|  |
| --- |
| **Lab: 05 Implementation of Supervised Learning Algorithm**  **Naïve Bayes** |

Dowload the SMS spam collection dataset from the provided link

<https://www.kaggle.com/datasets/ozlerhakan/spam-or-not-spam-dataset>

Task 01

Read and Pre-process the data then split the data into training testing dataset with the ratio of 75/25

Tasks 02

Build a classification model for the dataset using Naive Bayes classifier and calculate the accuracy score and confusion matrix for the model created

Task 03

Repeat the same process, mentioned in point 3, for KNN algorithm as well

Task 04

Write a proper comparative analysis for both of the models created.

|  |  |
| --- | --- |
| **Multinomial Naive Bayes** | **K-Nearest Neighbors (KNN)** |
| **Accuracy**:  The Multinomial Naive Bayes model achieved an accuracy of 99.07% on the test dataset. | **Accuracy**:  The K-Nearest Neighbors model achieved an accuracy of approximately 88.53% on the test dataset. |
| **Confusion Matrix**:  The confusion matrix shows that it correctly predicted 624 of non-spam emails and 119 of spam emails. It misclassified 0 of non-spam emails as spam and 7 of spam emails as non-spam. | **Confusion Matrix**:  The confusion matrix reveals that it correctly predicted 572 of non-spam emails and 92 of spam emails. It misclassified 52 of non-spam emails as spam and 34 of spam emails as non-spam. |

**Comparative Analysis:**

**Accuracy:**

The Multinomial Naive Bayes model achieved a slightly higher accuracy compared to the KNN model.

**Interpretability:**

Naive Bayes is probabilistic and provides clear insights into how the classification decision is made. In contrast, KNN does not offer such direct interpretability.

**Computational Complexity:**

Naive Bayes is computationally efficient, making it suitable for large datasets and real-time applications. KNN, on the other hand, can be computationally expensive, especially with a large number of training samples.

**Robustness**:

KNN tends to be more robust when dealing with noisy or overlapping data because it does not assume any specific data distribution. Naive Bayes, however, assumes feature independence and might be affected by irrelevant or correlated features.

**Parameter Sensitivity**:

KNN's performance is highly dependent on the choice of the number of neighbors (k). Tuning this hyperparameter is crucial for achieving optimal results. Naive Bayes has fewer hyperparameters to tune, making it simpler in this regard.