Introduction
For this project, we try to predict text messages as "spam" or "not spam" using three different machine learning models: Linear Regression, Logistic Regression, and Naive Bayes. The dataset used is sourced from this GitHub repo. It contains 1999 entries of text messages, each of which is labeled either as "not spam" or "spam". The goal of this project is to demonstrate the efficacy of machine learning algorithms in tackling the challenge of spam detection in real-world scenarios.

We employ the scikit-learn library, a popular machine learning toolkit, to train, test, and validate the performance of each model on the dataset. Through this project, we aim to explore and compare different classification techniques to identify the most suitable model for spam classification tasks.

Problem Formulation
Given the dataset of text messages, the task is to build a binary classification model that predicts whether a message is "spam" (labeled as 1) or "not spam" (labeled as 0). The input to the models are raw text messages, while the output for each input is a binary label indicating the message's classification.

Approaches and baselines
We split the raw data into a 4:1 ratio for training and test data, respectively. To process the text data, we utilize vectorization techniques, particularly the Term Frequency-Inverse Document Frequency (TF-IDF) method, to convert the input strings into feature vectors.

As a baseline approach, we implement the trivial model that predicts all messages as "not spam". This is akin to the example we discussed in class, where we predicted everyone as not having cancer. In this scenario, "not spam" represents the majority class, and the baseline gives us a measure of how well our classifiers perform compared to a naive, non-discriminative approach.

For the non-trivial models, we use the default parameters provided by scikit-learn. To evaluate the models' performance, we utilize 10-fold cross-validation, a widely used technique that ensures robust validation by partitioning the data into ten subsets and iteratively training and testing the models.

Evaluation Metric
We use the training and validated accuracy of the 3 non-trivial models through the scikit-learn metrics library. We compare it against the trivial model’s accuracy to gauge the success of the classifier.

Results

The table below summarizes the training accuracy and the validated results for the 4 classifiers.

Accuracy	Trivial Model	Linear Reg.	Logistic Reg.	Naive Bayes
Training	0.8599	0.7354	0.905	0.925
Validation	0.8599	-3.135e+24	0.9224	0.9144

Conclusion

Based on the results, it is evident that linear regression has the worst performance among the four classifiers, even when compared against the trivial model. Both logistic regression and naive Bayes outperform the trivial classifier. Out of the three non-trivial models, logistic regression demonstrates the best performance in terms of accuracy.

One notable observation is the apparent case of strong overfitting in the linear regression model, as indicated by the significant difference between the training and validated accuracy.

Overall, this mini-project offers valuable insights into the application of machine learning models for spam classification tasks. It highlights the importance of selecting appropriate algorithms and fine-tuning hyperparameters to achieve optimal performance.

Future Scope

While this project provides a strong foundation for spam classification, there are numerous avenues for future exploration. Some potential areas of improvement include:

- Experimenting with more advanced machine learning algorithms, such as support vector machines, decision trees, and random forests.
- Fine-tuning hyperparameters to mitigate overfitting and improve generalization performance.
- Investigating deep learning approaches, such as recurrent neural networks (RNNs) and transformers, for more sophisticated text classification tasks.
- Exploring ensemble techniques to combine the strengths of multiple classifiers for enhanced performance.