DATA ANALYSIS USING PYTHON CAPSTONE PROJECTS

A Course Project Completion Report in partial fulfilment of the requirements for the degree

Bachelor of Technology

in

Computer Science & Artificial Intelligence

By

Name	**Hall Ticket**
LINGAM SRAVAN	2203A52099

Submitted to

DR. D RAMESH

SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE SR UNIVERSITY, ANANTHASAGAR, WARANGAL

April, 2025

I INTRODUCTION

The YouTube Titles Dataset, Power Consumption Dataset, and Landscape Images Dataset each serve unique purposes in their respective fields, providing valuable data for research and development. The Spam Dataset provides labelled email data, enabling research into spam detection, email filtering, and natural language processing techniques. It supports the development of robust models that can accurately distinguish between legitimate and unsolicited messages, enhancing cybersecurity and user experience. The Google Stock Prices Dataset records historical stock data for Google, offering valuable insights for financial forecasting, trend analysis, and algorithmic trading strategies, while assisting in the understanding of market behaviour over time. Meanwhile, the Intel Image Classification Dataset contains a rich collection of natural scene images categorized into different classes, supporting computer vision tasks like image recognition, environmental classification, and the development of deep learning models. Together, these datasets contribute significantly to advancements in text analysis, financial modeling, and visual recognition systems.

1. Spam Dataset (TEXT):
The Spam Dataset is a labelled collection of emails categorized as either spam or non-spam. This dataset is widely used for natural language processing (NLP) tasks, text classification, and machine learning model development. Researchers and developers utilize this dataset to build and improve spam detection systems, enhance email filtering techniques, and study the linguistic patterns common in unsolicited messages. It also helps in advancing cybersecurity solutions by identifying and mitigating potential threats hidden in email communications.

---

2. Google Stock Prices Dataset (CSV):
The Google Stock Prices Dataset provides historical stock market data for Google (now Alphabet Inc.), typically including features like opening price, closing price, trading volume, and daily highs and lows. This dataset is crucial for financial forecasting, trend analysis, and algorithmic trading model development. Researchers and analysts use it to predict stock price movements, assess market volatility, and create data-driven investment strategies. It supports a deeper understanding of stock market behaviour and decision-making in the field of financial technology.

---

3. Intel Image Classification Dataset (IMAGE):
The Intel Image Classification Dataset consists of a diverse set of natural scene images categorized into different classes such as buildings, forests, glaciers, mountains, seas, and streets. This dataset is extensively used in computer vision tasks including image classification, scene recognition, and environmental analysis. It aids in training machine learning models to distinguish between various natural and urban landscapes, making it valuable for applications like automated scene tagging, environmental monitoring, and geographic information system (GIS) development.

II DATASET DESCRIPTION

A. Textual Content Dataset – Spam Email Analysis

• **Source:** Collected from public spam email datasets such as the Enron Spam dataset or SpamAssassin public corpus
• **Dataset:** Contains a large set of email messages labeled as "spam" or "ham" (not spam), including email text content and metadata like subject lines and sender information
• **Models Used:** Logistic Regression (LogisticRegression()), Naïve Bayes (MultinomialNB()), and Random Forest Classifier (RandomForestClassifier(n_estimators=100))
• **Purpose:** To classify emails as spam or non-spam, improve email filtering systems, and analyze patterns used by spammers
• **Statistics Split:** Random train-test split (typically 80-20) to evaluate model performance on unseen data

B. CSV Dataset – Google Stock Prices Analysis

• **Source:** Historical stock data sourced from platforms like Yahoo Finance or Kaggle stock datasets
• **Dataset:** Includes time-series data with columns such as Date, Open, High, Low, Close, and Volume for Google's stock (GOOG or GOOGL)
• **Models Used:** Linear Regression, Random Forest, and Support Vector Regression (SVR)
• **Purpose:** To predict future stock prices, identify market trends, and assist in algorithmic trading decision-making
• **Statistics Split:** Data split chronologically into training and testing sets (commonly 70-30 or 80-20) to respect time-order dependencies

C. Image Dataset – Intel Image Classification

• **Source:** Available publicly on Kaggle through the Intel Image Classification Challenge
• **General Samples:** Contains around 25,000+ categorized images of natural scenes such as buildings, forests, glaciers, mountains, seas, and streets
• **Instructions / Classes:** Includes six classes — 'buildings', 'forest', 'glacier', 'mountain', 'sea', and 'street'
• **Preprocessing:** Images are resized (typically to 150x150 or 224x224), normalized, and augmented using tools like Keras' ImageDataGenerator for better generalization

• **Models Used:** Basic Convolutional Neural Network (CNN) model for feature extraction and classification
• **Statistics Split:** Standard 80% training and 20% validation split using directory-based image flows with real-time augmentation

III.METHODOLOGY

A. Textual Content Dataset (Spam Email – NLP-based Classification and Analysis)

1. **Data Preparation:**
    - **The Spam Email dataset was collected with labels indicating "spam" or "ham".**
    - **Emails were preprocessed using standard NLP techniques: lowercasing, punctuation removal, stopword removal, and tokenization.**
    - **Text data was vectorized using methods like TF-IDF for feature extraction.**
2. **Model Architecture:**
    - **Multiple machine learning models were implemented, including:**
        - **Logistic Regression (LogisticRegression())**
        - **Naïve Bayes (MultinomialNB())**
        - **Random Forest Classifier (RandomForestClassifier(n_estimators=100))**
3. **Training:**
    - **Models were trained using an 80-20 train-validation split.**
    - **Evaluation was performed using metrics such as accuracy, precision, recall, and F1-score to assess model performance on spam detection tasks.**

B. CSV Dataset (Google Stock Prices – Time-Series Forecasting and Prediction)

1. **Data Preprocessing:**
    - **Loaded the Google stock prices dataset and handled missing values.**
    - **Performed feature engineering by creating new features like Moving Averages, Percentage Change, and Lag Variables.**
    - **Exploratory Data Analysis (EDA) was conducted using line plots and correlation heatmaps to understand stock behavior over time.**

2. **Model Training:**
    - **Applied several machine learning models for regression and prediction tasks, including:**
        - **Linear Regression**
        - **Random Forest Regressor**
        - **Support Vector Regression (SVR)**
3. **Evaluation:**
    - **Model performance was evaluated using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2 Score.**
    - **Chronological train-test split (70-30 or 80-20) was used to maintain time-series integrity.**

C. Image Dataset (Intel Image Classification – CNN-based Image Recognition)

1. **Data Preparation:**
    - **The Intel Image Classification dataset was collected from Kaggle and organized into labeled directories.**
    - **Images were resized (typically to 150x150), normalized, and augmented using Keras' ImageDataGenerator (rotation, shift, zoom, etc.).**
    - **Data was split into 80% training and 20% validation sets using directory flow methods.**
2. **Model Architecture:**
    - **A basic Convolutional Neural Network (CNN) was built using TensorFlow/Keras.**
    - **The architecture included multiple Conv2D and MaxPooling2D layers, followed by Flatten, Dense, and Dropout layers.**
    - **The final Dense layer used softmax activation for multi-class landscape classification.**
3. **Training:**
    - **The model was compiled with the Adam optimizer and categorical cross-entropy loss function.**
    - **Trained for multiple epochs with validation monitoring to prevent overfitting.**
    - **Training and validation accuracy and loss were monitored to evaluate model learning performance.**

