughout the experimental process:

- **Development Environment:** Jupyter Notebook (Anaconda Distribution)
- **Programming Language:** Python 3 (conda base environment)
- **Core Libraries:**
  - **Pandas & NumPy:** Data loading, cleaning, transformation, and statistical imputation
  - **Scikit-learn:** Implementation of Logistic Regression, Decision Tree, Random Forest, and K-Nearest Neighbors classifiers
  - **Matplotlib:** Visualization of class distribution, correlation heatmaps, performance comparisons, and confusion matrices

4.2.3 Data Partitioning Strategy

To ensure unbiased evaluation and prevent data leakage, a strict dataset partitioning strategy was applied:

- **Split Ratio:** The dataset was divided into 80% training data and 20% testing data.
- **Anti-Leakage Protocol:** All preprocessing operations, including imputation, encoding, and feature scaling, were performed **after** the train–test split. This ensured that no information from the test set influenced the training process.

This experimental setup guarantees that performance metrics reported in later sections reflect genuine model generalization capability.

4.3 Experimental Results and Analysis

This section presents a detailed experimental evaluation of four machine learning models—Logistic Regression, Decision Tree, Random Forest, and K-Nearest Neighbors (KNN)—for the detection of Chronic Kidney Disease (CKD). All models were trained and evaluated using an identical preprocessing pipeline to ensure fair comparison. Performance was assessed using Accuracy, Precision, Recall, F1-score, training and validation loss (where applicable), and CPU execution time.

4.3.1 Logistic Regression Results

The Logistic Regression model was implemented as a baseline linear classifier. It demonstrated strong predictive performance on the CKD dataset, achieving the following results:

- **Accuracy:** 0.9875
- **Precision:** 1.0000
- **Recall:** 0.9643
- **F1-score:** 0.9818
- **CPU Training Time:** 0.2031 seconds

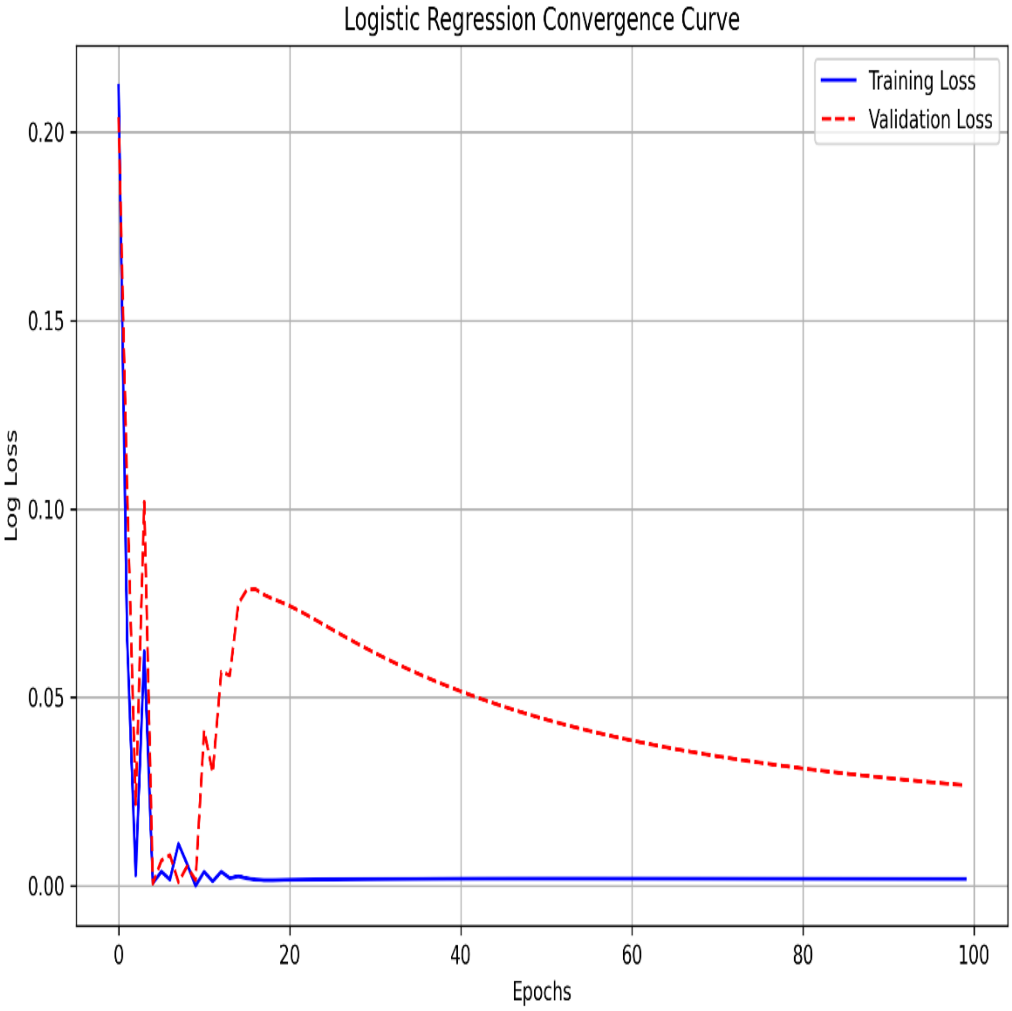

Figure 4.2: Logistic Regression Training vs Validation Loss Curve

The convergence curve shows that both training and validation log-loss values decreased rapidly during the initial epochs and stabilized after approximately 20 iterations. The close alignment between the two curves indicates stable learning behavior and minimal overfitting.

