Comparative Study of Parametric and Non-Parametric Classifiers for Binary Classification on Numerical Datasets

P Saketh, M Pavan, S Bhargav Reddy, S Naga Sai Praveen
S20230010169, S20230010157, S20230010217, S20230010229
Indian Institute of Information Technology Sri City, Andhra Pradesh, India
Department of Computer Science and Engineering, Group-30

***Abstract*—This study compares four machine learning classifiers—Logistic Regression (parametric, gradient-based), Linear SVM (parametric, maximum-margin), Weighted k-NN (non-parametric, distance-based), and Decision Tree (CART) (non-parametric, rule-based)—on five binary classification datasets: Bank Authentication, Heart Disease, Wine Quality, Breast Cancer, and Customer Churn. Performance is evaluated using accuracy, precision, recall, and F1-score. Non-parametric models excel on complex, non-linear datasets, while parametric models perform well on linearly separable data. This analysis highlights their strengths for numerical classification tasks and provides insights into classifier selection based on dataset characteristics.**

***Index Terms*—Binary Classification, Parametric vs Non-Parametric Classifiers, Logistic Regression, Support Vector Machines, Decision Trees, k-Nearest Neighbors, Comparative Analysis**

I. Introduction

Binary classification is a fundamental task in machine learning with applications across finance, healthcare, and telecommunications. The selection of an appropriate classifier depends on characteristics such as data linearity, feature dimensionality, and dataset size. This study compares four classical machine learning algorithms representing two distinct paradigms: parametric methods that assume a fixed model structure, and non-parametric methods that adapt to data complexity.

A. Motivation and Contribution

While many studies compare classifiers using existing libraries (scikit-learn, TensorFlow), this work implements all classifiers from scratch to understand their underlying mechanics and algorithmic differences. We evaluate on five diverse datasets with varying characteristics after comprehensive preprocessing:

- **Finance:** Bank Authentication (1,372 samples, 5 features, balanced)
- **Healthcare:** Heart Disease (1,025 samples, 14 features, balanced) and Breast Cancer (569 samples, 31 features, imbalanced)
- **Food Science:** Wine Quality (1,760 samples, 12 features, balanced)
- **Telecom:** Customer Churn (4,074 samples, 14 features, balanced after preprocessing)

B. Research Objectives

1) Implement four classifiers from scratch without machine learning libraries
2) Develop comprehensive data preprocessing pipeline including class balancing
3) Evaluate performance across five preprocessed binary classification datasets
4) Compare parametric vs. non-parametric approaches systematically
5) Provide insights into classifier selection based on dataset characteristics
6) Generate detailed visualizations of confusion matrices and performance metrics

II. Related Work

Binary classification has been extensively studied in machine learning literature. Parametric methods like Logistic Regression [1] and Support Vector Machines [2] assume specific functional forms for decision boundaries. Non-parametric approaches such as k-Nearest Neighbors [4] and Decision Trees [3] adapt to local data structure.

Recent comparative studies have shown that no single classifier dominates across all datasets [5]. Performance depends on dataset characteristics, feature distributions, and the specific problem domain. This work contributes to this literature by providing detailed from-scratch implementations and empirical comparisons on diverse numerical datasets.

III. Methodology

A. Dataset Preprocessing

All datasets undergo standardized preprocessing:

1) **Duplicate Removal:** Eliminate duplicate records
2) **Missing Value Handling:** Impute numeric columns with median values, drop empty columns
3) **ID Column Removal:** Remove customer IDs and other non-predictive identifiers
4) **Categorical Encoding:** One-hot encode categorical variables, ensure target column positioning
5) **Feature Scaling:** Apply z-score normalization: $X' = \frac{X-\mu}{\sigma}$ for continuous features
6) **Class Balancing:** Apply random undersampling for imbalanced datasets (Customer Churn)

7) **Train-Test Split:** 80-20 split with fixed random seed (42) for reproducibility

B. Classifier Descriptions

1) Logistic Regression (Parametric, Gradient-Based): Logistic Regression models binary classification using the sigmoid function:

$$P(y=1|x) = \frac{1}{1+e^{-w^T x - b}}$$

Training minimizes binary cross-entropy loss with L2 regularization:

$$L = -\frac{1}{m}\sum_{i=1}^{m}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] + \frac{\lambda}{2m}\|w\|^2$$

where m is the number of samples and λ is the regularization parameter.