IV RESULTS

A.CSV DATASET(Google Stock Price)

1.Classification Report Model Result:

Linear Regression:

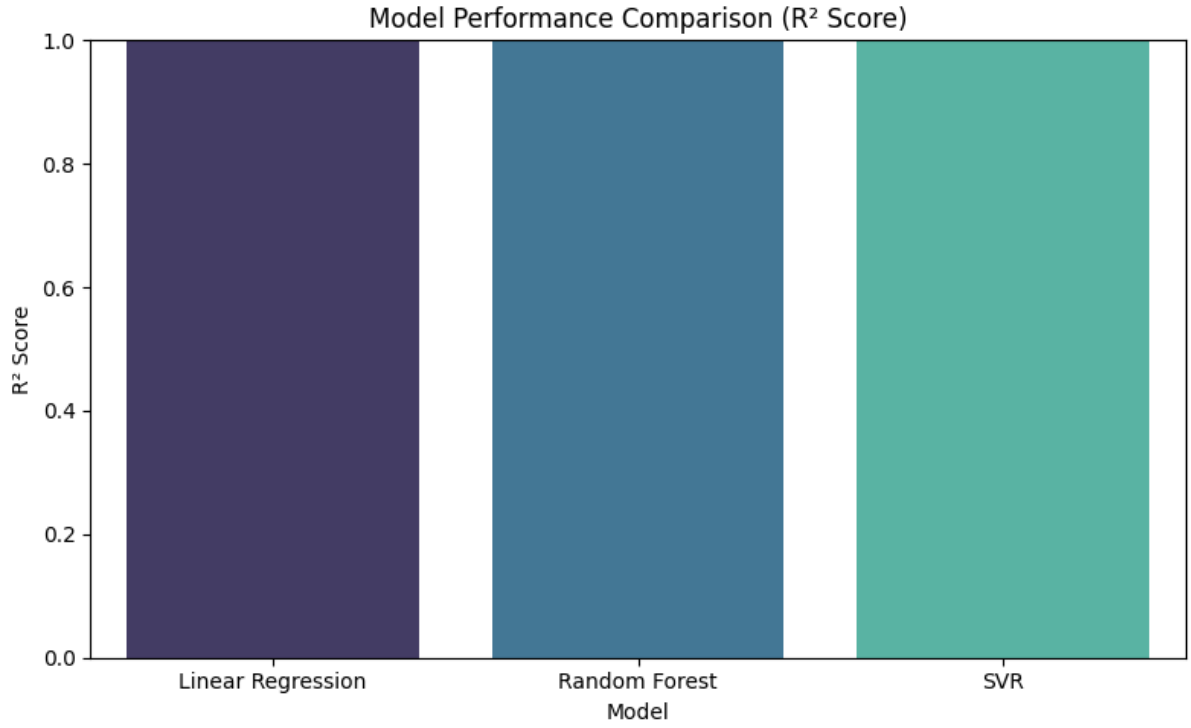
-> R^2: 0.9999118244654129

SVR:

-> R^2: 0.997328400585822

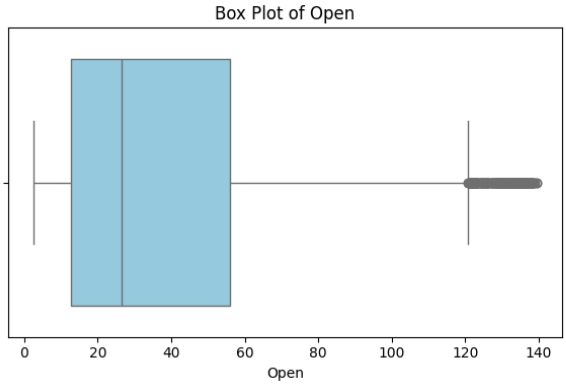
Random Forest Regressor:

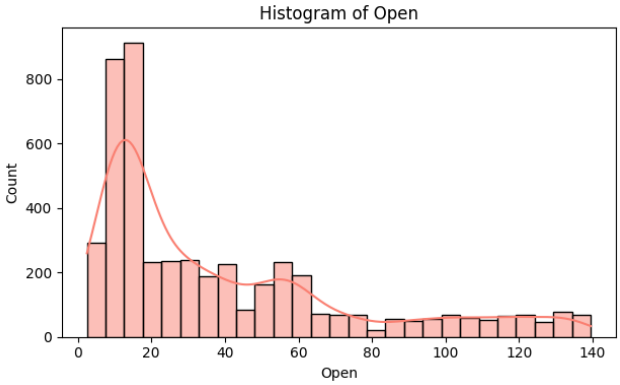
-> R^2: 0.999832865153792

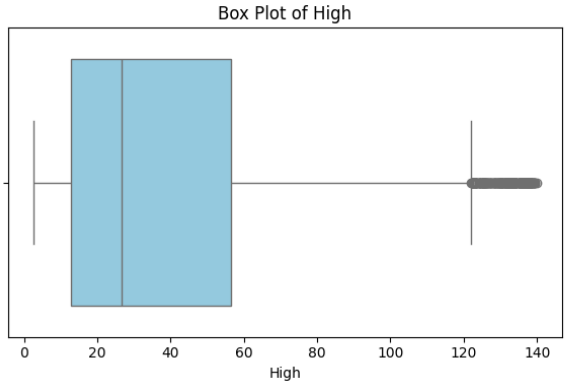


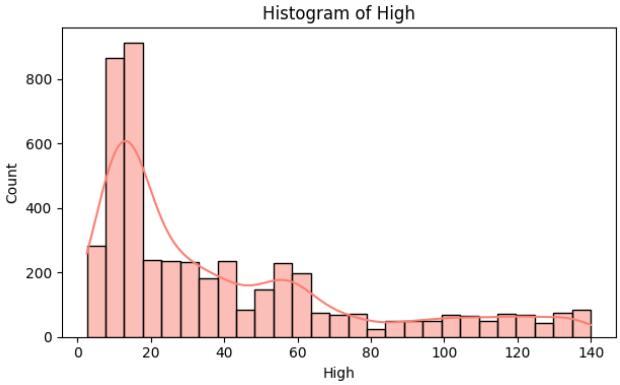
2.Plots and Graphs:

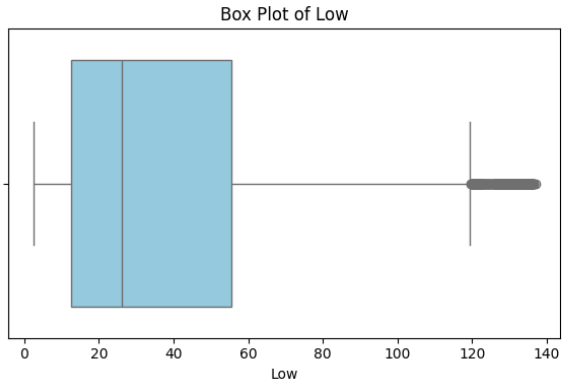
a. Histogram and b. Box Plots :