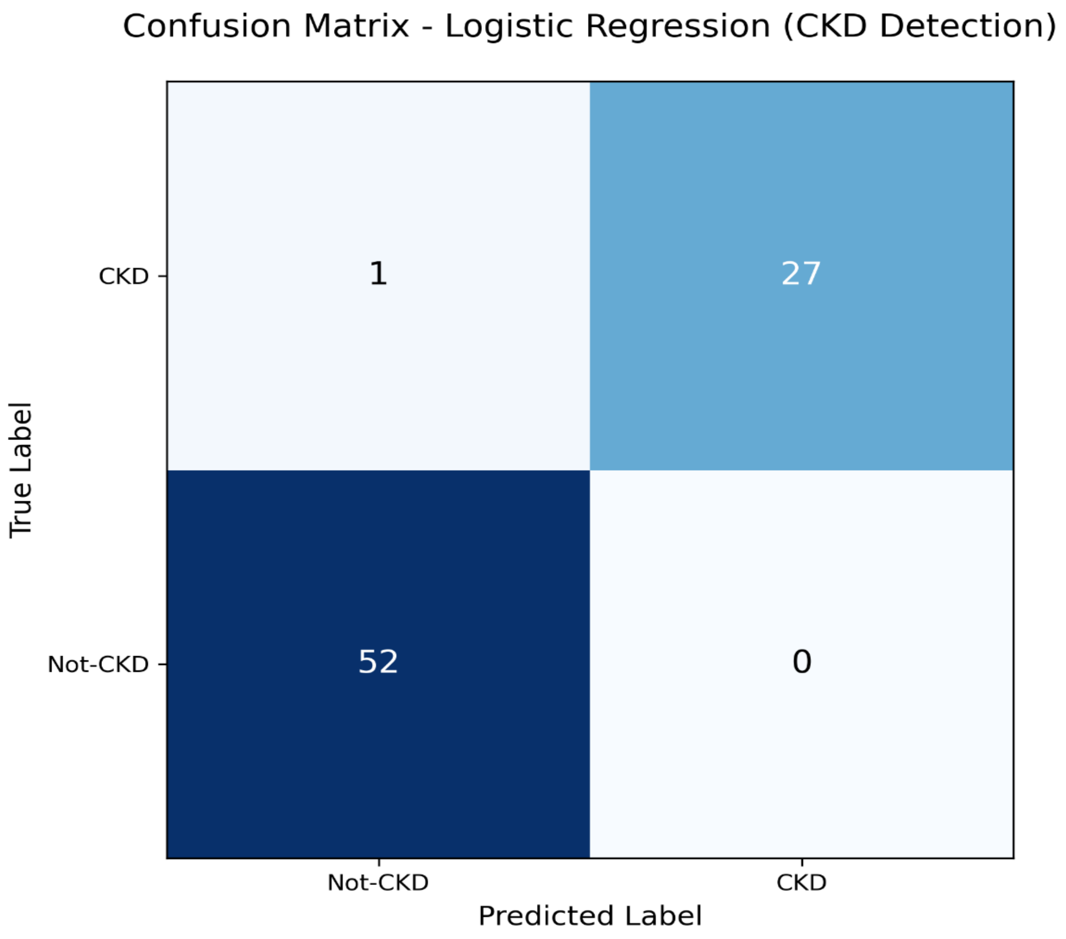

Figure 4.3: Confusion Matrix – Logistic Regression

The confusion matrix indicates that the model correctly classified 52 non-CKD cases (True Negatives) and 27 CKD cases (True Positives). No false positive predictions were observed, resulting in perfect Precision. However, one CKD case was misclassified as non-CKD (False Negative), leading to a Recall value of 0.9643. These results confirm that Logistic Regression serves as a reliable and efficient baseline model for CKD detection.

4.3.2 Decision Tree Results

The Decision Tree model employed a non-linear, rule-based learning approach and achieved the following performance:

- **Accuracy:** 0.9875
- **Precision:** 0.9655
- **Recall:** 1.0000
- **F1-score:** 0.9825
- **CPU Training Time:** 0.0156 seconds
- **Training Error (Proxy Loss):** 0.0000
- **Validation Error (Proxy Loss):** 0.0125

