Anusandhanvallari
Vol 2026, No.1
May 2026
ISSN 2229-3388

An Empirical Analysis of Machine Learning Algorithms for Chronic Kidney Disease Prediction Using NHANES Clinical Data A Comprehensive Empirical Study

[1]Yugesh B, [2]T. Anuradha

[1]Research Scholar, Department of Computer Science
Dravidian University, Kuppam, Chittoor, A.P.

[2]Professor, Department of Computer Science
Dravidian University, Kuppam
Chittoor, A.P.

Abstract

Chronic Kidney Disease (CKD) is a progressive and globally prevalent condition affecting over 850 million individuals worldwide. Early detection is critical to slowing disease progression and improving patient outcomes. This study presents a rigorous empirical evaluation of eight supervised machine learning algorithms — Gradient Boosting, Random Forest, Decision Tree, AdaBoost, Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Naive Bayes — applied to the NHANES 2021–2023 dataset for binary CKD classification. A structured preprocessing pipeline encompassing missing value imputation, outlier winsorization, feature encoding, leakage removal, and standardization was applied prior to model training. Results demonstrate that Gradient Boosting achieved superior performance with an accuracy of 98.83%, precision of 99.10%, recall of 99.22%, F1-score of 99.16%, and an ROC-AUC of 0.9986. The findings highlight ensemble tree-based methods as highly effective for CKD prediction and provide insights into feature relevance and clinical applicability of ML-driven diagnostic systems.

Keywords: Chronic Kidney Disease, Machine Learning, NHANES, Gradient Boosting, Random Forest, Classification, Clinical Prediction, Feature Engineering

1. Introduction

Chronic Kidney Disease (CKD) is characterized by a gradual loss of kidney function over time and represents one of the most significant and growing public health challenges of the 21st century. According to the Global Burden of Disease Study, CKD affects approximately 850 million people worldwide, making it more prevalent than diabetes and is rapidly increasing due to aging populations, rising rates of hypertension, and the global diabetes epidemic. The disease progresses through five stages — from mild kidney damage (Stage 1) to end-stage renal disease (ESRD, Stage 5) requiring dialysis or kidney transplantation.

Despite its severity, CKD often remains asymptomatic in its early stages. By the time patients present with clinical symptoms, irreversible kidney damage may have already occurred. Traditional diagnostic approaches rely on blood tests such as serum creatinine and estimated Glomerular Filtration Rate (eGFR), urinalysis for albuminuria, and imaging studies. While effective when conducted systematically, these approaches suffer from underutilization in primary care settings and disparities in access across socioeconomic groups.

The intersection of machine learning and clinical medicine offers an opportunity to transform CKD detection by enabling high-throughput, data-driven screening of large patient populations. The National Health and Nutrition

Examination Survey (NHANES), administered by the U.S. Centers for Disease Control and Prevention (CDC), provides a rich nationally representative dataset encompassing clinical, demographic, dietary, and laboratory variables, making it an ideal resource for developing CKD prediction models with real-world generalizability.

Previous studies have explored subsets of ML algorithms on CKD datasets, often using the UCI repository dataset which, while widely cited, is smaller and less representative than NHANES data. This paper addresses that gap by systematically evaluating eight distinct classification algorithms on the NHANES 2021–2023 CKD-staged dataset, applying a carefully designed preprocessing pipeline to ensure data integrity, and benchmarking models across five performance metrics.

1.1 Motivation and Problem Statement

The central problem addressed in this study is: given a set of clinical and demographic features extracted from population health surveys, can machine learning algorithms reliably distinguish individuals with CKD from those without, and which algorithmic approach offers the best balance of predictive accuracy, sensitivity, and generalizability for clinical deployment?

Answering this question requires not only training and evaluating models but also addressing the significant preprocessing challenges inherent in population health datasets — particularly high rates of missingness, measurement outliers, class imbalance, and risk of data leakage from clinically derived staging variables.

1.2 Objectives

- To perform systematic preprocessing of the NHANES 2021–2023 dataset, including missing data handling, outlier treatment, and feature encoding.
- To train and evaluate eight supervised machine learning classifiers for binary CKD prediction.
- To compare algorithm performance using accuracy, precision, recall, F1-score, and ROC-AUC metrics.
- To analyze confusion matrices and ROC curves for the best-performing models.
- To discuss clinical implications and limitations of the proposed approach.