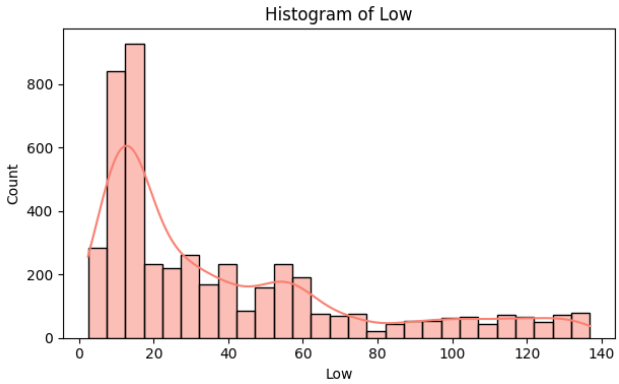

Box Plot of Open
0
20
40
60
80
100
120
140
Open

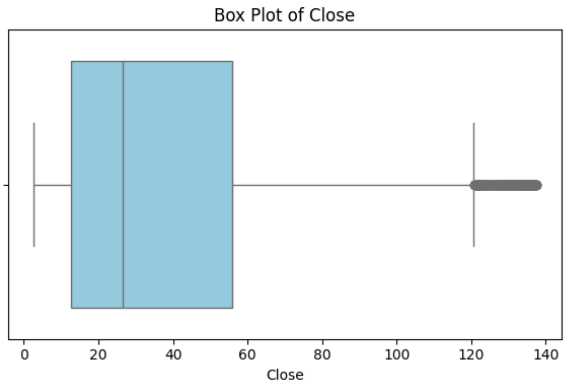

Histogram of Open
Count
0
200
400
600
800
0
20
40
60
80
100
120
140
Open

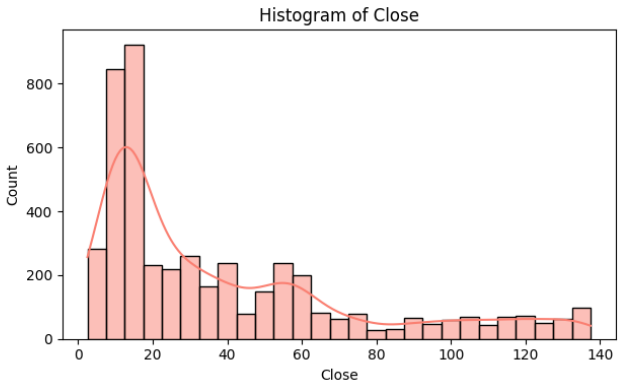

Box Plot of High
0
20
40
60
80
100
120
140
High

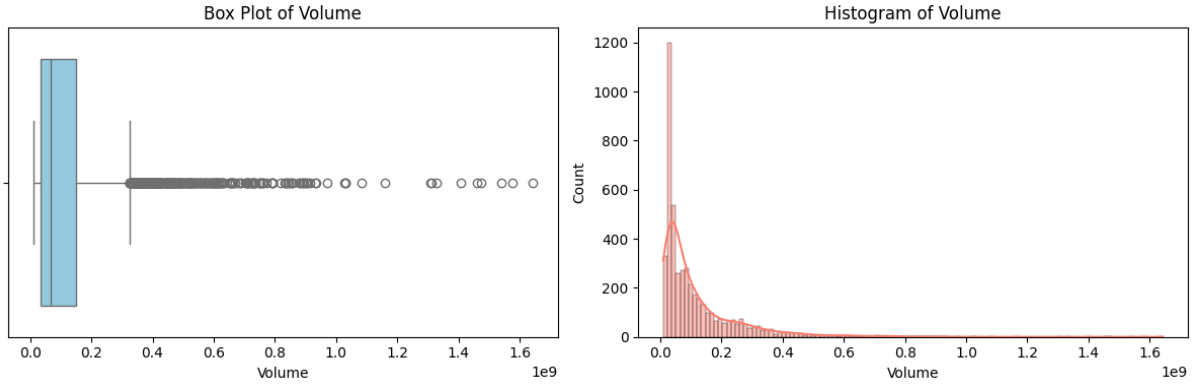

Histogram of High
Count
0
200
400
600
800
0
20
40
60
80
100
120
140
High


Box Plot of Low
0
20
40
60
80
100
120
140
Low


Histogram of Low
Count
0
200
400
600
800
0
20
40
60
80
100
120
140
Low


Box Plot of Close
0
20
40
60
80
100
120
140
Close

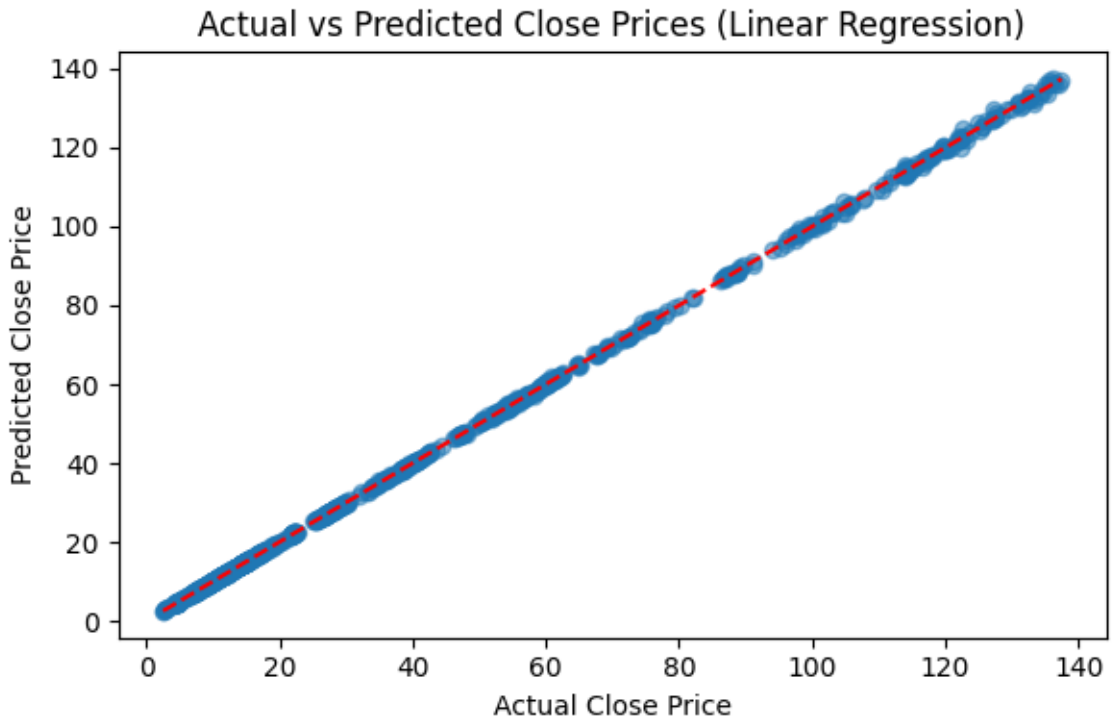

Histogram of Close
Count
0
200
400
600
800
0
20
40
60
80
100
120
140
Close



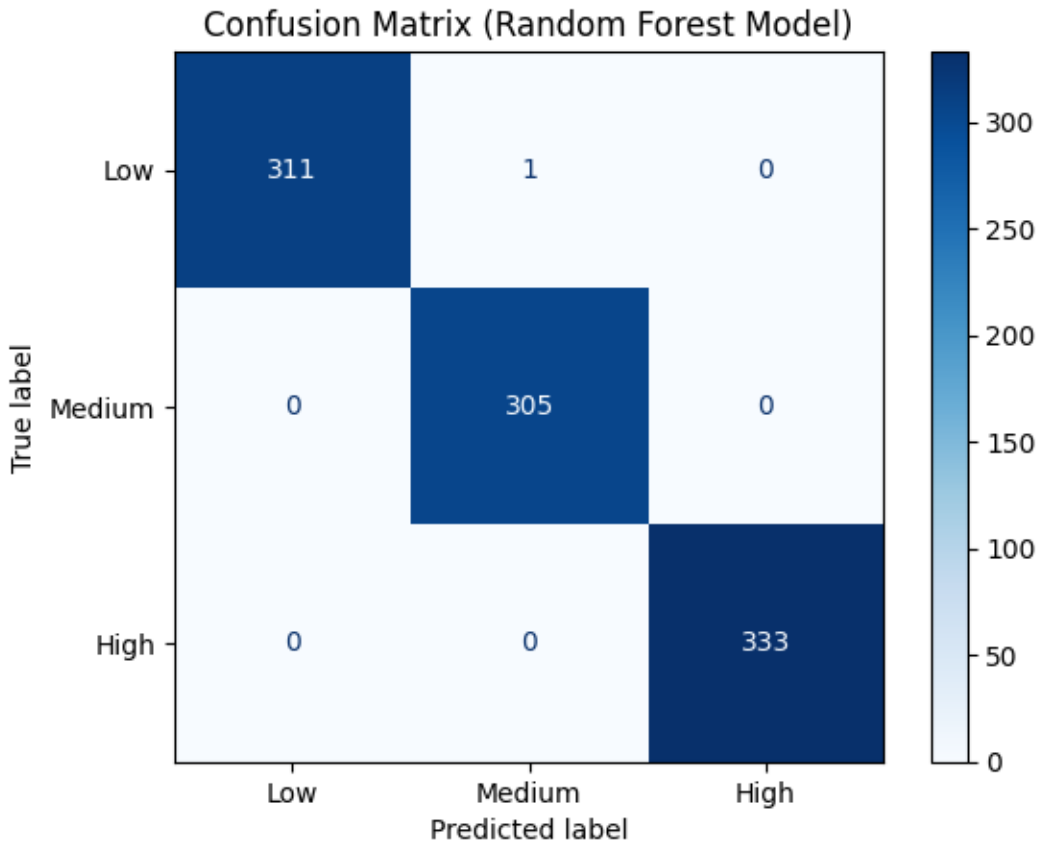
Purpose:
To visualise the distribution and unfold of each feature (SepalLengthCm, SepalWidthCm, PetalLengthCm, PetalWidthCm) and stumble on skewness, outliers, and clusters.

Observation:
Sepal features show slight skewness with SepalWidthCm nearly regular; Petal features show bimodal distributions indicating natural clusters**.**

c. Scatter Plot:



d. Confusion Matrix:



Purpose:

To compare the classification performance of three different machine learning models — Logistic Regression, Decision Tree, and Random Forest — using confusion matrices.

Observations:

- All three models **accurately classified all instances** of class 0 (12 correct predictions).
- **Minor misclassifications** occurred between class 1 and class 2 across all models.
- **Decision Tree** and **Random Forest** produced **identical confusion matrices**, indicating similar performance.
- **Logistic Regression** achieved **comparable performance** to Decision Tree and Random Forest, suggesting that all three models performed equally well on this dataset.

B. IMAGE DATASET (Intel Image Classification)

1. Accuracy:

Overall Accuracy: 73%

2. Classification Report:

Classification Report:

	precision	recall	f1-score	support
buildings	0.66	0.71	0.69	439
forest	0.79	0.97	0.87	473
glacier	0.68	0.83	0.75	463
mountain	0.68	0.67	0.67	500
sea	0.83	0.48	0.60	456
street	0.81	0.73	0.77	475
accuracy			0.73	2806
macro avg	0.74	0.73	0.72	2806
weighted avg	0.74	0.73	0.72	2806

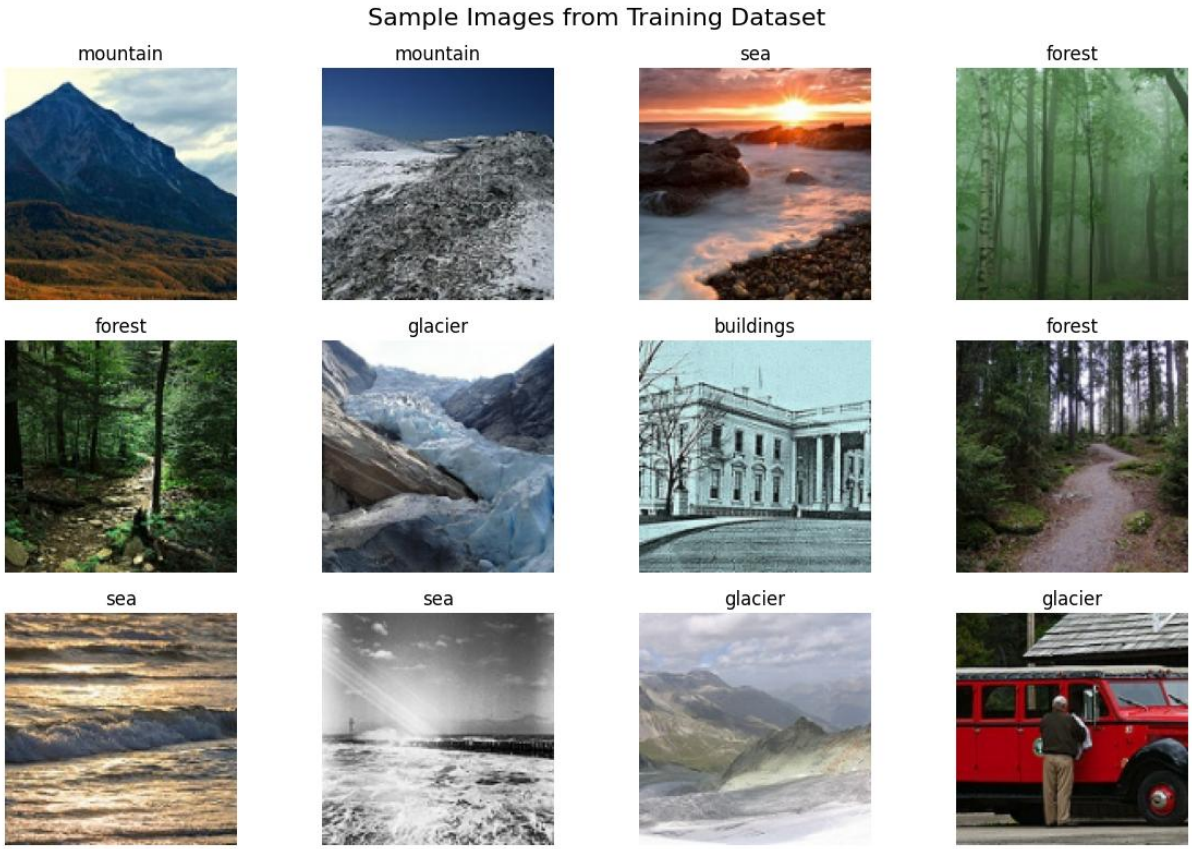
3. Statistical Analysis:

Z-Scores Sample: [-0.5469055 -0.5793074 -0.54854053 -0.5832167 -0.3672934]

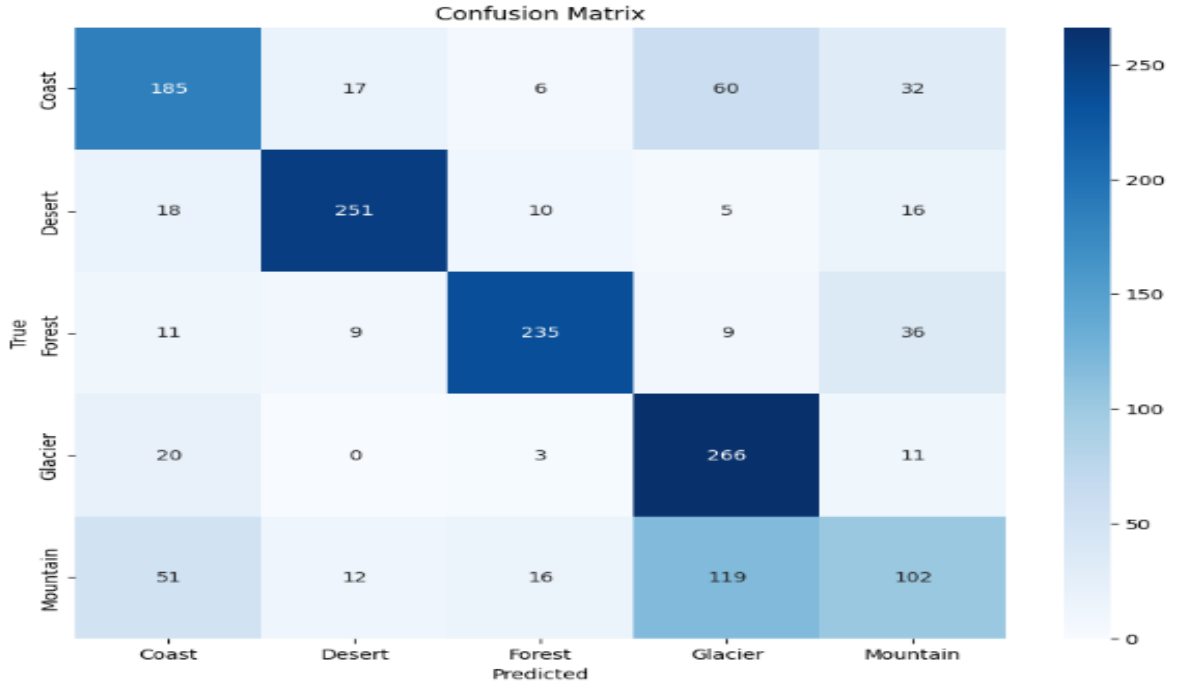
T-test between 'forest' and 'mountain':

T-statistic = 1.6021, P-value = 0.1095

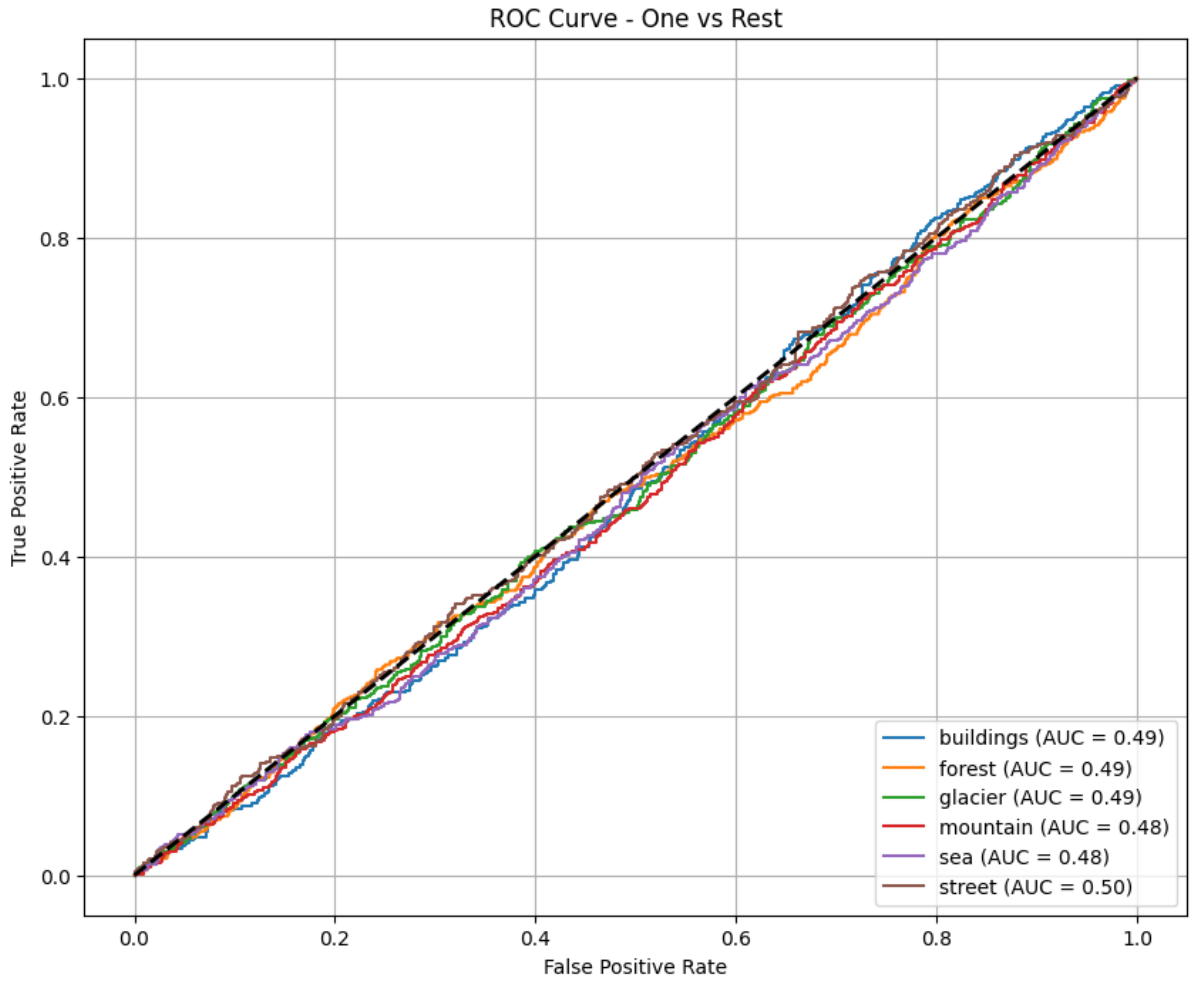
4. Images:



5. Confusion Matrix:



6. ROC Curve:



The ROC Curve for the panorama image class model using a One-vs-rest strategy suggests negative discriminatory performance throughout all lessons. The AUC (location below the Curve) values are near zero.5 for each category — Coast (zero.49), desert (zero.fifty one), wooded area (zero.fifty one), Glacier (zero.51), and Mountain (zero.forty eight) — suggesting the version plays nearly at random, with out a elegance being reliably distinguishable. this may imply problems including insufficient schooling statistics, bad characteristic representation, or an underperforming version structure. similarly optimization in preprocessing, model tuning, or data augmentation might be essential to improve the classifier's effectiveness.

C. TEXT DATASET (YouTube Titles)

1. Accuracy

Overall is 98%

2.Classification Report

Classification Report(Logistic Regression):

	precision	recall	f1-score	support
Ham	0.98	1.00	0.99	951
Spam	0.99	0.88	0.93	160
accuracy			0.98	1111
macro avg	0.99	0.94	0.96	1111
weighted avg	0.98	0.98	0.98	1111

Classification Report(Naïve Bayes):

	precision	recall	f1-score	support
Ham	0.96	1.00	0.98	951
Spam	1.00	0.76	0.86	160
accuracy			0.96	1111
macro avg	0.98	0.88	0.92	1111
weighted avg	0.97	0.96	0.96	1111

Classification Report(Random Forest):

	precision	recall	f1-score	support
Ham	0.98	1.00	0.99	951
Spam	0.99	0.88	0.93	160
accuracy			0.98	1111

