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
Name	Monesh M	Register No.	3122235001084
Due Date	06.01.2026		

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

1. Aim and 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.

2. Dataset Description

The Spambase dataset is a benchmark binary classification dataset used to identify spam emails. It consists of numerical features extracted from email content and a binary class label indicating spam or non-spam.

Dataset Reference:

- Kaggle – Spambase Dataset

3. Preprocessing Steps

- Loaded the dataset using Pandas
- Separated features and target labels
- Performed stratified train–test split
- Applied feature scaling using StandardScaler

4. Implementation Details

- Implemented Gaussian, Multinomial, and Bernoulli Naïve Bayes classifiers
- Implemented baseline KNN classifier
- Tuned KNN hyperparameters using GridSearchCV and RandomizedSearchCV
- Compared KDTree and BallTree neighbor search algorithms

5. Visualizations

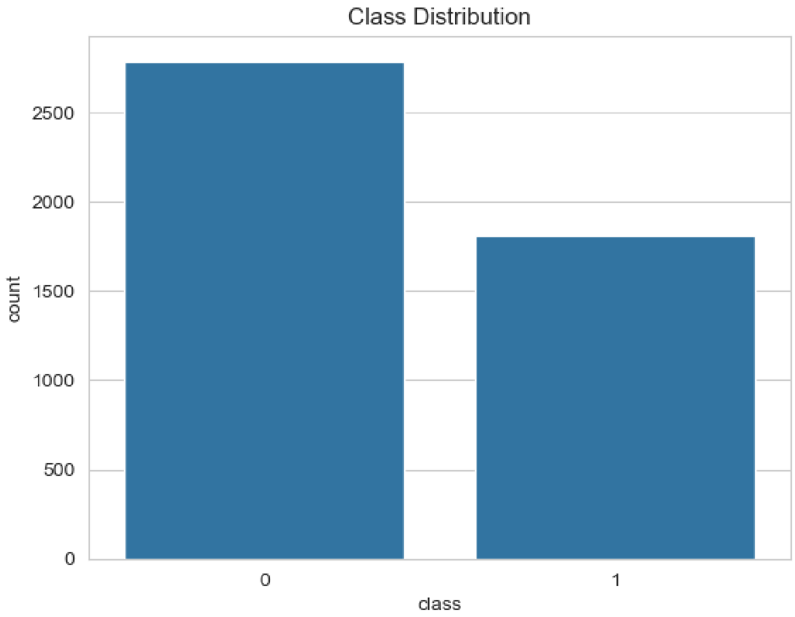

Figure 1: Class Distribution of Spam and Non-Spam Emails

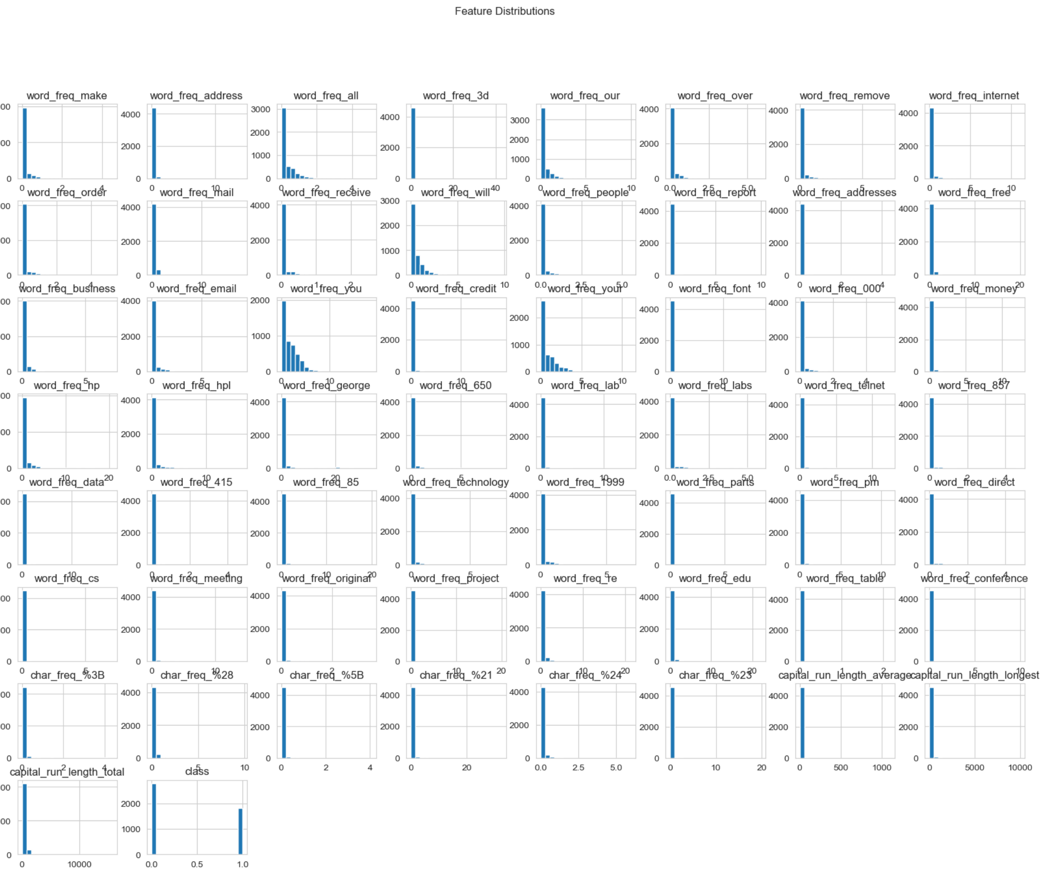

Figure 2: Feature Distribution Plot (Sample Features)