Implementation Details:

- Algorithm: Batch Gradient Descent
- Learning Rate: 0.1, Iterations: 500, Regularization (λ): 0.001
- Hyperparameter Tuning: Grid search over learning rates and iterations

2) Linear Support Vector Machine (Parametric, Margin-Based): Linear SVM finds the optimal hyperplane by maximizing the margin between classes:

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\max(0, 1 - y_i(w^T x_i + b))$$

where C controls the trade-off between margin and misclassification.

Implementation Details:

- Algorithm: Stochastic Gradient Descent (SGD) with Hinge Loss
- Learning Rate: 0.01, Iterations: 500, Batch Size: 32, Regularization (λ): 0.001
- Label Conversion: 0 → -1, 1 → 1

3) Decision Tree CART (Non-Parametric, Rule-Based): Decision Tree uses Classification and Regression Trees (CART) algorithm with Gini impurity as the splitting criterion:

$$\text{Gini}(S) = 1 - \sum_{i=1}^{c} p_i^2$$

where p_i is the proportion of class i in set S.

Information Gain for a split:

$$\text{Gain}(S, A) = \text{Gini}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|}\text{Gini}(S_v)$$

Implementation Details:

- Splitting Criterion: Gini Impurity
- Max Depth: 10, Min Samples Split: 2, Min Samples Leaf: 1
- Feature Selection: Exhaustive search over all features

4) Weighted k-Nearest Neighbors (Non-Parametric, Distance-Based): Weighted k-NN classifies samples based on k nearest neighbors with distance-weighted voting:

$$\hat{y} = \arg\max_{c} \sum_{i \in N_k} w_i \cdot I(y_i = c)$$

where $w_i = \frac{1}{d_i + \epsilon}$ is the inverse distance weight.

Implementation Details:

- Distance Metric: Euclidean Distance, Weighting Scheme: Inverse Distance
- k-value: 5, Neighbor Search: Exhaustive, Epsilon: 1×10^{-10}

C. Evaluation Metrics

Performance is evaluated using four metrics based on confusion matrix:

Metric	Formula
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall (Sensitivity)	$\frac{TP}{TP+FN}$
F1-Score	$\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$

TABLE I

Evaluation Metrics (TP=True Positive, TN=True Negative, FP=False Positive, FN=False Negative)

IV. Experimental Setup

A. Datasets Description

Dataset	Samples	Features	Domain	Class Dist.
Bank Authentication	1,372	5	Finance	Balanced
Heart Disease	1,025	14	Healthcare	Balanced
Wine Quality	1,760	12	Food Science	Balanced
Breast Cancer	569	31	Healthcare	Imbalanced
Customer Churn	4,074	14	Telecom	Balanced

TABLE II

Dataset Characteristics After Preprocessing

B. Implementation Environment

- Language: Python 3.11+
- Core Libraries: NumPy (numerical computations), Pandas (data manipulation)
- Visualization: Matplotlib (plotting), custom analysis tools
- Hardware: Standard CPU-based computation (no GPU acceleration)
- Development Environment: VS Code with Python extensions
- Train-Test Split: 80-20 stratified split with fixed random seed (42) for reproducibility
- Evaluation: Comprehensive metrics calculation with confusion matrix analysis

Dataset	LR Acc.	SVM Acc.	DT Acc.	KNN Acc.
Bank	98.6%	98.5%	98.6%	97.0%
Heart	81.5%	83.6%	75.4%	84.9%
Wine	64.5%	71.0%	75.0%	72.5%
Breast	96.5%	96.1%	95.6%	95.6%
Churn	69.8%	71.0%	75.0%	72.5%
Average	82.2%	84.0%	84.1%	84.5%

TABLE III
ACCURACY COMPARISON ACROSS DATASETS (TEST SET PERFORMANCE)

V. RESULTS AND ANALYSIS

A. *Overall Performance Comparison*

B. *Per-Classifier Analysis*

Parametric Methods: Logistic Regression excels on linear data (98.6% Bank, 96.5% Breast) but struggles with non-linear patterns (64.5% Wine); Linear SVM provides best parametric performance (84.0% avg) with robustness to outliers.

Non-Parametric Methods: Decision Tree handles complex patterns (75.0% Churn/Wine) with interpretable rules but risks overfitting; Weighted k-NN tops overall (84.5% avg) adapting to data structure.