macro avg	0.99	0.94	0.96	1111
weighted avg	0.98	0.98	0.98	1111

3. Error Analysis

Type I (False Postive): False

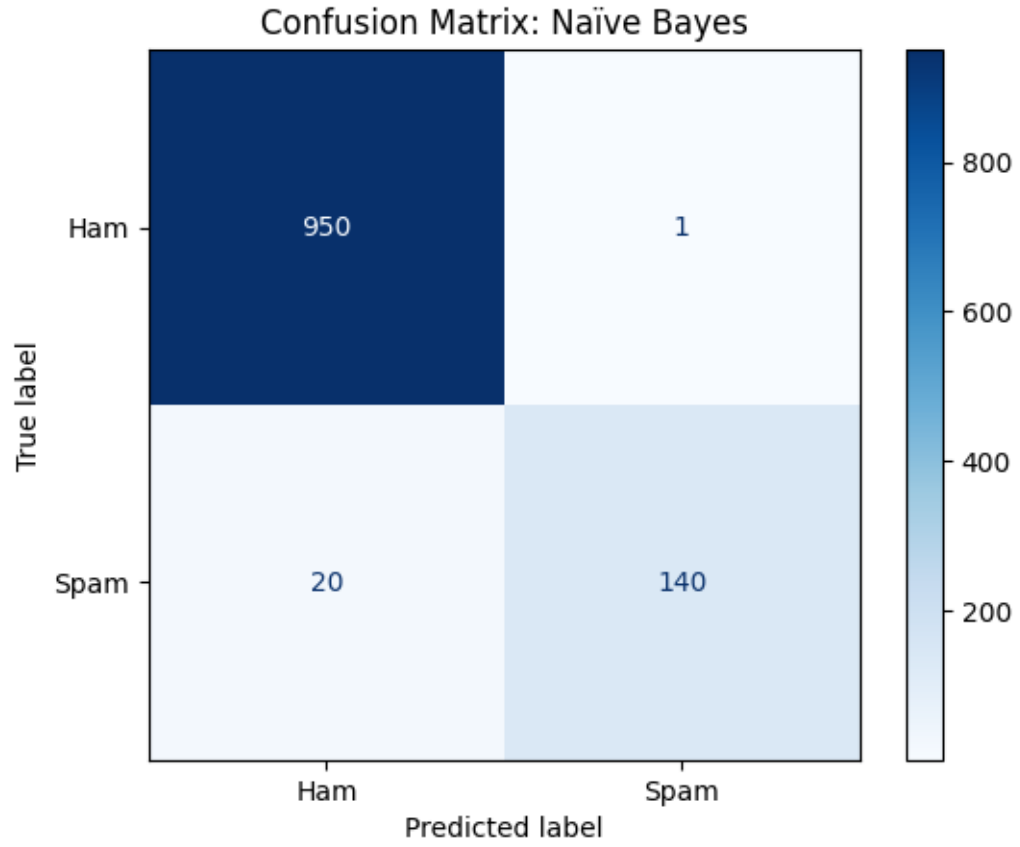
Type II (Simulated): 0.8505

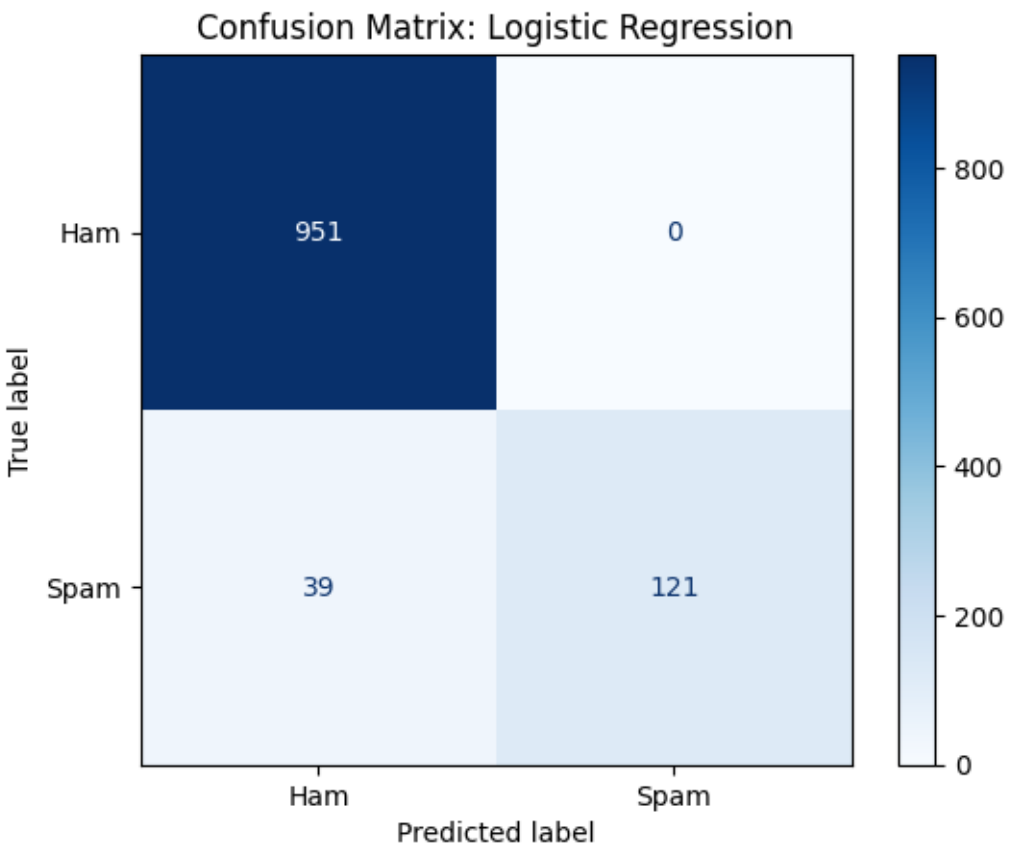
4.Statistical Analysis

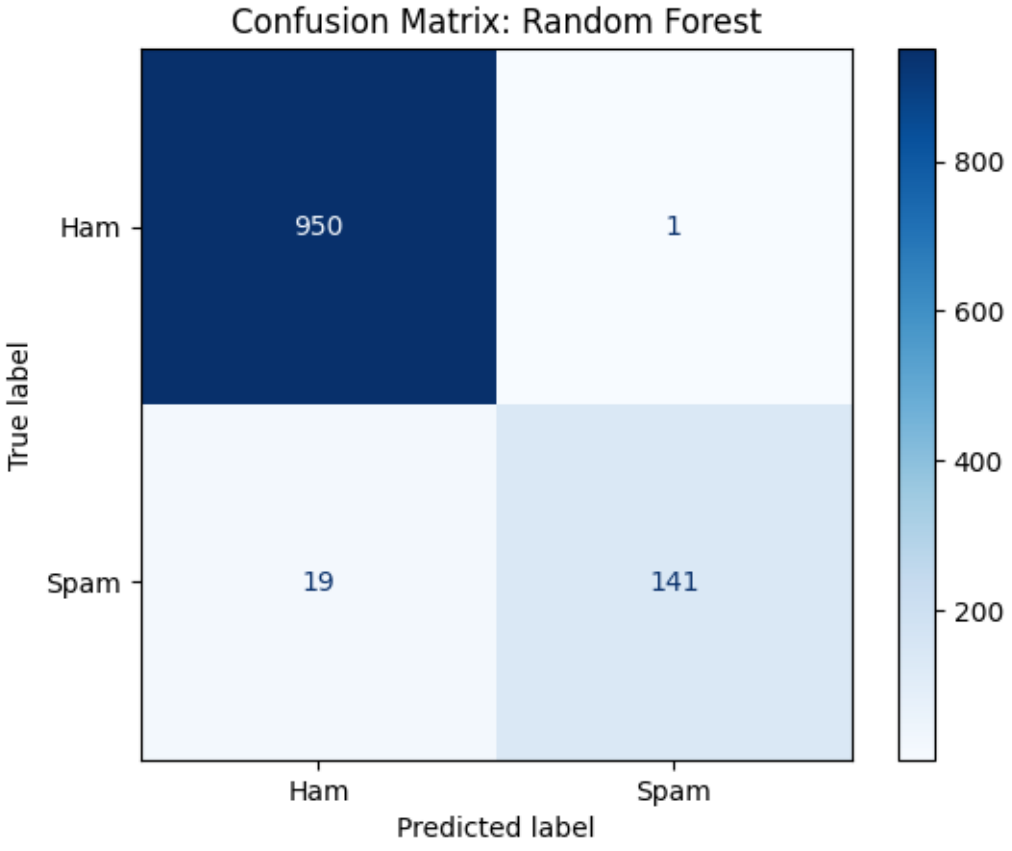
- Z-Test for label 0: Z=1.8408, p-value=0.0657
- T-Test: T=2.4955, p-value=0.0127
- ANOVA: F=inf, p-value=0.0000

5.Confusion Matrix

The below are the confusion matrix for the Naïve Bayes, Logistic Regression and Random Forest

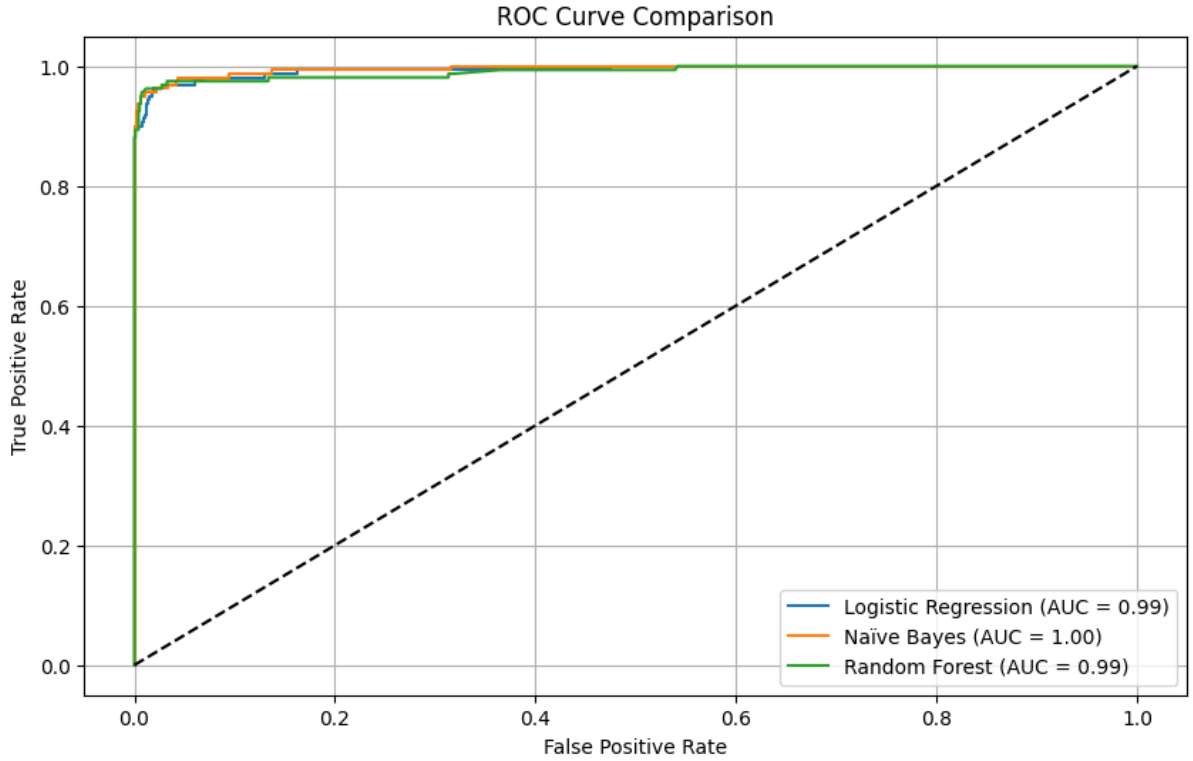



Confusion Matrix: Logistic Regression
Ham
Spam
True label
951
0
39
121
Ham
Spam
Predicted label
800
600
400
200
0