1.3 Contribution

This study makes the following contributions to the literature: (1) a comprehensive, step-by-step ML pipeline for NHANES CKD data; (2) an empirical comparison of eight ML algorithms under identical experimental conditions; (3) identification of Gradient Boosting as the dominant algorithm with near-perfect classification performance; and (4) discussion of practical considerations for clinical translation of CKD prediction systems.

2. Background and Related Work

2.1 Chronic Kidney Disease: Clinical Overview

Chronic Kidney Disease is clinically defined as abnormalities in kidney structure or function, present for more than three months. It is staged according to eGFR values and the albumin-to-creatinine ratio (ACR): Stage 1 (eGFR $\geq$ 90 mL/min/1.73m^2), Stage 2 (eGFR 60–89), Stage 3a (eGFR 45–59), Stage 3b (eGFR 30–44), Stage 4 (eGFR 15–29), and Stage 5 (eGFR < 15). Markers such as serum creatinine, blood urea nitrogen (BUN), serum albumin, phosphorus, bicarbonate, and calcium reflect kidney function and are often dysregulated in CKD.

Key risk factors include Type 2 diabetes (responsible for approximately 40% of CKD cases), hypertension (approximately 30%), glomerulonephritis, and hereditary conditions such as polycystic kidney disease.

Additionally, socioeconomic factors — including poverty income ratio, educational attainment, and ethnicity — have been shown to significantly influence CKD prevalence and outcomes, underscoring the importance of including these variables in predictive models.

2.2 Machine Learning in Healthcare Prediction

Machine learning methods have demonstrated consistent promise in clinical prediction tasks. Logistic Regression has historically served as the baseline for binary medical classification due to its interpretability and probabilistic output. Decision Trees provide transparent rule-based classification but are prone to overfitting. Ensemble methods such as Random Forest and Gradient Boosting address this limitation by aggregating predictions across multiple weak learners, substantially improving generalization. Support Vector Machines excel in high-dimensional feature spaces by identifying optimal separating hyperplanes, while K-Nearest Neighbors provides a simple non-parametric approach based on feature similarity. Naive Bayes, despite its independence assumption, has shown surprising efficacy in medical data with limited training samples. AdaBoost iteratively focuses on misclassified samples, making it robust for imbalanced datasets.

2.3 Prior Work on CKD Prediction

Polat et al. (2017) applied multiple ML classifiers to the UCI CKD dataset, reporting accuracies ranging from 87% to 99% depending on the algorithm and preprocessing approach. Salekin and Stankovic (2016) used a Random Forest with ten-fold cross-validation on UCI CKD data, achieving 99.3% accuracy. Islam et al. (2020) compared Logistic Regression, SVM, and Random Forest on UCI CKD data, finding ensemble methods consistently superior. However, these studies share a common limitation: the UCI CKD dataset contains only 400 records and is not representative of the broader U.S. population. Studies using NHANES data for CKD prediction are fewer. Norouzi et al. (2021) applied Gradient Boosting on NHANES 2015–2016 data, achieving 94% AUC, but did not compare across multiple algorithms systematically. The present study extends this body of work by using the most recent NHANES 2021–2023 cycle with 12,387 participants and employing a broader algorithmic comparison under controlled conditions.

Table 1. Summary of Representative Prior Work on CKD Prediction Using Machine Learning

Study	Dataset	Algorithms	Best Accuracy	AUC
Polat et al. (2017)	UCI CKD (400)	LR, SVM, DT, NB	99.0%	0.990
Salekin & Stankovic (2016)	UCI CKD (400)	Random Forest	99.3%	0.995
Islam et al. (2020)	UCI CKD (400)	LR, SVM, RF	98.5%	0.991
Norouzi et al. (2021)	NHANES 2015–16	Gradient Boosting	95.2%	0.940
Present Study (2024)	NHANES 2021–23	8 Algorithms	98.83%	0.9986

Table 1. Summary of representative prior studies on machine learning-based CKD prediction.

3. Dataset Description

3.1 NHANES 2021–2023 Overview

The National Health and Nutrition Examination Survey (NHANES) is a cross-sectional nationally representative survey conducted by the U.S. Centers for Disease Control and Prevention (CDC). It combines interview and physical examination data and has been conducted in two-year cycles since the 1960s. The 2021–2023 cycle, used in this study, enrolled a nationally representative sample of the non-institutionalized U.S. civilian population.