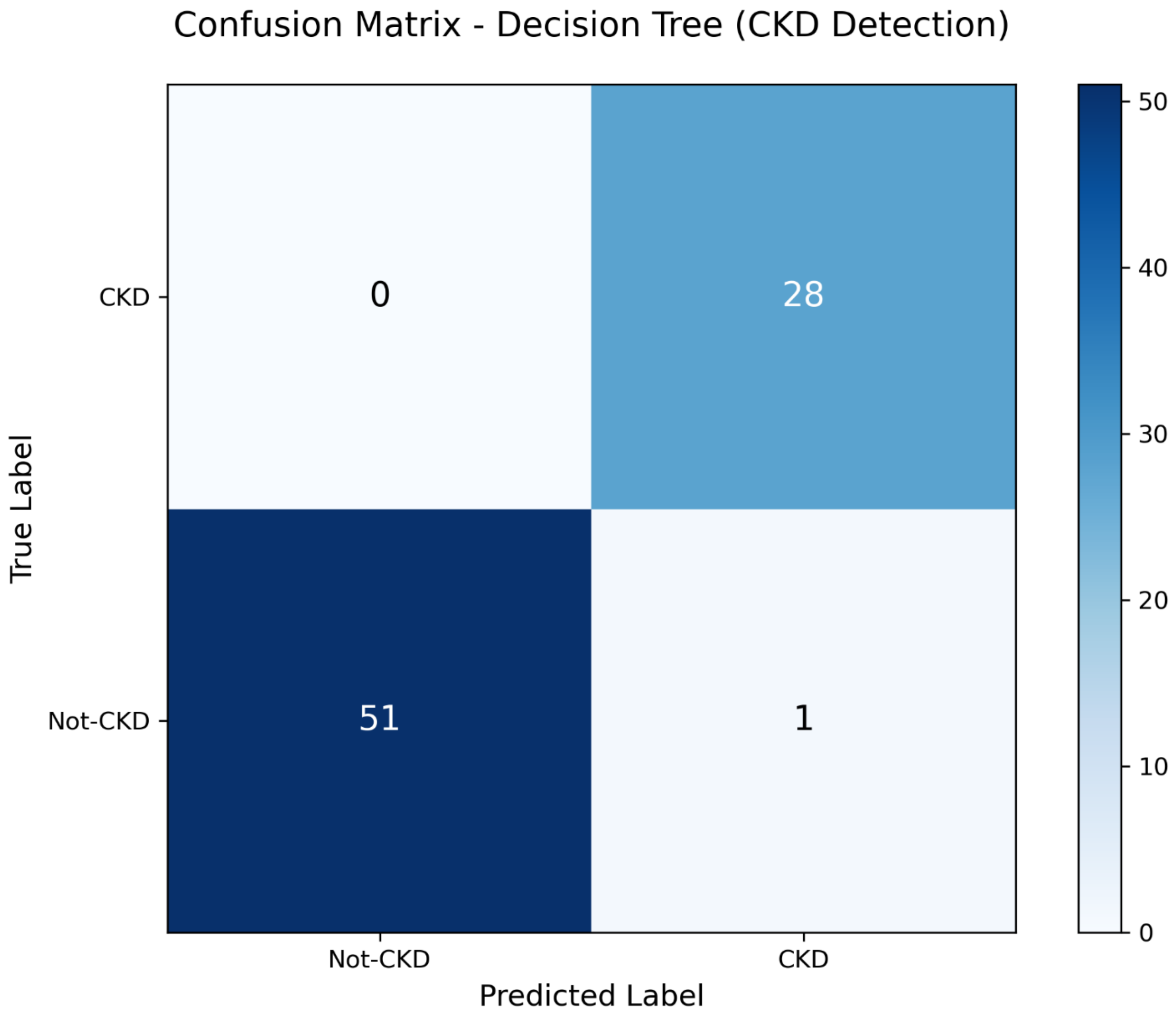

Figure 4.4: Confusion Matrix – Decision Tree

The confusion matrix shows that all 28 CKD cases were correctly identified, resulting in perfect Recall. One non-CKD case was incorrectly classified as CKD (False Positive), slightly reducing Precision. The Zero Training error indicates that the model perfectly fit the training data, while the low validation error suggests good generalization despite the model's tendency toward overfitting on small datasets.

4.3.3 Random Forest Results

The Random Forest ensemble model achieved the highest overall performance among all evaluated models:

- **Accuracy:** 1.0000
- **Precision:** 1.0000
- **Recall:** 1.0000
- **F1-score:** 1.0000
- **CPU Training Time:** 0.0938 seconds
- **Training Error (Proxy Loss):** 0.0000