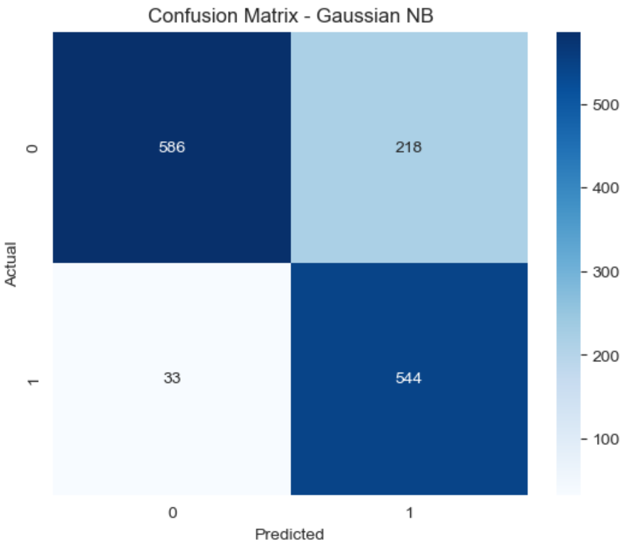

Figure 3: Confusion Matrix for Gaussian Naïve Bayes

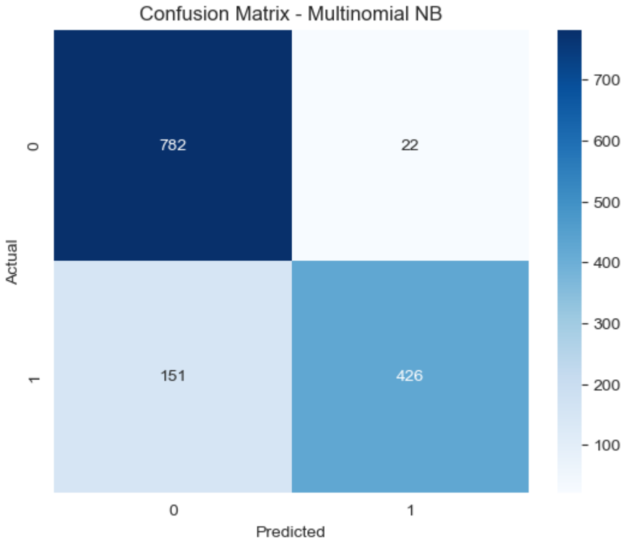

Figure 4: Confusion Matrix for Multinomial Naïve Bayes

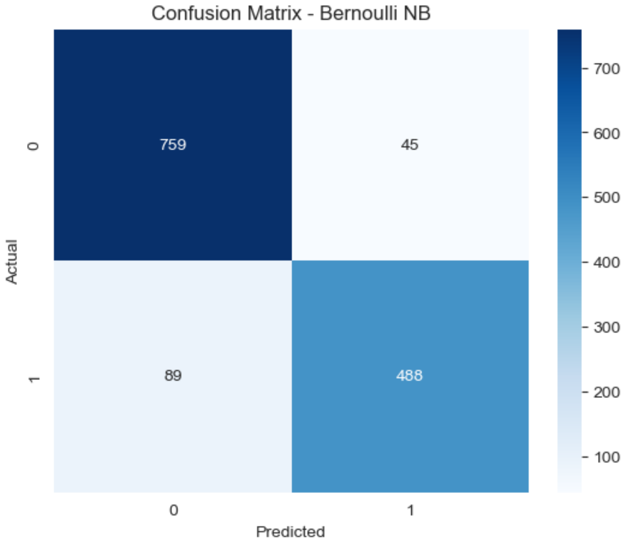

Figure 5: Confusion Matrix for Bernoulli Naïve Bayes

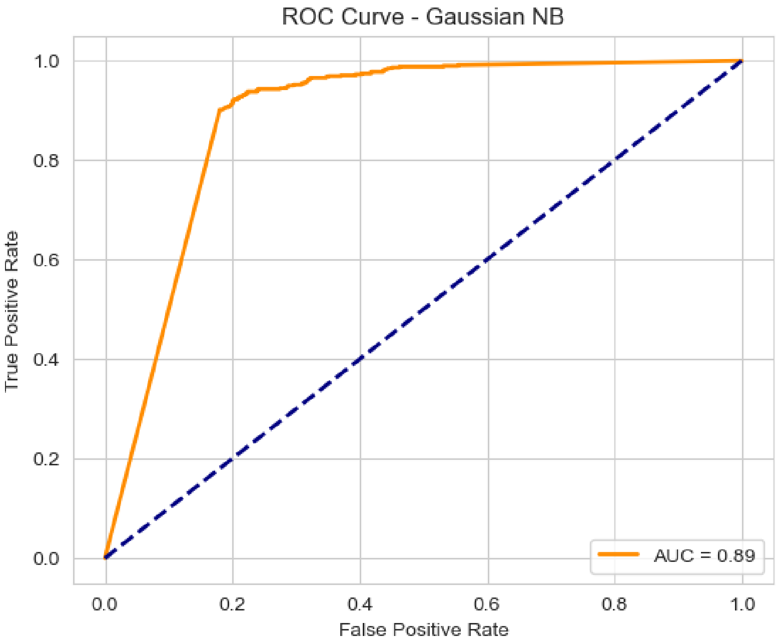

Figure 6: ROC Curve for Gaussian Naïve Bayes

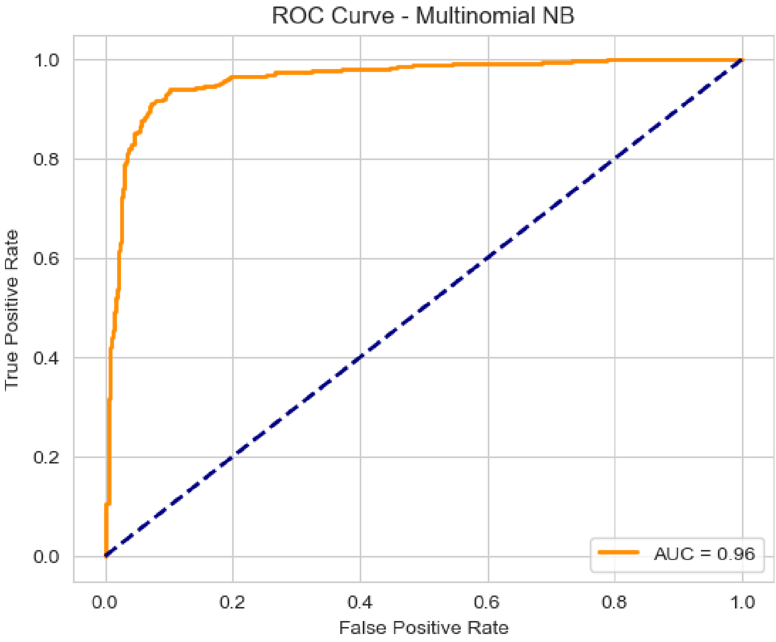

Figure 7: ROC Curve for Multinomial Naïve Bayes

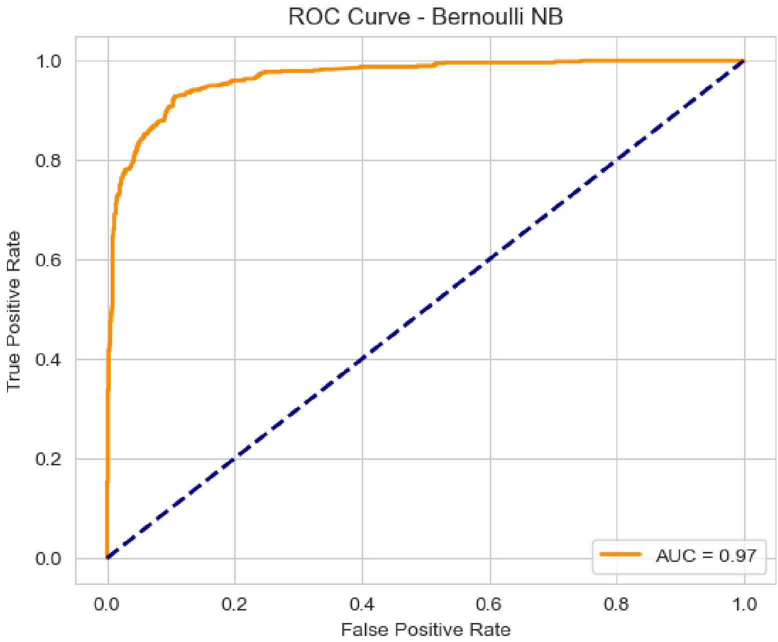

Figure 8: ROC Curve for Bernoulli Naïve Bayes

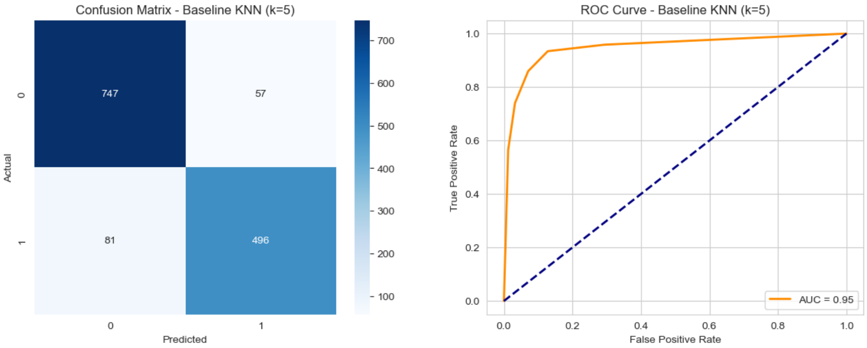

Figure 9: Accuracy vs. k Plot for KNN

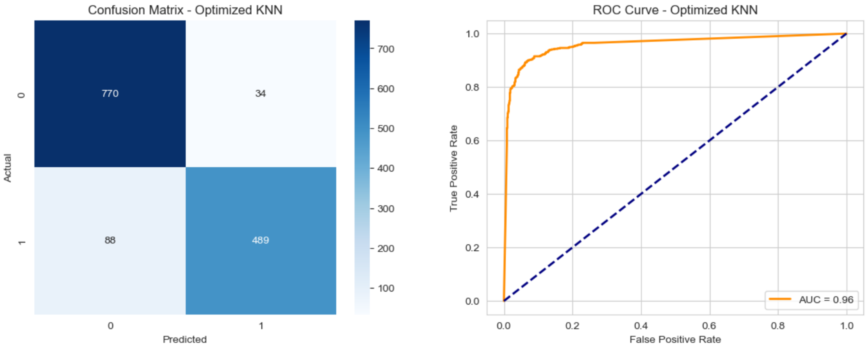

Figure 10: Training vs Validation Accuracy for KNN

6. Performance Tables

Table 1: Naïve Bayes Performance Metrics

Metric	**Gaussian NB**	**Multinomial NB**	**Bernoulli NB**
Accuracy	0.8339	0.7763	0.8762