The specific CKD-staged dataset used in this study was obtained from Kaggle (https://www.kaggle.com/datasets/alitaqishah/ckd-nhanes-2021-2023-staged-kidney-disease) and contains clinical, demographic, and laboratory data for 12,387 participants, with binary CKD status (ckd_present: 0 = No CKD, 1 = CKD) derived from eGFR and ACR values consistent with KDIGO 2012 guidelines.

3.2 Feature Set

The raw dataset contains 25 features spanning four domains: demographic characteristics, anthropometric measurements, clinical/laboratory values, and lifestyle factors. Table 2 provides a categorized overview of all features included in the raw dataset prior to preprocessing.

Table 2. Feature Categories in the NHANES 2021–2023 CKD Dataset

Category	Features	Clinical Relevance
Demographic	age, gender, ethnicity, education_level, poverty_income_ratio	Socioeconomic risk factors for CKD
Anthropometric	bmi, weight_kg, height_cm	Obesity-related CKD risk
Cardiovascular	bp_systolic, bp_diastolic	Hypertension as CKD risk/cause
Renal Lab Markers	serum_creatinine, egfr, blood_urea_nitrogen, albumin_creatinine_ratio, urine_creatinine, urine_albumin	Direct indicators of kidney function
Metabolic Lab	albumin_serum, phosphorus, bicarbonate, calcium, uric_acid	Secondary markers of CKD complications
Endocrine/Lifestyle	diabetes_diagnosed, ever_smoked, insulin_use, diabetes_pills, current_smoker	CKD risk factor and comorbidity markers
Target Variable	ckd_present	Binary: 0 = No CKD, 1 = CKD Present

Table 2. Features organized by clinical category with notes on relevance to CKD prediction.

3.3 Class Distribution

As illustrated in Figure 1 (Target Distribution), the dataset exhibits moderate class imbalance. Approximately 3,500 participants (28.3%) were classified as CKD-negative (ckd_present = 0) while approximately 8,800 participants (71.7%) were classified as CKD-positive (ckd_present = 1). This distribution reflects the high CKD burden in the NHANES population, which over-samples certain demographic groups at elevated risk. The imbalance was noted and addressed through stratified train-test splitting rather than resampling, to preserve the natural population distribution.

4. Methodology

4.1 Research Framework

The study follows a structured experimental design comprising six sequential phases: (1) data acquisition and exploratory analysis, (2) preprocessing pipeline execution, (3) feature engineering and encoding, (4) model training and validation, (5) performance evaluation and comparison, and (6) visualization and interpretation. All experiments were conducted using Python 3.11 with scikit-learn 1.3, pandas 2.0, numpy 1.24, seaborn 0.12, and matplotlib 3.7.

4.2 Data Preprocessing Pipeline

The preprocessing pipeline was designed with rigor to prevent data leakage and ensure reproducibility. Six discrete preprocessing steps were executed sequentially, each with corresponding visualization to audit the transformation effects.

Step 1: Missing Value Analysis and Column Removal

The percentage of missing values was calculated for every feature. Features with more than 60% missing observations were removed from the dataset to avoid introducing excessive imputation bias. As visualized in Figure 1 of the original analysis, features including insulin_use, diabetes_pills, current_smoker, phosphorus, and bicarbonate exceeded this threshold and were dropped. This threshold was selected based on established practice in clinical ML literature, which recommends discarding features with more than 50–60% missingness.

Step 2: Imputation of Remaining Missing Values

For remaining features, missing numerical values were imputed using the median (via sklearn's SimpleImputer with strategy='median'). Median imputation is preferred over mean imputation in clinical datasets due to its robustness to skewed distributions and extreme outliers, which are common in laboratory measurements. Categorical variables were imputed using the most frequent value (mode). Figure 2 (BMI Distribution) demonstrates that median imputation successfully preserved the original BMI distribution shape while filling missing values.

Step 3: Outlier Handling via Winsorization

Extreme outliers in key renal biomarkers — specifically serum_creatinine, blood_urea_nitrogen, eGFR, and albumin_creatinine_ratio — were treated using IQR-based winsorization. This technique clips values beyond Q1 - 1.5*IQR and Q3 + 1.5*IQR to the respective fence values, rather than removing them, preserving sample size while mitigating the influence of biologically implausible extreme values. As shown in Figure 3, winsorization of serum creatinine successfully removed the cluster of extreme outliers (values exceeding 10 mg/dL) while maintaining the physiologically plausible distribution range.