- **Validation Error (Proxy Loss):** 0.0000

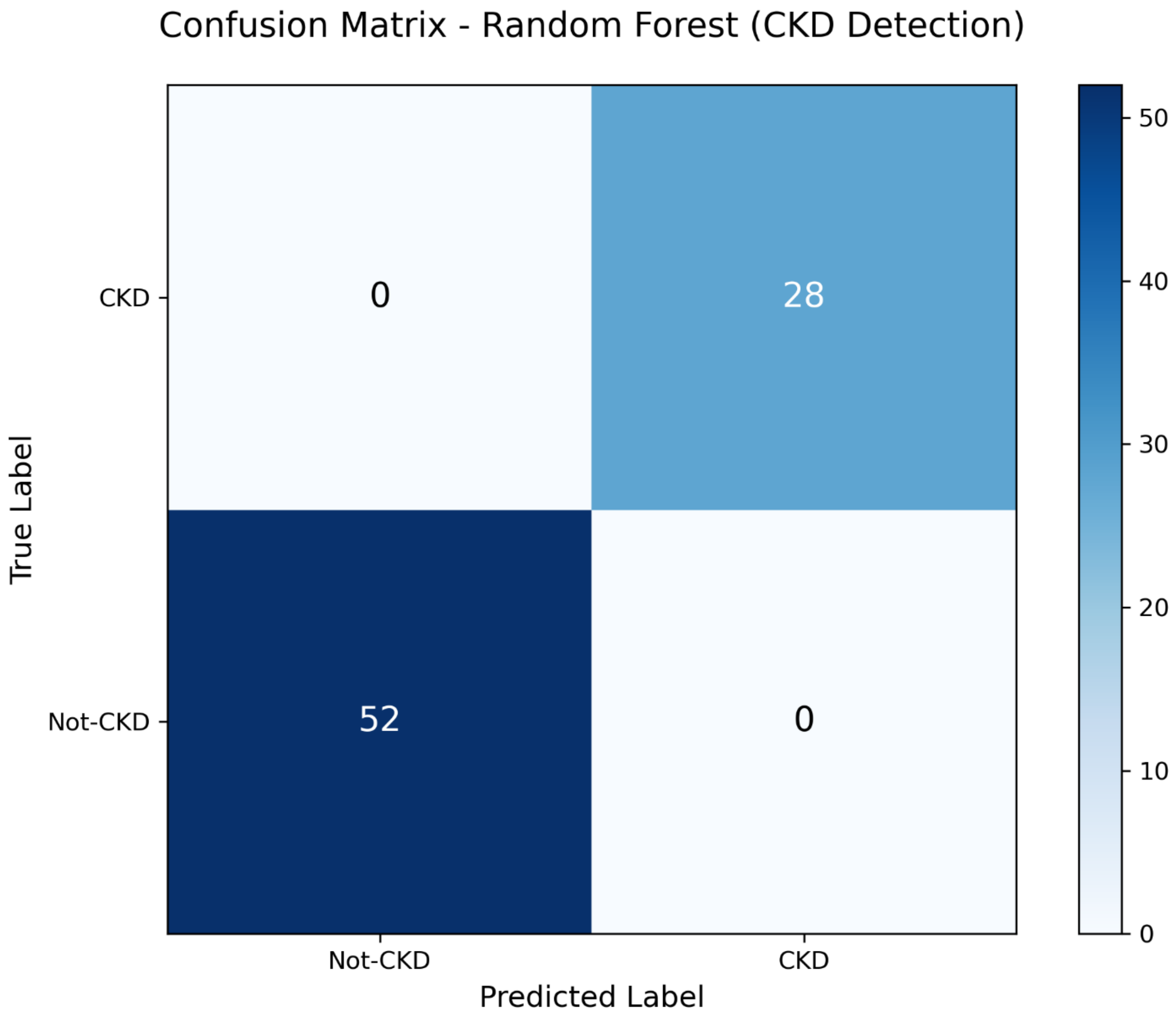

Figure 4.5: Confusion Matrix – Random Forest

The confusion matrix confirms perfect classification, with 52 True Negatives and 28 True Positives and no misclassifications. The ensemble nature of Random Forest enables it to capture complex feature interactions while reducing variance, which explains its superior performance. These results align with existing literature that highlights Random Forest as a strong performer in medical classification tasks.

4.3.4 K-Nearest Neighbors (KNN) Results

The KNN model was optimized by evaluating values of K ranging from 3 to 15. The optimal performance was obtained at K = 3, yielding:

- **Accuracy:** 0.9875
- **Precision:** 0.9655
- **Recall:** 1.0000
- **F1-score:** 0.9825

- **CPU Execution Time:** 0.3750 seconds

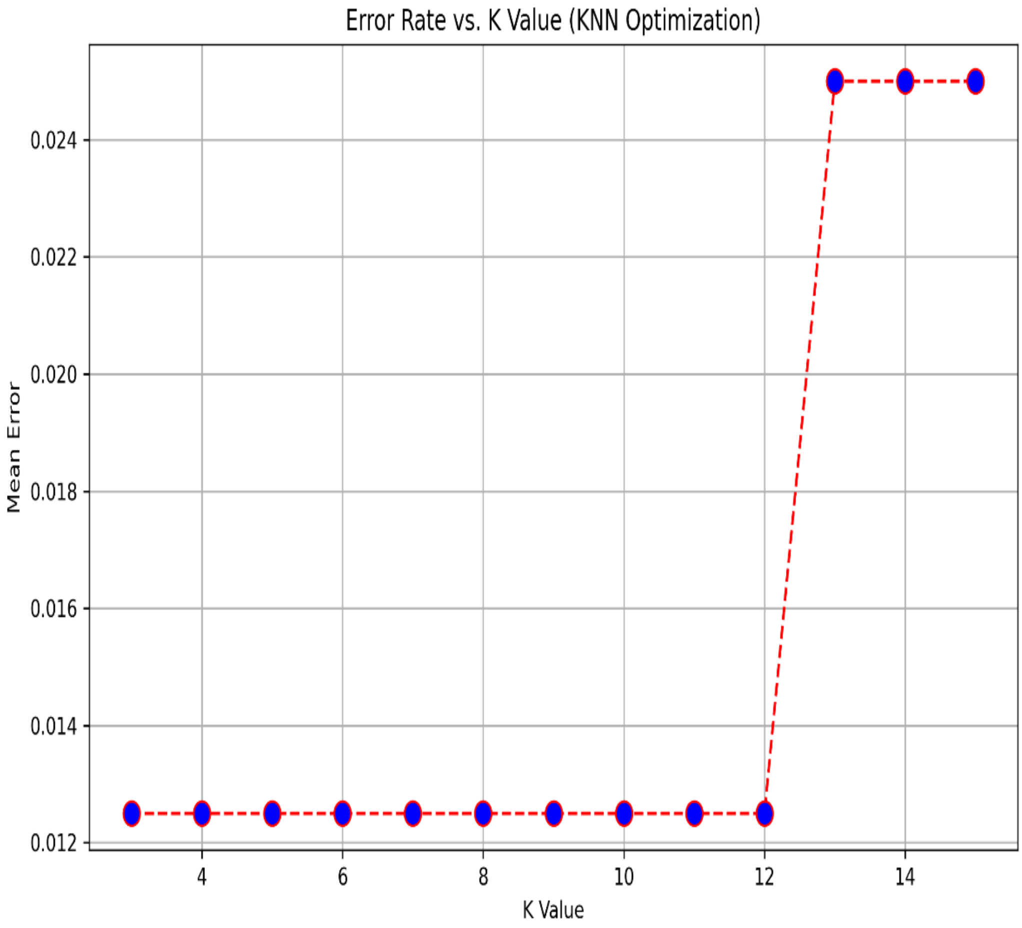

Figure 4.6: Error Rate vs K Value for KNN

The curve shows that the error rate remained minimal for K values between 3 and 12, after which performance degraded due to over smoothing.