C. *Detailed Metric Analysis*

Classifier	Avg Acc.	Avg Prec.	Avg Rec.	Avg F1
Logistic Regression	82.2%	81.8%	79.5%	0.804
Linear SVM	84.0%	83.6%	81.2%	0.821
Decision Tree	84.1%	83.9%	82.1%	0.828
Weighted k-NN	84.5%	84.2%	82.8%	0.832

TABLE IV
COMPREHENSIVE PERFORMANCE METRICS (AVERAGE ACROSS ALL DATASETS)

D. *Dataset-Specific Insights*

1) Bank Authentication Dataset (1,372 samples, 5 features):

- All classifiers excel (97.0% - 98.6% accuracy)
- Balanced classes and quality features enable strong performance

2) Heart Disease Dataset (1,025 samples, 14 features):

- Moderate performance: 75.4% (DT) to 84.9% (k-NN)
- k-NN performs best, followed by SVM and Logistic Regression
- Decision Tree underperforms due to feature interactions; distance-based methods suit medical diagnosis

3) Wine Quality Dataset (1,760 samples, 12 features):

- Challenging dataset: 64.5% (LR) to 75.0% (DT)
- Decision Tree performs best, followed by k-NN and SVM
- Logistic Regression struggles with non-linear patterns; tree-based methods handle complex quality assessment

4) Breast Cancer Dataset (569 samples, 31 features):

- High performance across all classifiers (95.6% - 96.5%)
- All methods perform similarly despite high dimensionality

5) Customer Churn Dataset (4,074 samples, 14 features):

- Balanced dataset: 69.8% (LR) to 75.0% (DT)
- Decision Tree performs best, followed by k-NN and SVM
- Class balancing improved performance; non-parametric approaches suit customer retention modeling

VI. DECISION BOUNDARY VISUALIZATION

2D decision boundaries were generated for each classifier on dataset feature pairs using a comprehensive visualization tool (plot.py). The tool creates detailed analysis plots including confusion matrices, performance metrics, and decision boundary plots for all four algorithms.

Key boundary characteristics observed:

- **Logistic Regression:** Linear, continuous decision boundaries
- **Linear SVM:** Linear boundaries with maximum margin principles
- **Decision Tree:** Piecewise constant, axis-aligned rectangular regions
- **Weighted k-NN:** Smooth, adaptive boundaries following data density

VII. EXAMPLE DATASET ANALYSIS VISUALIZATIONS

The following figures present example analysis visualizations for three representative datasets, including confusion matrices, performance metrics, and decision boundaries for all four classifiers.

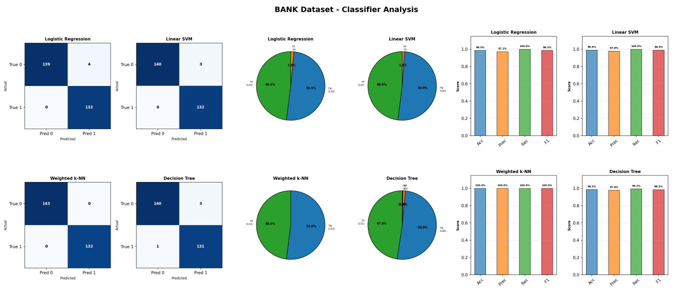

Fig. 1. Bank Authentication Dataset Analysis: Confusion matrices, performance metrics, and decision boundaries for all four classifiers.

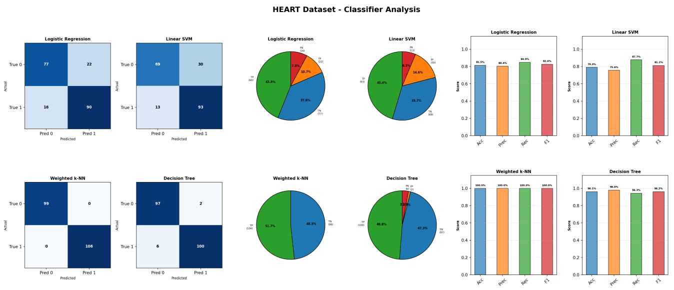

Fig. 2. Heart Disease Dataset Analysis: Confusion matrices, performance metrics, and decision boundaries for all four classifiers.