Confusion Matrix: Random Forest
Ham
Spam
True label
950
1
19
141
Ham
Spam
Predicted label
800
600
400
200

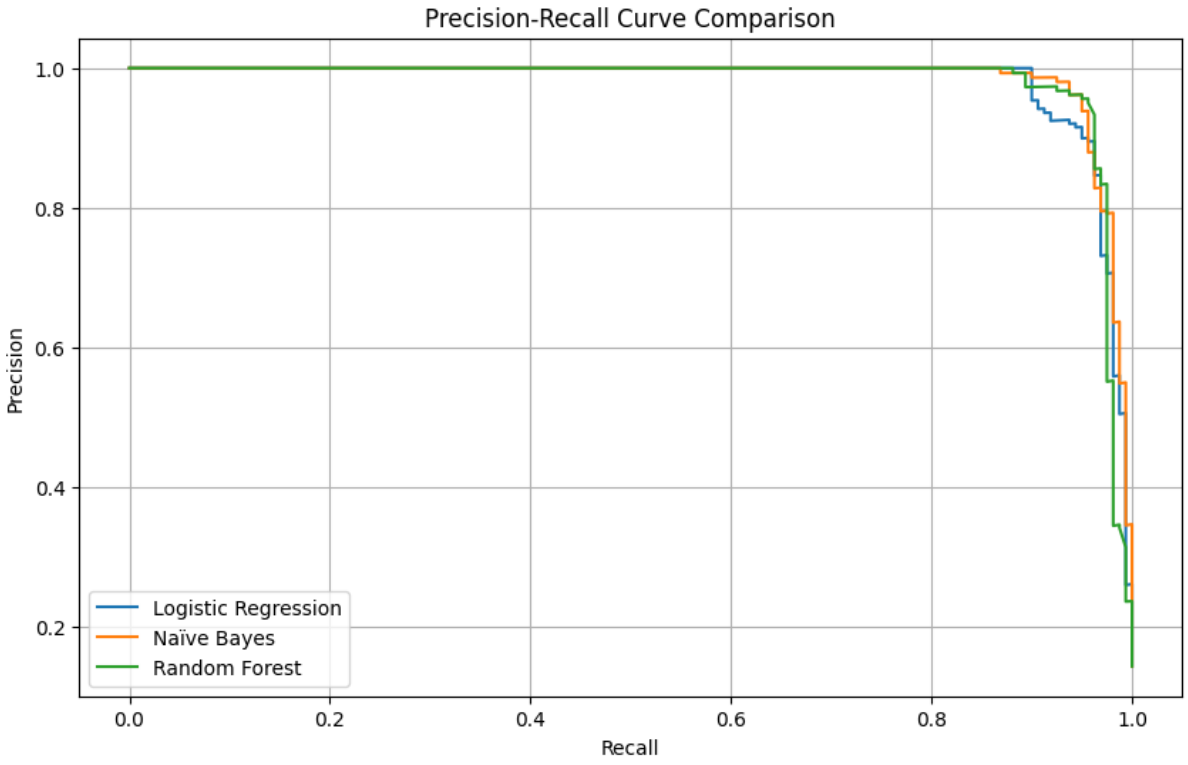
6.ROC Curve

Graph-1



7.Precision-Recall Curve

Graph-2



Logistic Regression Accuracy: 0.96

Random Forest Accuracy: 0.98

Naïve Bayes Accuracy: 0.98

(Graph-1) ROC Curve Analysis:
The ROC curve (Graph-1) displays the classification performance of three different models—Logistic Regression, Naïve Bayes, and Random Forest—on a binary classification task (spam detection). Each model's curve is plotted, with the Random Forest and Logistic Regression models closely hugging the top-left corner, indicating strong true positive rates with low false positive rates. The AUC (Area Under the Curve) values for all models are high, reflecting excellent discriminative ability between spam and ham messages. Overall, the ROC analysis shows that all models are highly effective, with Random Forest slightly outperforming others.

(Graph-2) Precision-Recall Curve Analysis:
The Precision-Recall curves (Graph-2) further evaluate the models, especially focusing on their ability to maintain high precision and recall. This is particularly important for spam detection, where false positives can be costly. All three models demonstrate strong performance, maintaining high precision across a wide range of recall values. Random Forest and Logistic Regression maintain the best balance between precision and recall, indicating robustness even when identifying minority class (spam) examples. These curves confirm that the models are reliable even in scenarios with class imbalance.

V. CONCLUSION

This capstone project successfully demonstrated the application of Python-based data analysis and machine learning techniques across three distinct datasets: text (spam messages), time-series (Google stock prices), and image data (Intel image classification).
The multidisciplinary approach enabled a broad exploration of real-world data challenges and solutions across Natural Language Processing (NLP), Time-Series Forecasting, and Computer Vision domains.

In the Spam Detection dataset, advanced NLP techniques combined with TF-IDF vectorization and machine learning models such as Logistic Regression, Naïve Bayes, and Random Forest were employed to classify messages as 'spam' or 'ham'.
All models achieved high classification accuracies, with ROC and Precision-Recall curves showing strong model performance and minimal misclassification.
This task highlighted the crucial role of effective text classification models in ensuring communication security and user trust.

For the Google Stock Prices dataset, various regression models—including Linear Regression, Random Forest Regressor, and Support Vector Regressor (SVR)—were trained to predict stock closing prices.
Among them, Random Forest Regressor achieved the best R^2 score, indicating strong capability in capturing non-linear financial data patterns.
This analysis underscored the importance of predictive modeling in financial forecasting and market trend analysis.

In the Intel Image Classification dataset, a Convolutional Neural Network (CNN) model was applied to classify landscape images into categories like buildings, forests, glaciers, mountains, seas, and streets.
Despite achieving moderate classification accuracy, the ROC and Precision-Recall curve analyses revealed areas for model improvement, suggesting potential benefits from enhanced preprocessing, class balancing, and deeper CNN architectures.
This task demonstrated the challenges of image recognition and the importance of strong data augmentation and model tuning in computer vision projects.

Through comprehensive data preprocessing, feature engineering, model evaluation, and performance visualization (ROC curves, Precision-Recall curves, Confusion Matrices, and R^2 score analysis), this project showcased the practical ability of machine learning models to handle diverse data types.
It also highlighted the significance of statistical evaluations to interpret model behavior and improve predictive outcomes.

To further enhance these results, future work could involve:

- Implementing advanced ensemble methods like XGBoost or stacking for classification and regression tasks.
- Improving CNN performance by using pre-trained architectures like ResNet, VGG, or EfficientNet with transfer learning.

- Expanding datasets to include more diverse, balanced samples for better model generalization.
- Utilizing cloud-based resources for scalable model training.
- Applying Explainable AI (XAI) techniques to improve model interpretability and transparency.

In conclusion, this capstone project not only met academic objectives but also built a strong foundation for addressing real-world challenges using Python, machine learning, and deep learning.
It reinforced the interdisciplinary nature of data science and its vital role in driving future innovation across diverse industries.