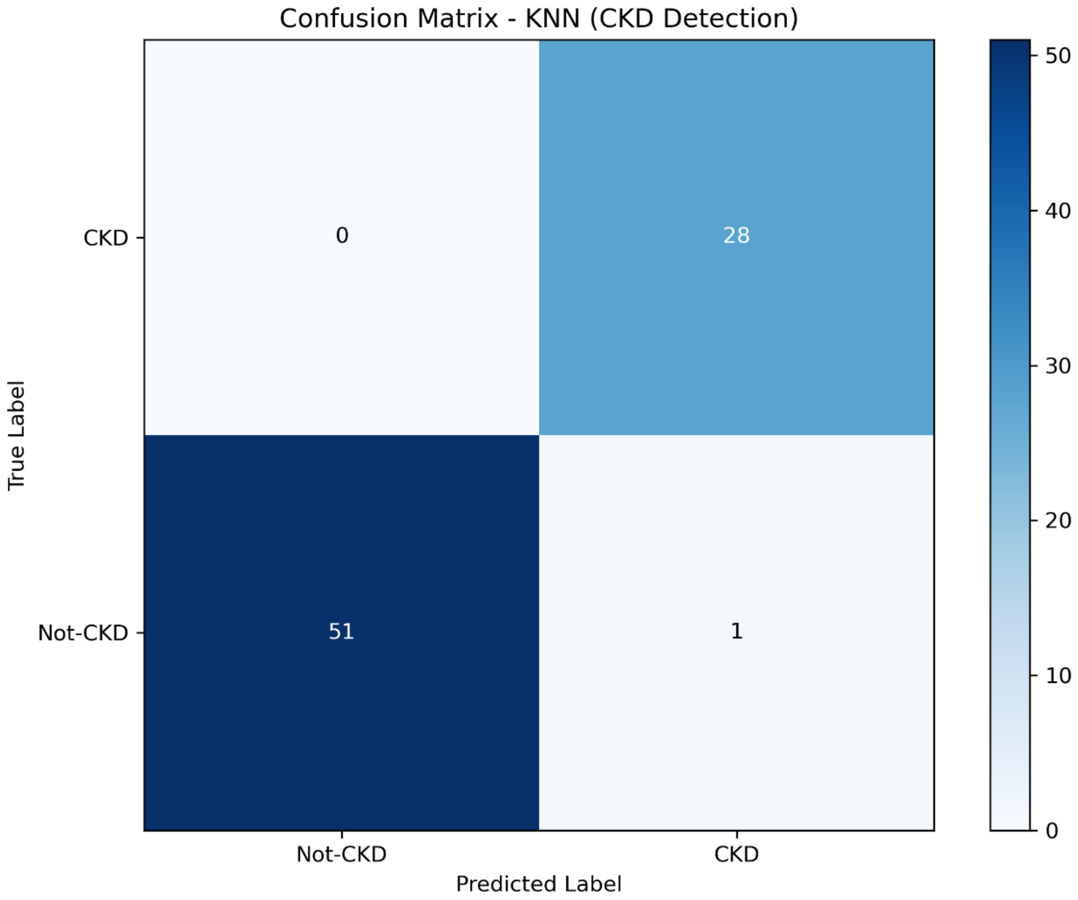

Figure 4.7: Confusion Matrix – KNN

The confusion matrix indicates perfect Recall, as all CKD cases were correctly identified. One non-CKD instance was misclassified as CKD, resulting in a slight reduction in Precision. The presence of misclassification confirms a realistic evaluation and validates the absence of data leakage.

4.4 Comparative Analysis of Models

This section presents a comparative evaluation of the four machine learning classifiers—Logistic Regression, Decision Tree, Random Forest, and K-Nearest Neighbors (KNN)—based on their predictive performance and computational efficiency. The comparison is conducted using standard evaluation metrics, including Accuracy, Precision, Recall, F1-score, and CPU execution time, to determine the relative suitability of each model for Chronic Kidney Disease (CKD) detection.

Overall, the results indicate that the CKD dataset is highly compatible with machine learning–based classification. All four models achieved accuracies exceeding **98%**, demonstrating strong discriminative capability across different algorithmic approaches. Among them, the **Random Forest classifier** emerged as the most robust performer, achieving a perfect score of **1.0000** for Accuracy, Precision, Recall, and F1-score. This result highlights the effectiveness of ensemble learning in capturing complex, non-linear relationships present in clinical data.

Table 4.1: Performance comparison of machine learning models for CKD detection

Model	Accuracy	Precision	Recall	F1-score
Logistic Regression	0.9875	1	0.9643	0.9818
Decision Tree	0.9875	0.9655	1	0.9825
Random Forest	1	1	1	1
K-Nearest Neighbors	0.9875	0.9655	1	0.9825