Step 4: Leakage Removal and Feature Encoding

The ckd_stage variable was removed prior to modeling, as it is a direct derivation of the eGFR and ACR thresholds used to define ckd_present, constituting a form of target leakage. The participant_id variable was removed as it carries no predictive information. The gender variable was encoded using Label Encoding (0/1), and the multi-class ethnicity variable was one-hot encoded using pd.get_dummies, yielding six binary ethnic group indicators.

Step 5: Feature Scaling

All numeric features (excluding the binary target) were standardized to zero mean and unit variance using sklearn's StandardScaler. This step is essential for distance-based algorithms (KNN, SVM) and regularized models (Logistic Regression) that are sensitive to feature magnitude differences. Tree-based methods (Random Forest, Gradient Boosting, Decision Tree, AdaBoost) are scale-invariant; however, standardization was applied uniformly for consistency across model comparisons.

Step 6: Train-Test Split

The dataset was divided into training (80%) and test (20%) subsets using sklearn's train_test_split with stratify=y to maintain the class distribution ratio in both splits. A fixed random_state of 42 was used to ensure reproducibility. No validation set was used for hyperparameter tuning; all models were evaluated using their default scikit-learn hyperparameters to provide a fair baseline comparison.

Table 3. Summary of Preprocessing Steps Applied to the NHANES CKD Dataset

Step	Operation	Method	Rationale
1	Missing Value Removal	Drop cols > 60% missing	Prevent high-imputation bias
2	Imputation	Median (numeric), Mode (categorical)	Robust to outliers and skewness
3	Outlier Treatment	IQR-based Winsorization	Preserve n while limiting extremes
4	Encoding & Leakage	LabelEncoder, get_dummies; drop ckd_stage	Remove multicollinearity/leakage
5	Scaling	StandardScaler (Z-score)	Equalize feature magnitudes
6	Data Split	80/20 stratified split	Preserve class balance

Table 3. Structured preprocessing pipeline applied before model training.

4.3 Machine Learning Algorithms

Eight supervised classification algorithms were selected to represent a broad spectrum of methodological approaches, from simple probabilistic models to complex ensemble learners:

4.3.1 Logistic Regression

Logistic Regression models the log-odds of CKD presence as a linear function of the input features, optimized via maximum likelihood estimation. It serves as the interpretable baseline model and provides probabilistic class membership outputs. Configured with max_iter=1000 to ensure convergence on the scaled feature set.

4.3.2 Decision Tree

A single Classification and Regression Tree (CART) was trained with maximum depth constrained to 5 levels to prevent overfitting while retaining sufficient complexity. The Gini impurity criterion was used for split selection.

The depth constraint is a critical hyperparameter for Decision Trees used in clinical settings, as it directly controls model interpretability.

4.3.3 Random Forest

An ensemble of 100 decision trees trained via bagging (bootstrap aggregation). Each tree is trained on a random bootstrap sample with random feature subsets considered at each split node. The ensemble prediction is the majority vote. Random Forests are highly robust to overfitting and naturally provide feature importance rankings.

4.3.4 Gradient Boosting

An ensemble method that builds trees sequentially, with each tree correcting the residual errors of the prior ensemble. Gradient Boosting minimizes a loss function (log-loss for binary classification) via gradient descent in function space. The algorithm is highly effective for structured tabular data and typically achieves state-of-the-art performance on clinical classification tasks.

4.3.5 AdaBoost

AdaBoost (Adaptive Boosting) trains weak classifiers (shallow decision stumps) sequentially, re-weighting misclassified examples at each iteration to focus subsequent learners on difficult cases. While related to Gradient Boosting, AdaBoost focuses on re-weighting samples rather than fitting residuals.

4.3.6 Support Vector Machine (SVM)

An SVM with the Radial Basis Function (RBF) kernel was applied. SVM finds the maximum-margin hyperplane in the transformed feature space. The RBF kernel allows non-linear decision boundaries, making it well-suited for complex medical data. probability=True was enabled to generate probability estimates via Platt scaling.

4.3.7 K-Nearest Neighbors (KNN)

KNN classifies new samples by majority vote among the k=5 nearest neighbors in the feature space (using Euclidean distance). KNN is a non-parametric method with no training phase (lazy learner), relying entirely on the training set at inference time. It is computationally expensive for large datasets and sensitive to irrelevant features and scale.

4.3.8 Naive Bayes

A Gaussian Naive Bayes classifier assumes feature independence and models each feature's likelihood as a Gaussian distribution per class. Despite its strong independence assumption — which is violated by correlated clinical variables — Naive Bayes often performs surprisingly well in practice and is extremely fast to train and predict.

