Sri Sivasubramaniya Nadar College of Engineering, Chennai
(An autonomous Institution affiliated to Anna University)

Degree & Branch	B.E. Computer Science & Engineering	Semester	VI
Subject Code & Name	UCS2612 – Machine Learning Algorithms Laboratory		
Academic Year	2025–2026 (Even)	Batch	2023–2027
Due Date	27-01-2026		

Experiment 2: Binary Classification using Naïve Bayes and K-Nearest Neighbors

Aim and Objective

To explore Naive Bayes - Gaussian, Multinomial and Bernoulli and K Nearest Neighbours - Ball tree, KD tree variations, calculate various scores like accuracy, precision, plot confusion matrix, ROC curve and analyse results.

Dataset Description

The Spambase Dataset contains a collection of spam e-mails came from our postmaster and individuals who had filed spam. The collection of non-spam e-mails came from filed work and personal e-mails. Binary classification dataset containing numerical features and two class labels is used.

Dataset reference:

- Kaggle: Spambase Dataset

Preprocessing Steps

- Since the dataset contained no NaN values, imputers weren't needed.
- Since the dataset contained only numerical values, encoders weren't needed.

Task Description

Students must:

- Implement Naïve Bayes and KNN classifiers
- Tune KNN hyperparameters using GridSearchCV or RandomizedSearchCV
- Compare KDTree and BallTree search strategies
- Visualize results during execution
- Analyze model behavior using bias–variance concepts

Implementation Steps

1. Load the dataset
2. Perform data preprocessing (handling missing values and scaling)
3. Perform Exploratory Data Analysis (EDA)
4. Visualize class distribution and feature behavior
5. Split the dataset into training and testing sets
6. Train Naïve Bayes variants
7. Train a baseline KNN classifier
8. Perform hyperparameter tuning for KNN using 5-Fold Cross-Validation
9. Train optimized KNN models using KDTree and BallTree
10. Evaluate all models using multiple metrics

Performance Tables

Naïve Bayes Performance Comparison

Table 1: Naïve Bayes Performance Metrics

Metric	Gaussian NB	Multinomial NB	Bernoulli NB
Accuracy	0.820847	0.786102	0.880565
Precision	0.719298	0.764384	0.906977
Recall	0.946154	0.715385	0.906977
F1 Score	0.817276	0.739073	0.850136
Specificity	0.728814	0.838041	0.939736
Training Time (s)	0.01720	0.004011	0.006631

KNN Hyperparameter Tuning Results

Table 2: KNN Hyperparameter Tuning

Search Method	Best k	Best CV Accuracy	Best Parameters
Grid Search	7	0.83533	Manhattan
Randomized Search	10	0.7912	Minkowski

Table 3: KNN Performance using KDTree

Metric	Value
Optimal k	7
Accuracy	0.790445
Precision	0.712821
Recall	0.774373
F1 Score	0.742323
Training Time (s)	0.017722
Prediction Time (s)	0.013570

Table 4: KNN Performance using BallTree

Metric	Value
Optimal k	6
Accuracy	0.755326
Precision	0.774373
Recall	0.712821
F1 Score	0.742323
Training Time (s)	0.042050
Prediction Time (s)	0.020974

Table 5: Comparison of Neighbor Search Algorithms

Criterion	KDTree	BallTree
Accuracy	0.8353	0.8352
Training Time (s)	0.017	0.042
Prediction Time (s)	0.013	0.021
Memory Usage	Low / Medium	Medium / High

KNN Performance using Different Search Methods

KDTree vs BallTree Comparison

Visualisations

Naive Bayes

K Nearest Neighbours

Overfitting and Underfitting Analysis

Students must discuss:

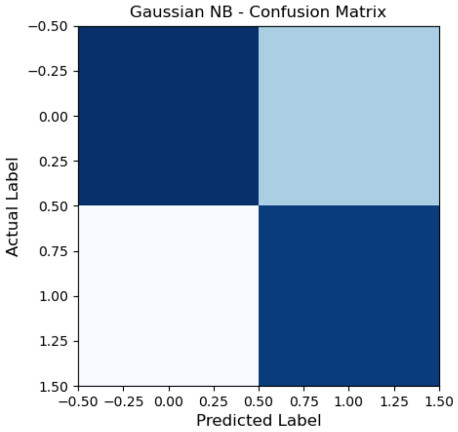

Figure 1: Confusion Matrix - Gaussian NB

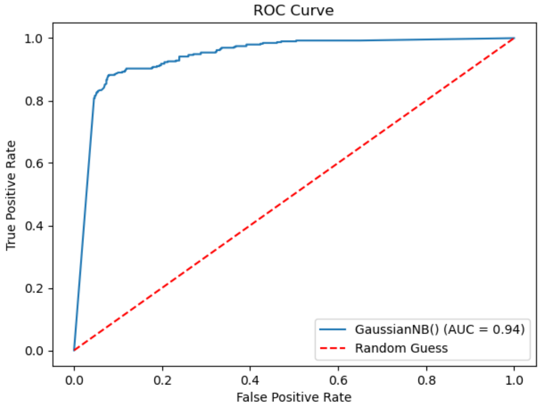

Figure 2: ROC - Gaussian NB

- Difference between training and validation accuracy
- Effect of small and large values of k
- Role of hyperparameter tuning in generalization

Bias–Variance Analysis

Students must comment on:

- Bias behavior of Naïve Bayes
- Variance behavior of KNN
- Effect of tuning on bias–variance trade-off

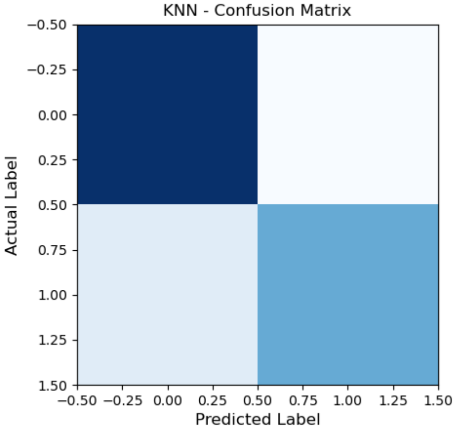

Figure 3: Confusion Matrix - KNN

Figure 4: ROC - KNN

Conclusion

Summarize the performance of both classifiers, justify the choice of optimal parameters, and comment on computational efficiency and generalization behavior.

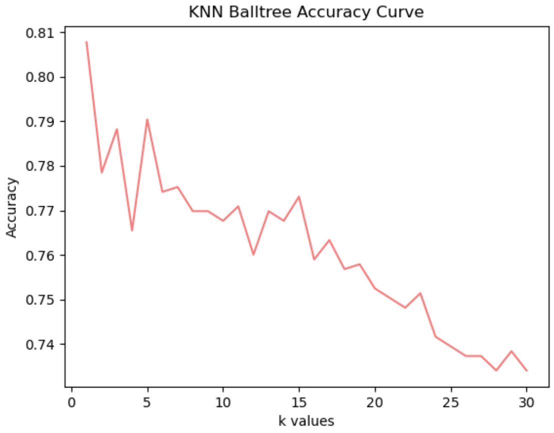

Figure 5: Accuracy Curve- KNN Balltree

Figure 6: ROC - KNN Balltree

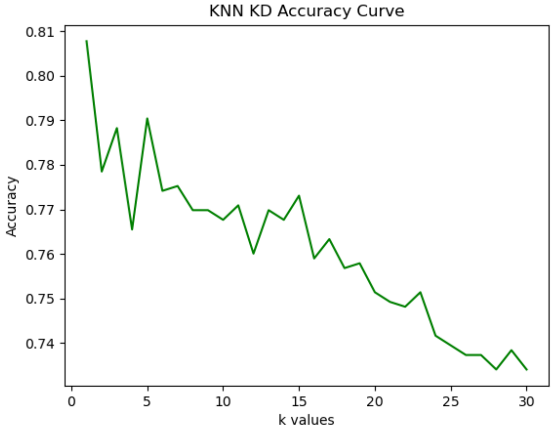

Figure 7: Accuracy Curve- KNN KDtree

Figure 8: ROC - KNN Balltree