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		

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

Objective

To implement Naïve Bayes and K-Nearest Neighbors (KNN) classifiers for a binary classification problem, evaluate them using multiple performance metrics, visualize model behavior, and analyze overfitting, underfitting, and bias–variance characteristics.

Dataset

A benchmark binary classification dataset containing numerical features and two class labels is used.

Dataset reference:

- Kaggle: Spambase Dataset

Brief Theory (For Lab Understanding)

Naïve Bayes

Naïve Bayes is a probabilistic classifier that works well for high-dimensional data. It is fast, simple to implement, and assumes independence among features. Different variants handle different types of input data.

K-Nearest Neighbors (KNN)

KNN is an instance-based learning algorithm that classifies samples based on similarity. The choice of the number of neighbors (k) strongly influences performance. Feature scaling is important for distance-based methods like KNN.

Neighbor Search Methods

KDTree and BallTree are used to speed up nearest neighbor searches. They mainly affect computation time and memory usage, not classification accuracy.

Hyperparameter Tuning

Hyperparameter tuning helps identify the best model settings using validation data. Grid Search and Randomized Search are commonly used approaches.

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

Required Visualizations (During Coding)

Students must generate and include:

- Class distribution plot
- Feature distribution plots
- Confusion matrix for each classifier
- ROC curve for each classifier
- Accuracy vs. k plot for KNN
- Training vs. validation accuracy plot

Performance Metrics to be Reported

- Accuracy
- Precision
- Recall
- F1 Score
- Specificity
- False Positive Rate
- Training Time
- Prediction Time

Naïve Bayes Performance Comparison

Table 1: Naïve Bayes Performance Metrics

Metric	Gaussian NB	Multinomial NB	Bernoulli NB
Accuracy			
Precision			
Recall			
F1 Score			
Specificity			
Training Time (s)			

KNN Hyperparameter Tuning Results

Table 2: KNN Hyperparameter Tuning

Search Method	Best k	Best CV Accuracy	Best Parameters
Grid Search			
Randomized Search			

KNN Performance using Different Search Methods

KDTree vs BallTree Comparison

Overfitting and Underfitting Analysis

Students must discuss:

Table 3: KNN Performance using KDTree

Metric	Value
Optimal k	
Accuracy	
Precision	
Recall	
F1 Score	
Training Time (s)	
Prediction Time (s)	

Table 4: KNN Performance using BallTree

Metric	Value
Optimal k	
Accuracy	
Precision	
Recall	
F1 Score	
Training Time (s)	
Prediction Time (s)	

Table 5: Comparison of Neighbor Search Algorithms

Criterion	KDTree	BallTree
Accuracy		
Training Time (s)		
Prediction Time (s)		
Memory Usage	Low / Medium	Medium / High

- 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

Conclusion

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

Report Format (Mandatory)

1. Aim and Objective
2. Dataset Description
3. Preprocessing Steps
4. Implementation Details
5. Visualizations
6. Performance Tables
7. Overfitting and Underfitting Analysis
8. Bias–Variance Analysis
9. Observations and Conclusion

References

- Scikit-learn: Naïve Bayes
- Scikit-learn: KNN
- Scikit-learn: Hyperparameter Optimization
- Spambase Dataset