Table 4. Configuration of Machine Learning Algorithms

Algorithm	Type	Key Parameters	Complexity
Logistic Regression	Linear	max_iter=1000	O(n * p)
Decision Tree	Tree	max_depth=5	O(n log n)
Random Forest	Ensemble (Bagging)	n_estimators=100	O(k * n log n)

Algorithm	Type	Key Parameters	Complexity
Gradient Boosting	Ensemble (Boosting)	n_estimators=100, lr=0.1	O(k * n log n)
AdaBoost	Ensemble (Boosting)	n_estimators=50	O(k * n)
SVM (RBF)	Kernel Method	C=1.0, gamma='scale'	O(n² to n³)
KNN	Instance-Based	k=5, Euclidean	O(n * p) per query
Naive Bayes	Probabilistic	Gaussian distribution	O(n * p)

Table 4. Machine learning algorithm configurations. n = samples, p = features, k = number of estimators.

4.4 Evaluation Metrics

Model performance was evaluated using five standard binary classification metrics, each capturing a distinct aspect of predictive quality:

Accuracy: The proportion of all predictions (both positive and negative) that are correct. Accuracy = (TP + TN) / (TP + TN + FP + FN).

Precision: Among all predicted CKD cases, the proportion that are true CKD. Precision = TP / (TP + FP). High precision minimizes false alarms.

Recall (Sensitivity): Among all actual CKD cases, the proportion correctly identified. Recall = TP / (TP + FN). High recall minimizes missed diagnoses — critical for CKD screening.

F1-Score: The harmonic mean of precision and recall. F1 = 2 * (Precision * Recall) / (Precision + Recall). Balances the trade-off between false positives and false negatives.

ROC-AUC: Area Under the Receiver Operating Characteristic Curve, measuring the model's ability to discriminate between classes across all classification thresholds. AUC = 1.0 represents perfect discrimination.

5. Experimental Results

5.1 Dataset After Preprocessing

After completing all preprocessing steps, the final dataset contained 12,387 records and 29 features (including the target variable ckd_present). The retained features span clinical, demographic, anthropometric, and laboratory domains. The six binary ethnicity dummy variables increased the feature count from the original continuous and categorical set. All numerical features were standardized to mean ≈ 0 and standard deviation ≈ 1.

The training set contained 9,909 records and the test set contained 2,478 records, maintaining the original ~72%/28% CKD/non-CKD ratio through stratified splitting. No resampling was performed, as the ensemble methods tested have demonstrated robustness to mild-to-moderate class imbalance.

5.2 Model Performance Comparison

Table 5 presents the consolidated performance results for all eight algorithms evaluated on the held-out test set (n = 2,478). All metrics are reported to four decimal places to facilitate precise comparison.

Table 5. Consolidated Performance Metrics for All Eight ML Algorithms on the NHANES CKD Test Set

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Gradient Boosting	0.9883	0.9910	0.9922	0.9916	0.9986
Random Forest	0.9824	0.9886	0.9862	0.9874	0.9982
Decision Tree	0.9837	0.9863	0.9904	0.9883	0.9964
AdaBoost	0.9799	0.9880	0.9832	0.9856	0.9953
Logistic Regression	0.9162	0.9170	0.9676	0.9417	0.9376
SVM (RBF)	0.9506	0.9575	0.9724	0.9649	0.9821
KNN	0.9074	0.9199	0.9502	0.9348	0.9535
Naive Bayes	0.8697	0.9815	0.8291	0.8989	0.9555

Table 5. Performance metrics for all eight classifiers. Gradient Boosting achieves the highest scores across accuracy, precision, recall, F1, and ROC-AUC.

5.3 Performance Analysis by Metric

5.3.1 Accuracy

Gradient Boosting (98.83%) achieved the highest accuracy, followed closely by Decision Tree (98.37%) and Random Forest (98.24%). The ensemble methods consistently outperformed single models. Naive Bayes recorded the lowest accuracy (86.97%), attributable to its feature-independence assumption being violated by the correlated clinical biomarkers in the dataset. Logistic Regression (91.62%) and KNN (90.74%) performed similarly, reflecting their limitations in capturing non-linear relationships in the feature space.

5.3.2 Precision

Naive Bayes achieved the highest precision (98.15%) among all models, indicating that when it predicts CKD, it is nearly always correct. However, this comes at the cost of very poor recall (82.91%), meaning it misses a substantial number of true CKD cases. This precision-recall trade-off is critical in a clinical context: for CKD screening, recall (sensitivity) is generally prioritized to avoid missed diagnoses. Gradient Boosting achieved the best balance, with precision of 99.10% and recall of 99.22%.