Precision	0.7178	0.7199	0.8716
Recall	0.9532	0.7080	0.8044
F1 Score	0.8189	0.7139	0.8367
Specificity	0.5495	0.6406	0.6382
Training Time (s)	0.0042	0.0027	0.0045

Table 2: KNN Hyperparameter Tuning

Search Method	**Best Parameters**	**Best CV Accuracy**
Grid Search	k=15, distance, KDTree	0.9209
Randomized Search	k=15, distance, BallTree	0.9209

Table 3: KNN Performance using KDTree

Metric	**Value**
Optimal k	15
Accuracy	0.9175
Precision	0.9065
Recall	0.8815
F1 Score	0.8939
Training Time (s)	0.0256
Prediction Time (s)	0.2356

Table 4: KNN Performance using BallTree

Metric	**Value**
Optimal k	15
Accuracy	0.9175
Precision	0.9065
Recall	0.8815
F1 Score	0.8939
Training Time (s)	0.0075
Prediction Time (s)	0.1619

Table 5: Comparison of Neighbor Search Algorithms

Criterion	**KDTree**	**BallTree**
Accuracy	0.9175	0.9175
Training Time (s)	0.0256	0.0075
Prediction Time (s)	0.2356	0.1619
Memory Usage	Low / Medium	Medium / High

7. Overfitting and Underfitting Analysis

- Small values of k resulted in overfitting
- Large values of k caused underfitting
- Hyperparameter tuning improved generalization

8. Bias–Variance Analysis

- Naïve Bayes shows higher bias due to feature independence assumption
- KNN exhibits higher variance for small k values
- Hyperparameter tuning balances bias–variance trade-off

9. Observations and Conclusion

Naïve Bayes classifiers were computationally efficient, while tuned KNN models achieved higher accuracy. BallTree demonstrated better computational efficiency compared to KDTree. Proper preprocessing, visualization, and hyperparameter tuning significantly improved model performance and generalization.

References

- Scikit-learn – Naïve Bayes Documentation
- Scikit-learn – KNN Documentation
- Scikit-learn – Hyperparameter Optimization
- Kaggle – Spambase Dataset