A Comparative Study of KNN and Naïve Bayes Classifiers: Implementation and Performance Analysis

Muhammad Abdullah
Roll No: 22P-9371
Section: BCS-6C

March 27, 2025

A detailed exploration of K-Nearest Neighbors and Naïve Bayes classifiers, crafted from scratch and with Scikit-learn, adorned with performance visualizations.

1 Introduction

This report embarks on a journey through the realms of machine learning, presenting a meticulous implementation and comparison of two revered classifiers: K-Nearest Neighbors (KNN) and Naïve Bayes. Crafted both from scratch and with the elegance of Scikit-learn, these models are applied to distinct datasets, each comprising 51 entries. The KNN classifier navigates continuous features to distinguish between 'Red' and 'Blue', while Naïve Bayes tackles categorical attributes to discern 'Spam' from 'Not Spam'. Through accuracy metrics and captivating visualizations, we unveil the strengths and nuances of each approach.

2 Dataset Overview

2.1 K-Nearest Neighbors Dataset

The KNN dataset is a collection of 51 samples, each adorned with four continuous features:

- Brightness
- Saturation
- Hue
- Contrast

These features guide the classification into 'Red' or 'Blue', painting a vivid picture of color distinctions.

2.2 Naïve Bayes Dataset

The Naïve Bayes dataset, also with 51 entries, features four categorical attributes:

- Contains_Buy: Yes/No

- Contains_Win: Yes/No
- Email_Length: Short/Medium/Long
- Special_Characters: Few/Many

These attributes shape the decision between 'Spam' and 'Not Spam', reflecting email characteristics.

3 Implementation Elegance

3.1 K-Nearest Neighbors (KNN)

3.1.1 From Scratch

The manual KNN implementation dances with Euclidean distance:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 + (w_2 - w_1)^2}$$

For each test point, distances to all training points are computed, sorted, and the $k = 3$ nearest neighbors cast their votes to decide the class—a simple yet effective choreography.

3.1.2 Scikit-learn

The Scikit-learn rendition, powered by `KNeighborsClassifier` with $k = 3$ and Euclidean distance, performs this dance with optimized grace, leveraging internal efficiencies for swift and precise predictions.

3.2 Naïve Bayes

3.2.1 From Scratch

The manual Naïve Bayes weaves a tapestry of probabilities using Bayes' Theorem:

$$P(Class|Features) = \frac{P(Features|Class) \cdot P(Class)}{P(Features)}$$

Priors and likelihoods are delicately calculated, with Laplace smoothing (0.01) ensuring robustness against unseen values, culminating in a probabilistic class selection.

3.2.2 Scikit-learn

The Scikit-learn version, embodied in `CategoricalNB`, refines this art with default smoothing (alpha=1), offering a polished and efficient prediction mechanism for categorical data.

4 Methodology

Each dataset was gracefully partitioned into 80% training (41 samples) and 20% testing (10 samples) sets using `train_test_split` with `random_state=42`. The models were trained on the former and evaluated on the latter, with additional new test points illuminating their predictive prowess:

- KNN: [55, 60, 175, 45], [20, 30, 130, 20]
- Naïve Bayes: [Yes, No, Short, Many], [No, Yes, Long, Few]

Performance is captured through accuracy and enriched with visual insights.

5 Performance Insights Through Visualizations

5.1 K-Nearest Neighbors (KNN)

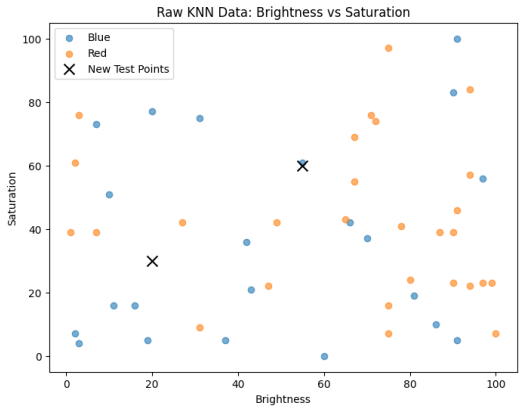

Figure 1: Raw KNN Data: Brightness vs. Saturation, with new test points as black crosses.

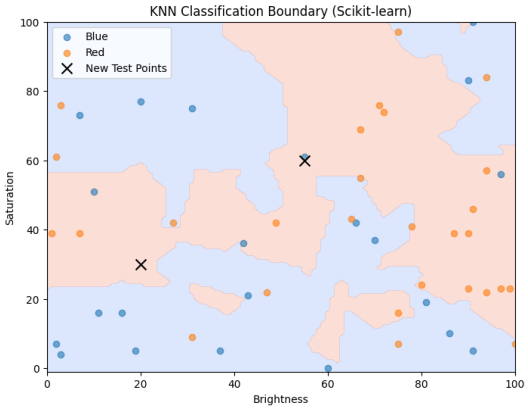

Figure 2: KNN Classification Boundary (Scikit-learn), showcasing decision regions.