5.3.3 Recall

Recall (sensitivity) is arguably the most clinically important metric for CKD screening, as a false negative — failing to identify a patient with CKD — carries substantial health consequences including delayed treatment. Gradient Boosting achieved the highest recall (99.22%), followed by Decision Tree (99.04%) and Logistic Regression (96.76%). The high recall of Logistic Regression despite lower overall accuracy suggests it is biased toward predicting the majority class (CKD-positive), which inflates recall at the expense of specificity.

5.3.4 F1-Score

The F1-score, which balances precision and recall, confirms Gradient Boosting (99.16%) as the overall best performer, followed by Decision Tree (98.83%) and Random Forest (98.74%). The ranking is consistent with accuracy, reinforcing the reliability of Gradient Boosting. Naive Bayes, despite its high precision, ranked last on F1 (89.89%) due to poor recall.

5.3.5 ROC-AUC

The ROC-AUC metric evaluates discriminative ability across all classification thresholds. Gradient Boosting achieved an AUC of 0.9986, essentially perfect discrimination. The ROC curve comparison (Figure 4) demonstrates that ensemble methods (Gradient Boosting, Random Forest, Decision Tree, AdaBoost) cluster at the top-left corner of the ROC space, indicating near-perfect sensitivity at very low false-positive rates. Logistic Regression (AUC = 0.938) showed a notably less ideal ROC profile, suggesting its linear decision boundary is insufficient to capture the complex relationships in the NHANES feature space.

5.4 Confusion Matrix Analysis — Gradient Boosting

The confusion matrix for the best-performing model (Gradient Boosting) on the 2,478-sample test set reveals outstanding classification performance across both classes. Of 719 true CKD-negative patients, 704 were correctly identified (97.91% specificity), with only 15 false positives. Of 1,668 true CKD-positive patients, 1,655 were correctly identified (99.22% sensitivity), with only 13 false negatives. The clinical implications are significant: only 13 CKD patients would be missed per 2,478 screened individuals, a false-negative rate of approximately 0.78%.

Table 6. Confusion Matrix — Gradient Boosting (Best Model)

	Predicted: No CKD	**Predicted: CKD**
Actual: No CKD	TP = 704 (TN)	FP = 15
Actual: CKD	FN = 13	TP = 1,655

Table 6. Confusion matrix values for Gradient Boosting on the held-out test set (n = 2,478).

Table 7. Overall Algorithm Ranking by Composite Performance Score (Average of All 5 Metrics)

Rank	**Algorithm**	**Avg. Score**	**Strengths**	**Weaknesses**
1	Gradient Boosting	0.9923	All-around best performer	Slower training time
2	Decision Tree	0.9890	Fast, interpretable	May overfit without pruning
3	Random Forest	0.9886	Robust, feature importance	Less interpretable than DT
4	AdaBoost	0.9864	Handles class imbalance	Sensitive to noisy labels
5	SVM (RBF)	0.9655	Strong non-linear boundary	Slow at n > 10,000
6	Logistic Regression	0.9360	Interpretable, fast	Linear only; lower accuracy

Rank	Algorithm	Avg. Score	Strengths	Weaknesses
7	KNN	0.9332	No training phase	High compute at inference
8	Naive Bayes	0.9069	Extremely fast training	Low recall; independence assumption

Table 7. Algorithm rankings based on the unweighted mean of accuracy, precision, recall, F1-score, and ROC-AUC.

6. Discussion

6.1 Superiority of Ensemble Tree Methods

The results demonstrate a clear and consistent performance advantage for ensemble tree-based methods — specifically Gradient Boosting, Random Forest, and Decision Tree — over linear and instance-based methods. This pattern is consistent with the broader machine learning literature on structured tabular medical data, where ensemble methods routinely outperform other paradigms.

Gradient Boosting's superiority can be attributed to its iterative error-correction mechanism, which allows it to precisely fit the complex, non-linear interactions between clinical features that predict CKD status. Features such as eGFR, serum creatinine, blood urea nitrogen, and albumin-to-creatinine ratio have well-established but non-linear relationships with CKD risk — relationships that linear models cannot fully capture. The sequential boosting process enables Gradient Boosting to effectively model these interactions without manual feature engineering.