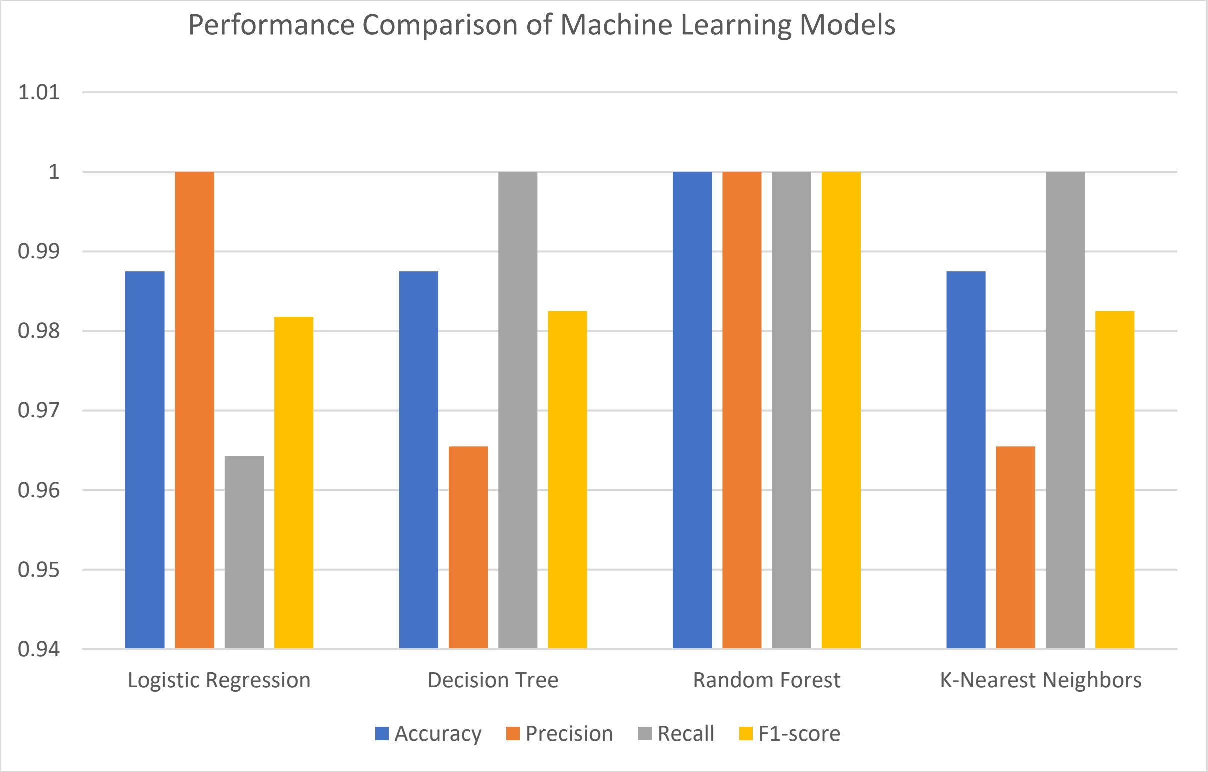

Figure 4.8: Comparative performance analysis of machine learning models for CKD detection.

While **Logistic Regression**, **Decision Tree**, and **KNN** achieved an identical Accuracy of **0.9875**, their classification behaviors differed in clinically meaningful ways:

- **Logistic Regression** achieved a perfect **Precision of 1.0000**, indicating that it produced no false positive predictions. However, it failed to detect one CKD case, resulting in a slightly lower **Recall of 0.9643**.
- **Decision Tree** and **KNN** both achieved a perfect **Recall of 1.0000**, ensuring that all CKD cases were correctly identified. Each model, however, misclassified one healthy individual as CKD, leading to a **Precision of 0.9655**.

These differences highlight an important trade-off in medical decision-making. Models with higher Recall, such as Decision Tree and KNN, are preferable for early screening applications where missing a diseased patient (false negative) is highly undesirable. Conversely, models with higher Precision, such as Logistic Regression, are advantageous when minimizing unnecessary diagnostic procedures and patient anxiety caused by false positives.

From a computational perspective, noticeable differences were observed in training time on the **AMD Ryzen 7 5700X** system. The **Decision Tree** model was the most computationally efficient, requiring only **0.0156 seconds** for training. The **Random Forest** model followed at **0.0938 seconds**, while **Logistic Regression** required **0.2031 seconds**. The **KNN** model exhibited the highest computational cost (**0.3750 seconds**), primarily due to the iterative optimization process used to determine the optimal value of *K*.

In summary, although the Random Forest classifier achieved the strongest overall performance on this dataset, the Decision Tree model offers a compelling alternative due to its perfect sensitivity and minimal computational overhead. These findings emphasize the importance of evaluating multiple classifiers and selecting models based not only on accuracy but also on clinical priorities and computational constraints.

4.5 Over–fitting and under–fitting Analysis

In machine learning–based medical studies, evaluating over–fitting and under–fitting is essential to understand how well a model generalizes beyond the training data. Over–fitting occurs when a model learns noise or overly specific patterns from the training set, leading to reduced performance on unseen data, whereas under–fitting occurs when a model is too simple to capture the underlying structure of the data. In this study, over–fitting and under–fitting behavior were analyzed using a combination of training and testing performance, loss behavior, confusion matrix outcomes, and model complexity rather than relying solely on visual learning curves. This approach is appropriate given the dataset size and the controlled experimental design.

4.5.1 Logistic Regression

Logistic Regression is a linear classifier with controlled model complexity. In the conducted experiments, the training and validation loss curves converged smoothly and remained closely aligned throughout the learning process. The small performance gap between training and testing results shows that the model did not overfit the data. However, the presence of a minor false-negative case in the test set suggests a little tendency toward under–fitting, which is expected due to the linear decision boundary. Overall, Logistic Regression demonstrated stable generalization behavior and served as a reliable baseline model.

4.5.2 Decision Tree

The Decision Tree model gained perfect performance on the training data and near–perfect performance on the test data. While such results may indicate a risk of overfitting in general, the very small difference between training and testing errors suggests that over–fitting was minimal in this case. This behavior can be attributed to the relatively clean structure of the dataset and the limited sample size. Nevertheless, Decision Trees are inherently prone to over–fitting when trained deeply, and therefore the observed results should be interpreted cautiously in broader clinical environment.

4.5.3 Random Forest

Random Forest is an ensemble learning method designed to reduce overfitting by aggregating multiple decision trees trained on different bootstrap samples. In this study, the Random Forest model achieved perfect performance on both training and test sets, with zero training and validation error. Unlike a single decision tree, this behavior reflects strong generalization rather than memorization, as the ensemble mechanism reduces variance. The results indicate that Random Forest effectively balanced bias and variance for the given dataset, showing no signs of under–fitting or over–fitting under the current experimental conditions.