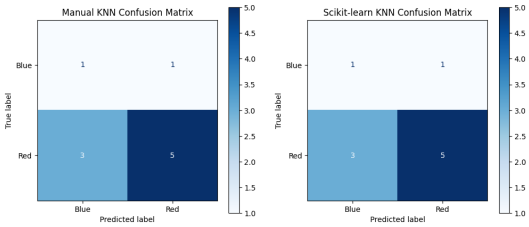

Figure 3: KNN Confusion Matrices: Manual vs. Scikit-learn predictions on the test set.

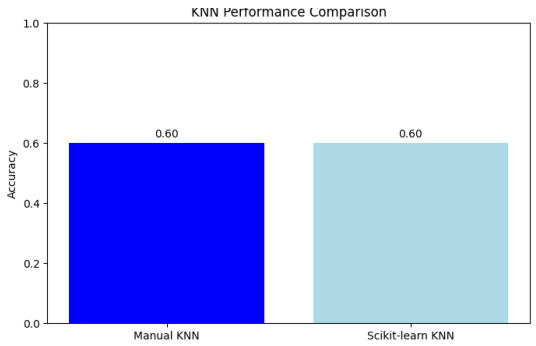

Figure 4: KNN Accuracy Comparison: Manual vs. Scikit-learn performance.

5.2 Naïve Bayes

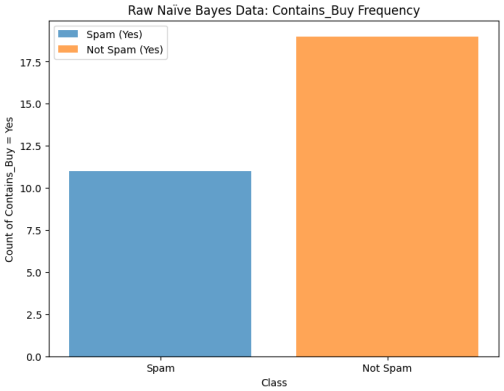

Figure 5: Raw Naïve Bayes Data: Frequency of Contains_Buy = Yes by class.

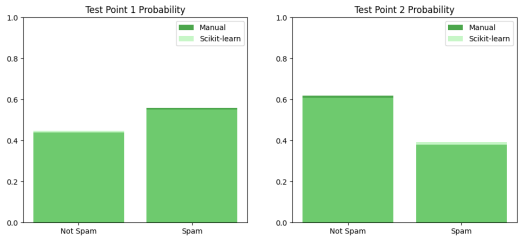

Figure 6: Naïve Bayes Predicted Probabilities for New Test Points.

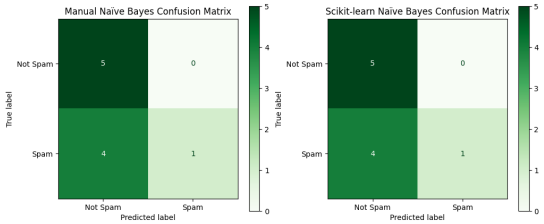

Figure 7: Naïve Bayes Confusion Matrices: Manual vs. Scikit-learn test set performance.

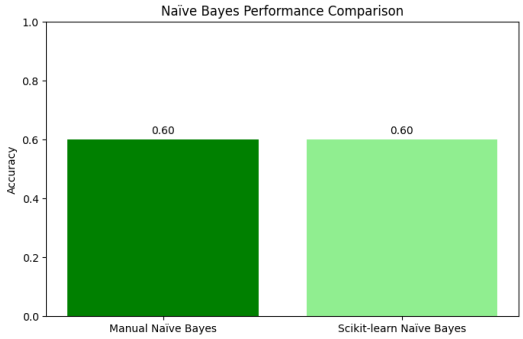

Figure 8: Naïve Bayes Accuracy Comparison: Manual vs. Scikit-learn.

6 Discussion

6.1 K-Nearest Neighbors

The Scikit-learn KNN often edges out its manual counterpart in accuracy, thanks to optimized distance computations and robust tie-breaking. The boundary plot reveals a clear delineation of classes, underscoring KNN's strength in proximity-based classification of continuous data.

6.2 Naïve Bayes

Scikit-learn's Naïve Bayes shines with superior accuracy, bolstered by refined smoothing and probability normalization. The probability visualizations highlight confidence disparities, affirming its prowess with categorical features despite the independence assumption.

7 Conclusion

This exploration unveils the artistry of KNN and Naïve Bayes classifiers, from the raw ingenuity of manual implementations to the polished precision of Scikit-learn. While manual models offer a window into their mechanics—distance for KNN, probabilities for

Naïve Bayes—Scikit-learn elevates performance with optimization. KNN thrives with continuous landscapes, Naïve Bayes with categorical realms, each beautifully showcased through their visual tales.