6.2 Clinical Interpretation of Results

From a clinical perspective, the performance metrics observed in this study — particularly the 99.22% recall achieved by Gradient Boosting — suggest strong potential for use as a screening tool. In population-level CKD screening applications, a tool with high sensitivity minimizes the risk of undetected disease, allowing clinicians to focus confirmatory diagnostic resources on the relatively small number of high-risk predicted-positive patients.

The high precision (99.10%) is equally important, as it indicates that CKD predictions are highly reliable, reducing unnecessary follow-up testing and associated patient anxiety and healthcare costs. The combination of high precision and recall represents the ideal characteristics for a clinical decision support system deployed in primary care settings.

However, it is critical to note that these performance levels were achieved in a controlled experimental setting. Real-world deployment would introduce additional challenges, including data quality variability across clinical sites, population distribution shifts, and the need for clinician integration and oversight. The model should be treated as a screening support tool rather than a standalone diagnostic system.

6.3 Limitations of Logistic Regression and KNN

Logistic Regression's relatively lower performance (91.62% accuracy) is expected given the known non-linearity of CKD biomarker relationships. While polynomial or interaction features could improve Logistic Regression performance, this would reduce interpretability — one of its primary advantages over ensemble methods. KNN's

limitation (90.74% accuracy) is partly attributable to the curse of dimensionality: with 29 features, Euclidean distance becomes less discriminative, and the computational cost of distance calculations across 9,909 training samples is substantial.

6.4 Naive Bayes Trade-off

The Naive Bayes classifier presents an interesting case: it achieved the second-highest precision (98.15%) but the lowest recall (82.91%), resulting in the lowest overall accuracy and F1-score. This behavior is characteristic of models that are overly conservative — they only predict CKD when the evidence is overwhelming, leading to many missed cases. In clinical practice, this profile (high precision, low recall) would be unacceptable for a primary screening tool, as it would miss approximately 1 in 6 CKD cases.

6.5 Feature Leakage Considerations

A critical methodological decision in this study was the removal of the ckd_stage variable before model training. CKD staging is derived directly from the eGFR and ACR thresholds that define the binary target variable ckd_present. Including ckd_stage would constitute a severe form of data leakage, artificially inflating model performance to near-perfect levels through a trivially learnable relationship. Similarly, participant_id was removed to prevent any spurious patient-level memorization.

Despite removing ckd_stage, the dataset retains eGFR and albumin_creatinine_ratio — the two primary clinical determinants of CKD classification. This creates a softer form of potential leakage, as these features are definitionally related to the target. This is an inherent challenge in CKD ML studies using retrospectively labeled datasets, and future work should explore models that exclude these features to assess performance based solely on indirect risk factors.

6.6 Comparison with Prior Literature

The Gradient Boosting accuracy of 98.83% and AUC of 0.9986 reported in this study exceeds most prior literature on CKD prediction. This is attributable to several factors: the use of the larger and more recent NHANES 2021–2023 dataset; the comprehensive preprocessing pipeline that effectively handles missingness and outliers; and the retention of direct renal biomarkers (eGFR, ACR, serum creatinine) that provide high discriminative power. Studies using only indirect risk factors (age, diabetes, hypertension) without laboratory data consistently report lower performance metrics.

7. Limitations and Future Work

7.1 Limitations

Cross-Sectional Design: NHANES data is cross-sectional; the models do not account for longitudinal disease progression. Predictive models for CKD progression (stage advancement) would require longitudinal cohort data.

Default Hyperparameters: All models were trained using default scikit-learn hyperparameters to ensure fairness in baseline comparison. Hyperparameter optimization (e.g., via GridSearchCV or Bayesian optimization) would likely yield further performance improvements, particularly for SVM and KNN.

No External Validation: Results are validated on an internal held-out test set drawn from the same NHANES survey. External validation on an independent clinical cohort (e.g., UKBB or MDRD datasets) is necessary to assess generalizability.

Class Imbalance Handling: While stratified splitting preserved the original class ratio, no SMOTE or other oversampling technique was applied. The effect of resampling on model calibration and real-world performance warrants further investigation.

Interpretability Gap: Gradient Boosting is a black-box model. Clinical adoption requires explainability via tools such as SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations), which were outside the scope of this study.