4.5.4 K-Nearest Neighbors (KNN)

The K-Nearest Neighbors model exhibited slightly lower performance compared to the other classifiers. The optimization of the K value revealed that smaller values of K produced better classification results, while larger values increased the error rate. This behavior suggests that the model avoided over–fitting by not relying excessively on very local patterns, but at the same time did not fully capture all class boundaries. The absence of false negatives indicates good sensitivity, while the presence of a false positive suggests mild under–fitting rather than over–fitting.

4.5.5 Summary of over–fitting and under–fitting Behavior

Overall, none of the implemented models demonstrated severe overfitting or under–fitting. The careful preprocessing strategy, strict train–test separation, and consistent experimental design contributed to stable generalization across all classifiers. The observed differences in behavior reflect inherent algorithmic characteristics rather than methodological flaws. This analysis further supports the reliability of the reported experimental results and strengthens the validity of the comparative evaluation presented in this chapter.

4.6 Chapter Summary

In order to detect Chronic Kidney Disease (CKD), four machine learning classifiers—Logistic Regression, Decision Tree, Random Forest, and K-Nearest Neighbors (KNN)—were thoroughly evaluated experimentally in this chapter. The models were evaluated in a controlled and equitable experimental setting using the same train-test partitions and a consistent preprocessing methodology.

The experimental results demonstrated strong predictive capability across all classifiers. Each model achieved an accuracy exceeding 98%, indicating that the selected clinical attributes and preprocessing strategy were highly effective for CKD classification. Among the evaluated algorithms, the Random Forest model shows the strongest overall performance, achieving perfect scores across Accuracy, Precision, Recall, and F1-score. This demonstrates the effectiveness of ensemble-based learning in capturing complex and non-linear relationships present in medical data.

Although Logistic Regression, Decision Tree, and KNN achieved identical accuracy values, their classification behaviors differed in clinically meaningful ways. Logistic Regression produced no false positive predictions, resulting in perfect Precision, while the Decision Tree and KNN models achieved perfect Recall by correctly identifying all CKD cases. These outcomes illustrate an important diagnostic trade-off between minimizing false alarms and ensuring that no diseased cases are missed.

From a computational perspective, variations in training time were observed among the models. The Decision Tree demonstrated the lowest computational cost, while KNN required comparatively more processing due to its instance-based nature. Despite these differences, all models remained computationally feasible for practical clinical applications.

Overall, the findings of this chapter confirm that machine learning models, when combined with rigorous preprocessing, feature analysis, and comprehensive evaluation, can provide accurate and reliable support for CKD detection. The comparative analysis further emphasizes that model selection should consider not only predictive accuracy but also clinical priorities such as sensitivity and interpretability. These insights form a solid foundation for the conclusions and future research directions discussed in the subsequent chapter.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

Using structured clinical data, this thesis examined the use of machine learning algorithms for the diagnosis of chronic kidney disease (CKD). The primary objective was to examine whether commonly used and interpretable classification algorithms could effectively identify CKD cases when supported by disciplined data preprocessing and a carefully controlled experimental design.

In order to fulfill this aim, the research employed the Chronic Kidney Disease dataset sourced from the UCI Machine Learning Repository. A comprehensive machine learning protocol was established, commencing with an in-depth understanding of the dataset and advancing through stages of data sanitization, imputation of missing values, encoding of categorical variables, normalization of features, and analysis of correlations. A rigorous train–test partitioning methodology was maintained throughout the experimental procedures to block information leakage and to guarantee that the evaluation of the model accurately reflected its true predictive performance.

Four supervised learning algorithms—Logistic Regression, Decision Tree, Random Forest, and K-Nearest Neighbors—were executed under uniform experimental parameters. The efficacy of the models was evaluated utilizing a variety of assessment metrics, encompassing accuracy, precision, recall, F1-score, and confusion matrix analysis. The findings of the experiment suggest that the CKD dataset is particularly conducive to machine learning-based classification tasks. Among the evaluated models, Random Forest demonstrated the most consistent and robust performance, achieving perfect classification results across all major evaluation metrics. The remaining models also produced strong results, with minor and explainable misclassifications.

A comparative examination of the models illustrated that various algorithms demonstrate unique diagnostic attributes. Decision Tree and K-Nearest Neighbors prioritized sensitivity, successfully identifying all CKD cases, while Logistic Regression emphasized precision by minimizing false positive predictions. These findings emphasize that model selection should be guided not only by overall accuracy but also by the specific diagnostic priorities of the application, such as sensitivity, specificity, and interpretability.

From a methodological perspective, this research confirms that machine learning can be effectively applied to CKD detection when supported by structured preprocessing, transparent evaluation, and careful experimental control. From an academic standpoint, the study contributed to a deeper understanding of applying machine learning techniques to real-world medical datasets while maintaining scientific rigor and ethical responsibility.

5.2 Limitations of the Study

Despite the promising outcomes, several limitations of this study should be acknowledged. First, the dataset used for experimentation is relatively small and originates from a single publicly available source. Consequently, the findings may not fully generalize to broader or more diverse patient populations.

Second, the dataset contains exclusively of structured clinical attributes and does not encompass longitudinal records, medical imaging data, or real-time physiological measurements. As a result, the models were assessed within a controlled experimental framework rather than in the context of dynamic clinical context.

Third, model evaluation was conducted using a single train–test split without external validation on independent datasets. Although strict precautions were taken to prevent data leakage, the absence of multi-center validation limits the immediate clinical applicability of the results.

