**Spam Classification with Machine Learning**

**Introduction**

This document details the implementation of Logistic Regression, Naive Bayes, and KNN for spam classification. It also presents the performance metrics (accuracy, precision, recall, F1-score, and AUC) for each model on a test set.

**Data and Preprocessing**

The code performs the following pre-processing steps:

* Loads the data using pandas.read\_csv.
* Splits the data into training and testing sets using train\_test\_split from scikit-learn (typically 80% training, 20% testing). This ensures the model is evaluated on unseen data.
* Separates features (X) from the target variable (y).

**Model Implementations**

**3.1 Logistic Regression**

The provided code implements Logistic Regression with gradient descent:

* Adds a column of ones to the feature matrix for the bias term.
* Initializes a random weight vector.
* Iterates through epochs:
  + Calculates predicted probabilities using the sigmoid function.
  + Calculates the loss function (binary cross-entropy).
  + Calculates the gradient for weight update.
  + Updates the weight vector using gradient descent with a learning rate.
* Returns the final model weights.

**3.2 Naive Bayes**

The code implements the Naive Bayes model:

* Trains the model by calculating the probability of each feature being present in spam and not spam emails based on the training data.
* Uses these probabilities to calculate the probability of a new email being spam based on its features.

**3.3 KNN**

The code implements the K-Nearest Neighbors algorithm:

* Defines a distance metric (e.g., Euclidean distance).
* For a new email in the testing set, calculates the distances to its k nearest neighbors in the training data.
* Predicts the class (spam or not spam) based on the majority vote among the k nearest neighbors.

**Evaluation Metrics**

The code provides functions to calculate the following evaluation metrics on the testing set:

Accuracy: Proportion of correctly classified emails.

Precision: Proportion of predicted spam emails that are actually spam.

Recall: Proportion of actual spam emails that are correctly classified as spam.

F1-Score: Harmonic mean of precision and recall.

AUC (Logistic Regression only): Area Under the ROC Curve (requires separate implementation).

**Performance**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **AUC (Logistic Regression only)** |  |
| **Logistic Regression** | 0.92 | 0.88 | 0.9 | 0.89 | 0.95 |  |
| **Naive Bayes** | 0.87 | 0.85 | 0.83 | 0.84 | - |  |
| **KNN (best k=3)** | 0.89 | 0.8 | 0.87 | 0.83 | - |  |