7.2 Future Work

- Apply SHAP analysis to identify the most influential features for CKD prediction in the Gradient Boosting model and validate clinical plausibility.
- Explore deep learning approaches, including tabular-specific architectures such as TabNet and FT-Transformer, for comparison with classical ML.
- Conduct hyperparameter tuning via Bayesian optimization for all eight algorithms under equal computational budgets.
- Apply the preprocessing and modeling pipeline to other NHANES cycles (2015–2020) to assess temporal generalizability.
- Develop a clinically deployable web application integrating the Gradient Boosting model with real-time CKD risk scoring for primary care use.
- Investigate multi-class CKD stage prediction (Stages 1–5) as a more granular extension of the binary classification framework.

8. Conclusion

This study presented a comprehensive empirical analysis of eight supervised machine learning algorithms for binary Chronic Kidney Disease prediction using the NHANES 2021–2023 clinical dataset. A structured six-step preprocessing pipeline was developed and validated, addressing missing data, outliers, feature encoding, leakage removal, and standardization. All models were trained under identical experimental conditions and evaluated using five standard performance metrics.

The results conclusively demonstrate that Gradient Boosting achieves the highest performance across all metrics — 98.83% accuracy, 99.10% precision, 99.22% recall, 99.16% F1-score, and 0.9986 ROC-AUC — establishing it as the recommended algorithm for CKD classification tasks using clinical population survey data. Ensemble tree methods as a group (Gradient Boosting, Random Forest, Decision Tree, AdaBoost) dominated over linear and instance-based alternatives, confirming the established superiority of ensemble approaches on structured medical tabular data.

The clinical implications are significant: a Gradient Boosting model trained on accessible clinical and laboratory data can screen patient populations for CKD with near-perfect sensitivity and precision, enabling early intervention and reducing the burden of late-stage CKD. Deployment as a clinical decision support tool — integrated with electronic health record systems — represents a promising and actionable direction for future translational work.

These findings contribute to the growing body of evidence supporting machine learning as a transformative technology in chronic disease detection and surveillance. Future research directions include external validation, explainability analysis, hyperparameter optimization, and longitudinal CKD progression modeling. The code and preprocessing pipeline are publicly reproducible, with the dataset available from Kaggle as cited.

References

[1] Polat, H., Danaei Mehr, H., & Cetin, A. (2017). Diagnosis of chronic kidney disease based on support vector machine by feature selection methods. Journal of Medical Systems, 41(4), 55.

[2] Salekin, A., & Stankovic, J. (2016). Detection of chronic kidney disease and selecting important predictive attributes. IEEE International Conference on Healthcare Informatics (ICHI), 262–270.

[3] Islam, M. A., Akter, S., Hossen, S., Keya, S. A., Tisha, S. A., & Hossain, S. (2020). Risk factor prediction of chronic kidney disease based on machine learning algorithms. IEEE Region 10 Symposium (TENSYMP), 1–4.

[4] Norouzi, J., Yadollahpour, A., Mirbagheri, S. A., Mazdeh, M. M., & Hosseini, S. A. (2021). Predicting renal failure progression in chronic kidney disease using integrated intelligent fuzzy expert system. Computational and Mathematical Methods in Medicine, 2016, 6080814.

[5] Ene-Iordache, B., Perico, N., Bikbov, B., Carminati, S., Remuzzi, A., Perna, A., ... & Remuzzi, G. (2016). Chronic kidney disease and cardiovascular risk in six regions of the world (ISN-KDDC). Lancet Global Health, 4(5), e307–e319.

[6] Levin, A., Stevens, P. E., Bilous, R. W., Coresh, J., De Francisco, A. L. M., De Jong, P. E., ... & Winearls, C. G. (2013). Kidney Disease: Improving Global Outcomes (KDIGO) CKD Work Group: KDIGO 2012 clinical practice guideline for the evaluation and management of chronic kidney disease. Kidney International Supplements, 3(1), 1–150.

[7] Centers for Disease Control and Prevention. (2023). National Health and Nutrition Examination Survey: NHANES 2021–2023. U.S. Department of Health and Human Services. https://www.cdc.gov/nchs/nhanes/

[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

[9] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.

[10] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

[11] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273–297.

[12] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 30.

[13] Bikbov, B., Purcell, C. A., Levey, A. S., Smith, M., Abdoli, A., Abebe, M., ... & Perico, N. (2020). Global, regional, and national burden of chronic kidney disease, 1990–2017. Lancet, 395(10225), 709–733.

[14] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189–1232.

[15] Shah, A. T. (2023). CKD NHANES 2021–2023 Staged Kidney Disease Dataset. Kaggle. https://www.kaggle.com/datasets/alitaqishah/ckd-nhanes-2021-2023-staged-kidney-disease