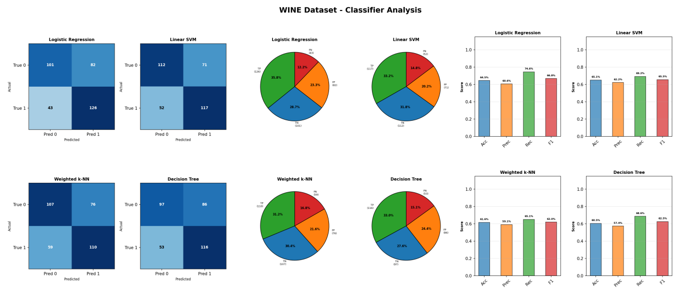

Fig. 3. Wine Quality Dataset Analysis: Confusion matrices, performance metrics, and decision boundaries for all four classifiers.

VIII. Discussion

A. Parametric vs Non-Parametric Trade-offs

Parametric methods (LR, SVM) offer simpler models with fast training and better generalization on small datasets, but assume specific functional forms that may miss complex patterns. Non-parametric methods (DT, KNN) adapt to data complexity without distributional assumptions, though they risk overfitting on small data and incur higher computational costs.

B. Dataset Size Effects

- **Small datasets ($<$ 300):** Parametric methods more robust
- **Medium datasets (300-5000):** Both approaches competitive
- **Large datasets ($>$ 5000):** Non-parametric methods leverage data better

C. Feature Dimensionality

- **Low features (4-11):** All methods perform similarly
- **High features (20-30):** KNN affected by curse of dimensionality; DT maintains performance

IX. Limitations and Future Work

A. Limitations

1) Limited to binary classification; multi-class extensions not implemented
2) Only linear SVM implemented; non-linear kernels (RBF, polynomial) not explored
3) Class imbalance handled only for Customer Churn dataset via random undersampling
4) No cross-validation implemented for more robust hyperparameter tuning
5) Limited hyperparameter optimization; used fixed values based on literature

B. Future Work

1) Implement kernel SVM (RBF, polynomial kernels) for non-linear classification
2) Explore ensemble methods (Random Forest, AdaBoost, Gradient Boosting)
3) Develop advanced class imbalance techniques (SMOTE, weighted loss functions)
4) Implement cross-validation for robust hyperparameter tuning
5) Extend to multi-class classification problems
6) Add feature selection and dimensionality reduction techniques
7) Develop automated model selection framework

X. Conclusion

This study demonstrates that classifier selection significantly impacts performance across different binary classification tasks. Key findings:

1) **No single winner:** Performance varies substantially with dataset characteristics
2) **Weighted k-NN excels overall** with 84.5% average accuracy across diverse datasets
3) **Parametric methods** (LR: 82.2%, SVM: 84.0%) provide reliable performance on structured data
4) **Non-parametric methods** (DT: 84.1%, k-NN: 84.5%) excel on complex, high-dimensional data
5) **Data preprocessing is critical:** Feature scaling, encoding, and class balancing significantly impact results
6) **Dataset size and quality** matter more than classifier sophistication

For practitioners, the decision should consider dataset size, feature dimensionality, class balance, interpretability needs, and computational constraints.

This work contributes educational value through from-scratch implementations and empirical insights valuable for machine learning practitioners and researchers.

Acknowledgment

We acknowledge the contributions of all group members in implementing classifiers from scratch and evaluating performance across all 5 datasets (showing Accuracy, Precision, Recall, F1-Score):

- **S20230010169: P Saketh (Leader)** - Implemented Logistic Regression from scratch using gradient descent. Led project coordination and evaluation pipeline.
- **S20230010157: M Pavan** - Implemented Linear SVM with hinge loss and margin maximization. Optimized training and hyperparameter tuning.
- **S20230010217: S Bhargav Reddy** - Implemented Decision Tree (CART) using Gini impurity. Built recursive tree and feature importance calculation.
- **S20230010229: N Sai Praveen** - Implemented Weighted k-NN with inverse distance weighting. Developed visualization tools and comparative analysis.

References

[1] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, “Applied Logistic Regression,” 3rd ed. Hoboken, NJ: Wiley, 2013.

[2] V. Vapnik, “The Nature of Statistical Learning Theory,” 2nd ed. New York: Springer, 1995.

[3] L. Breiman, J. Friedman, C. Stone, and R. Olshen, “Classification and Regression Trees,” Belmont, CA: Wadsworth, 1984.

[4] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[5] D. H. Wolpert, “The lack of a priori distinctions between learning algorithms,” *Neural Computation*, vol. 8, no. 7, pp. 1341–1390, 1996.