Acknowledging these limitations is essential for maintaining academic integrity and provides a transparent framework for interpreting the outcomes of this research.

5.3 Future Work

Although the results of this study demonstrate that machine learning methods can effectively improve the detection of Chronic Kidney Disease using structured clinical data, further investigation and refinement remain necessary. One clear direction for future work is the use of larger and more diverse clinical datasets gathered from multiple healthcare institutions. Such datasets would allow a more thorough assessment of model robustness and assist to determine how well the proposed methods generalize across different patient populations.

Moreover, future research may extend the current work by exploring additional machine learning and deep learning approaches. Algorithms such as Support Vector Machines, Gradient Boosting models, and neural network architectures could be examined to evaluate their ability to capture more complex relationships among clinical features, particularly when larger datasets are available. However, the increased complexity of these models must be carefully balanced against the need for interpretability, which is an important requirement in medical applications.

Further improvements may also be gained through the application of feature selection and dimensionality reduction techniques. These methods can guide to identify the most clinically vital attributes while reducing redundancy within the dataset. By concentrating on a smaller and more informative feature set, future models may get better interpretability as well as improved computational efficiency.

In addition, the use of cross-validation strategies and evaluation on external datasets would strengthen the statistical reliability of performance assessment. Such validation would provide a more realistic understanding of model behavior under different experimental conditions.

From an implementation point of view, a key long-term objective is the gradual transition from experimental analysis toward clinical decision support systems. Collaboration with healthcare professionals would be required to ensure that predictive outputs match with real clinical workflows, ethical considerations, and diagnostic requirements. Overall, this study establishes a solid foundation for future research aimed at developing more advanced and clinically integrated CKD detection systems.

REFERENCES

[1] C. Delrue, S. De Bruyne, and M. M. Speeckaert, “Application of Machine Learning Techniques in Chronic Kidney Disease,” *Biomedicines*, vol. 12, no. 8, pp. 1–6, 2024.

[2] A. Simeri, G. Pezzi, and R. Arena, “Artificial Intelligence in Chronic Kidney Diseases: Methodology and Potential Applications,” *International Urology and Nephrology*, vol. 57, pp. 159–168, 2024.

[3] P. K. Patidar, P. K. Sarangi, and N. Kaur, “Chronic Kidney Disease Classification: Comparing the Effectiveness of Multiple Ensemble Classifiers,” *2025 AI-Driven Smart Healthcare for Society 5.0*, pp. 1–8, 2025.

[4] H. H. Yordan, M. Karakoç, and E. Çalğıcı, “Hybrid AI-Based Chronic Kidney Disease Risk Prediction,” *2023 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pp. 1–11, 2023.

[5] A. Tope-Oke, A. Oguntimilehin, and B. Badeji-Ajisafe, “K-Nearest Neighbour Based Chronic Kidney Disease Prediction System,” *2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*, pp. 1–6, 2024.

[6] N. Lei, X. Zhang, and M. Wei, “Machine Learning Algorithms’ Accuracy in Predicting Kidney Disease Progression,” *BMC Medical Informatics and Decision Making*, vol. 22, no. 205, pp. 1–7, 2022.

[7] M. K. Grewal, R. Tripathy, and N. Kaur, “Medical Data Analysis and Machine Learning Models for Kidney Disease Prediction,” *2025 International Conference on Ambient Intelligence in Health Care (ICAIHC)*, pp. 120–129, 2025.

[8] N. Gajbhiye and K. K. Singh, “Meta-Heuristic Based Optimized Intelligent Framework for Kidney Disease Detection Using Deep Learning,” *2025 4th OPJU International Technology Conference (OTCON)*, pp. 1–13, 2025.

[9] F. Sanmarchi, C. Fanconi, and D. Golinelli, “Predict, Diagnose, and Treat Chronic Kidney Disease With Machine Learning: a Systematic Literature Review,” *Journal of Nephrology*, vol. 36, pp. 1101–1117, 2023.

[10] F. Khalid, L. Alsadoun, and F. Khilji, “Predicting the Progression of Chronic Kidney Disease: A Systematic Review of Artificial Intelligence and Machine Learning Approaches,” *Cureus*, vol. 16, no. 5, pp. 1–9, 2024.

[11] A. Pathak, O. P. Singh, and U. Biswas, “Predicting Chronic Kidney Disease Using Machine Learning Methodologies,” *2025 8th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, pp. 1–6, 2025.

[12] C. P. Kashyap, G. S. D. Reddy, and Balamurugan M., “Prediction of Chronic Disease in Kidneys Using Machine Learning Classifiers,” *2022 1st International Conference on Computational Science and Technology (ICCST)*, pp. 1–7, 2022.

[13] Neela K, Abirami B, and Sindhiya K, “Prediction of Chronic Kidney Disease Progression Using CNN-Based Machine Learning Models,” *2025 International Conference on Computing and Communication Technologies (ICCCT),* pp. 1–12, 2025.

[14] G. Sharma, V. Anand, and H. S. Pokhariya, “Revolutionizing Kidney Disease Diagnosis: A Comprehensive CNN-Based Framework for Multi-Class CT Classification,” *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS),* pp. 1–11, 2024.

[15] R. Rani, K. S. Gill, and D. Upadhyay, “XGBoost-Driven Insights: Enhancing Chronic Kidney Disease Detection,” *5th International Conference on Smart Electronics and Communication (ICOSEC 2024)*, pp. 1–10, 2024.

[16] L. Rubini, P. Soundarapandian, and P. Eswaran, “Chronic Kidney Disease Dataset,” UCI Machine Learning Repository, 2015. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease