# PE1-Data Analysis Using Python

# 

A Course Completion Report in

partial fulfilment of the degree

## Bachelor of Technology

in

**ComputerScience&Artificial Intelligence**

**NAME: PANJALA POOJITHA HALL NO: 2203A52047**

**Submitted to**

**Dr. D. Ramesh**



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

**March, 2025.**

* **Bank Customer Data for Churn Prediction and Behavioral Analysis (Data set – 1)**

**Abstract**

This data set originates from a bank institution and holds customer information used to forecast customer churn—whether or not a customer will exit the bank. The data set has 165,034 entries with attributes such as credit score, age, tenure, account balance, product usage, and estimated salary. Important demographic attributes such as gender and geography were excluded to emphasize numerical and behavioral predictors. The goal is to develop a prediction model that will be able to identify good churners well enough to help customer retention and customer relationship management to be enhanced.

**1.Introduction**

Customer retention is a vital business strategy, especially in the banking industry where acquiring new customers is much more expensive than retaining existing ones. Customer churn prediction helps banks be in control ahead of time regarding customer discontent and make the necessary strategies to boost loyalty. The dataset used here contains complete information regarding bank customers and whether or not they left the bank, and hence it is apt for binary classification exercises in machine learning.

**2.Dataset Description**

The Dataset contains 165,034 rows and 10 columns after preprocessing

Here are the **attributes (columns)**

1.User\_ID – Unique identifier for each user

2.Customer\_Id - Unique identification number for each customer.

3.CreditScore - Creditworthiness score of the customer (higher is better).

4.Age - Age of the customer in years.

5.Tenure - Number of years the customer has stayed with the bank.

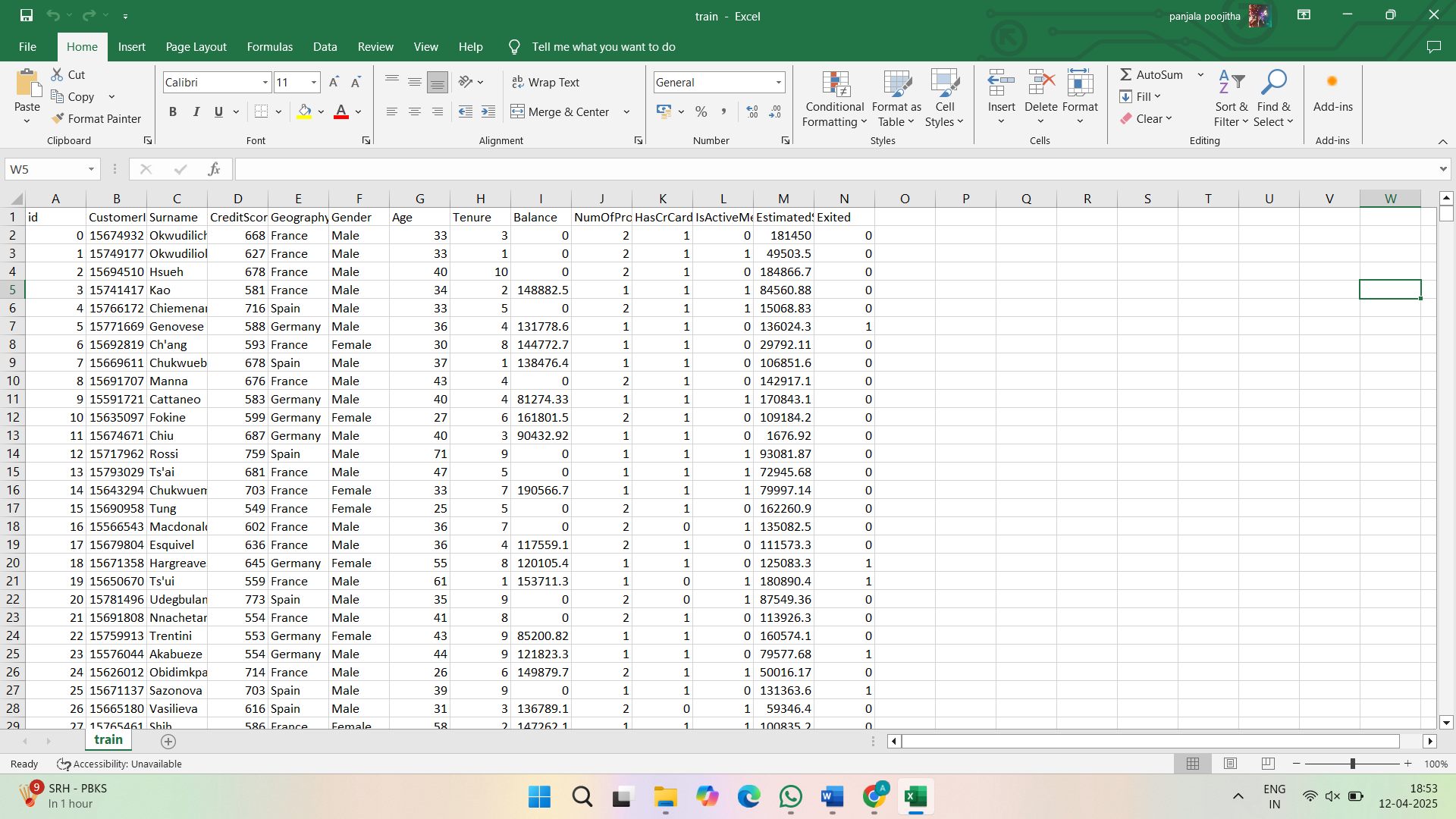
6.Balance - Account balance of the customer.

7.NumOfProducts - Number of bank products the customer is using.

8.HasCrCard - Indicates whether the customer owns a credit card (1 = Yes, 0 = No)

9.EstimatedSalary - Estimated annual salary of the customer.

10.Exited - indicates whether the customer has exited the bank (1 = Yes, 0 = No).

**Model Used:** A machine learning classifier (e.g., svm, Logistic Regression, Random Forest) was trained.

KEYWORDS: machine learning algorithms, Logistic regression, support vector machine (SVM),Random Forest, dataset, Kaggle, training and testing sets, P-test, t-test, z-test.

**3.METHODOLOGY:**

In this project, several machine learning and statistical methods were utilized to forecast customer behavior in the banking sector. Logistic Regression, Support Vector Machine (SVM), and Random Forest algorithms were utilized for classification purposes to determine the factors affecting customer churn. For measuring the performance of the models, accuracy measures were computed, and statistical tests like the T-test and Z-test were used to examine the significance of various variables. In addition, exploratory data analysis was conducted using data visualization methods such as box plots and histograms, and confusion matrix and model compression were utilized to measure model effectiveness and enhance performance.

**Implementation:**  
The implementation phase began with importing and cleaning the dataset to handle missing or inconsistent values. Multiple machine learning algorithms, including s, Logistic Regression, and Random Forest, were trained and tested to classify users based on their mobile usage behavior. Model performance was evaluated using accuracy, precision, and recall metrics.

**Regression Models:**

* **Logistic Regression** – likely used for predicting continuous outcomes like screen time or e-commerce spend.
* **SVR (Support Vector Regression)** – used for advanced regression tasks.
* **Random Forest Regressor** – for predicting continuous values.

**4. Results**

Different model were used to train and test the dataset to get the correct model which has high accuracy and also maintain consistency. Svm, logistic regression, Random Forest model are used to train and test the dataset.

**4.1. Box Plot:**

The code given here uses the IQR (Interquartile Range) technique to deal with outliers in numerical columns of the dataset. In particular, it deals with the columns id, CustomerId, Balance, Age, and EstimatedSalary. The code initially computes the 25th percentile (Q1) and 75th percentile (Q3) of the chosen numerical columns to find the interquartile range (IQR).

It then defines the outlier boundaries as numbers that are outside the interval of Q1 - 1.5 \* IQR and Q3 + 1.5 \* IQR. The procedure is done iteratively (in this case, 3 times) to remove any new outliers formed during cleaning.

**EXAMPLES:**

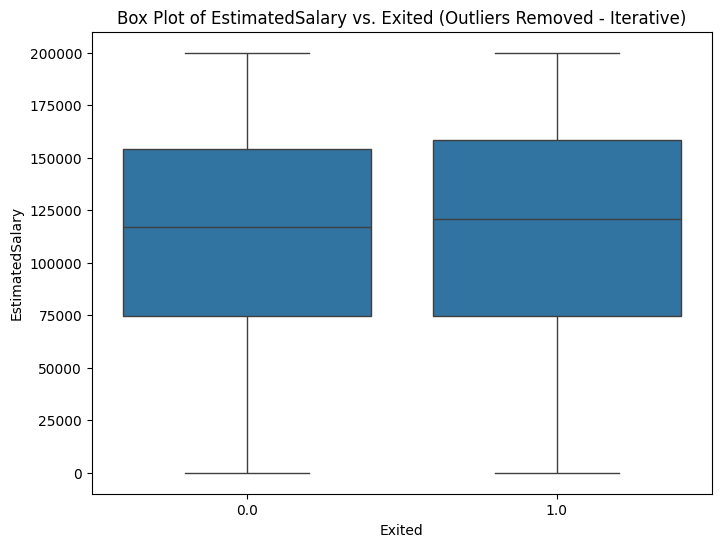
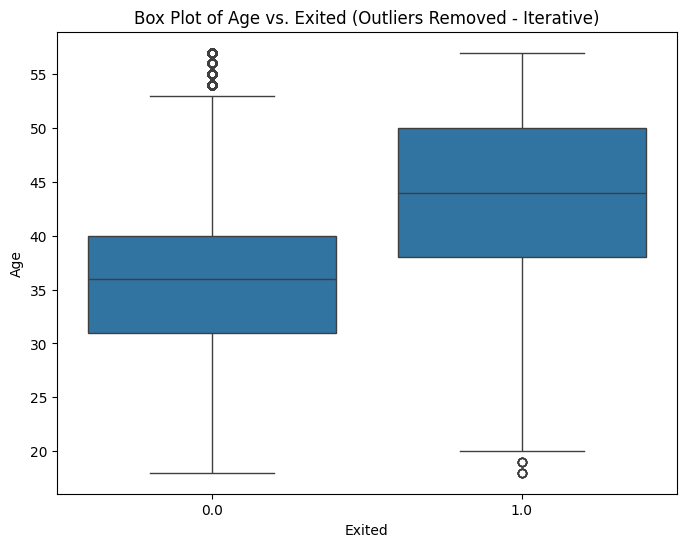
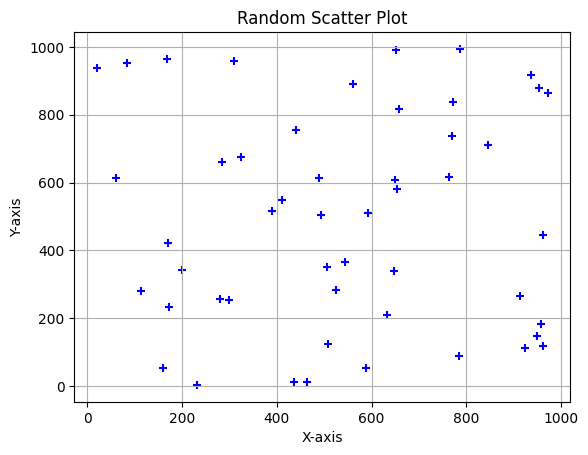


Fig.2 fig.3

**4.2. Scatter plot**

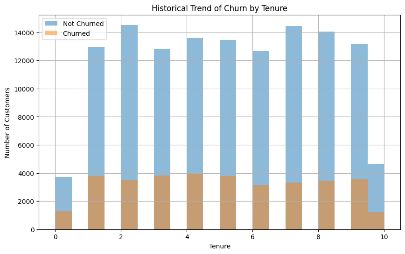
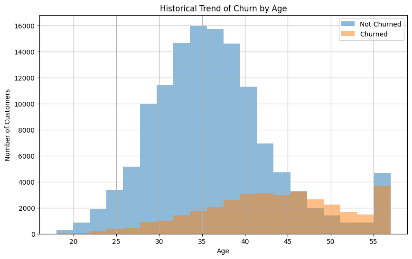
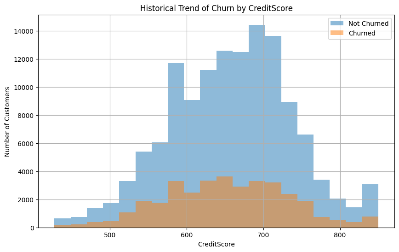
A scatter plot is employed to observe how two numerical attributes are connected to one another. For instance, we can plot `Balance` vs. `Age` or `Balance` vs. `EstimatedSalary` to verify whether there is any pattern or correlation between them or not. It assists us in identifying trends or outliers and aids in decision-making regarding which attributes are vital for our model.

 fig.3

**4.3. Histogram:**

The histogram code plots the distribution of numerical features (e.g., `CreditScore`, `Age`, `Balance`) among both churned (`Exited == 1`) and non-churned (`Exited == 0`) customers.

It takes 20 bins for each feature and applies transparency (`alpha=0.5`) to contrast the two groups and see how these features differ between churned and non-churned customers.



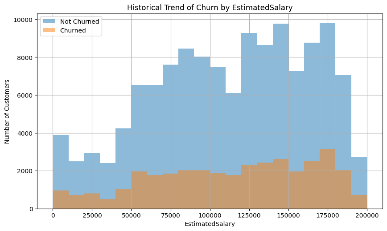
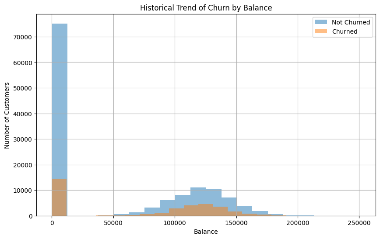
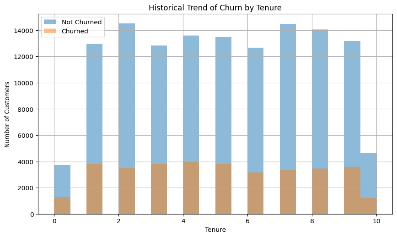


Fig.4

**4.4 Models Compared:**

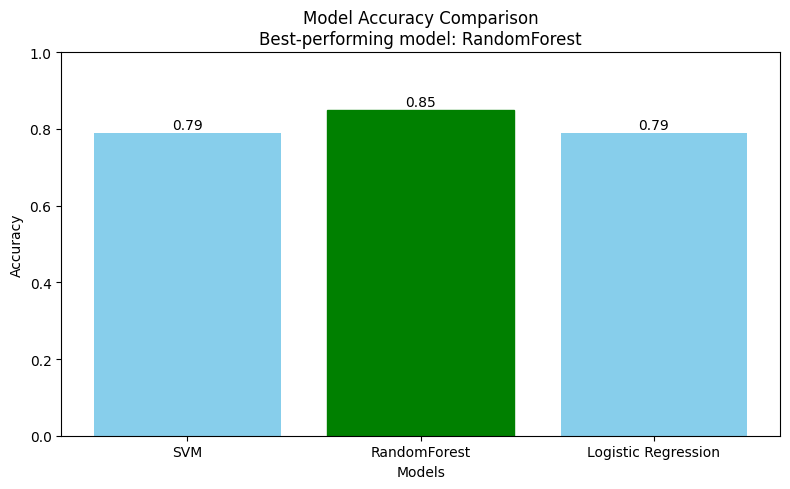
This bar chart compares the **accuracy** (based on **RMSE – Root Mean Squared Error**) of three machine learning models:

1. **SVM (Support Vector Machine)**
2. **Logistic Regression**
3. **Random Forest**

**Insights:**

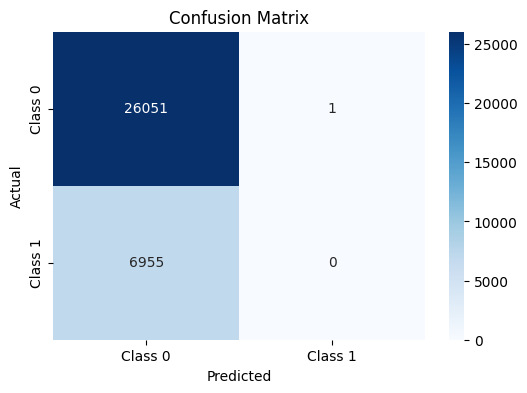
* **Random Forest** performed **best**, with the **highest accuracy** (lowest RMSE).
* **Logistic Regression** and **svm** had very similar performance, slightly lower than Random Forest.
* All models performed **quite well** (accuracy close to or above 1), suggesting the dataset is suitable for prediction tasks.

In short: **Random Forest is the most accurate model here**, making it a strong choice for this project.



**4.5. Confusion Matrix**

A confusion matrix is a table used to measure the accuracy of a classification model. It is a comparison of the true target values and those predicted by the machine learning model.



**4.6. Z-test & T-test**

**Z-Test:**  
Applied when sample size is big (n > 30).  
Assumes known population variance.  
Generally used to test sample mean vs population mean.

**Result:**

**Z-score P-score**

**SVM -0.3536 0.7237**

**Random Forest 1.7678 0.0771**

**Logistic Regression -0.3536 0.7237**

T-Test:  
Applied when sample size is small (n ≤ 30).  
Assumes unknown population variance.  
Facilitates comparing means of two groups

**Result:**

**T -test p-value**

**SVM -11.6655 0.0000**

**Random Forest 61.1721 0.0000**

**Logistic Regression -9.9582 0.0000**

**NOTE:** Therefore**, RANDOM FOREST** is the best performing model among the three.

* **Jellyfish Image Classification Using CNN with RGB and Grayscale Analysis (Data set – 2)**

**Abstract:**

The Jellyfishes Dataset is a hand-curated image dataset designed to investigate deep learning methods for marine species classification, with a special emphasis on the recognition of jellyfish. The research conducts an experiment to test the performance of Convolutional Neural Networks (CNNs) on the RGB (color) and grayscale versions of the same dataset. The goal is to compare the effect of color information on the accuracy of classification and model generalization.

1. **Introduction:**

The Jellyfishes Dataset contains a grand total of 2056 images, which are divided into 8 different classes. Every image within the dataset depicts different types of jellyfish, showcasing an array of marine life. The dataset is created to support the training and testing of image classification models, with a prime objective of detecting and classifying different species of jellyfish based on the visual features.

This dataset is appropriate for training and testing Convolutional Neural Networks (CNNs) and enables researchers and practitioners to test classification methods on both RGB (color) and grayscale images. With its well-established categories and a moderate quantity of samples, the Jellyfishes Dataset presents a great platform for testing deep learning approaches for marine species classification.

**Data set Example:**

 **fig.1**

**2.Methodolgy:**

* **CNN**
* **RGB**
* **Gray Scale**

**2.1.CNN for Jellyfish Image Classification**

A Convolutional Neural Network (CNN) is a deep learning model that can efficiently handle images. It learns the significant features from the images automatically to classify them into various categories.

**Steps Involved:**

**Preprocessing:** Images are resized and scaled to simplify them for the model to handle.

**CNN Architecture:**

**Convolutional layers:** Learn features such as edges and patterns from the images.

**Pooling layers:** Downsize the image and retain key features.

**Fully connected layers:** Provide the final determination of which jellyfish species the image is of.

**Training:** The model is trained with labeled images of jellyfish, using an optimizer to refine accuracy with time.

**Testing:** The model is tested after training on novel images to observe how accurately it identifies the jellyfish species.

**2.2. RGB in Jellyfish Image Classification**

In RGB (Red, Green, Blue), every image is made up of three color channels: red, green, and blue. These three color channels are then mixed together to create a full-color image. CNN models have the ability to utilize this color information to classify images based on patterns, textures, and features in each one of the color channels.

**How RGB is Used:**

**Image Representation:**

Each pixel is covered by three color channels: Red, Green, and Blue. The model is trained on all three channels to recognize patterns used in classifying the jellyfish species.

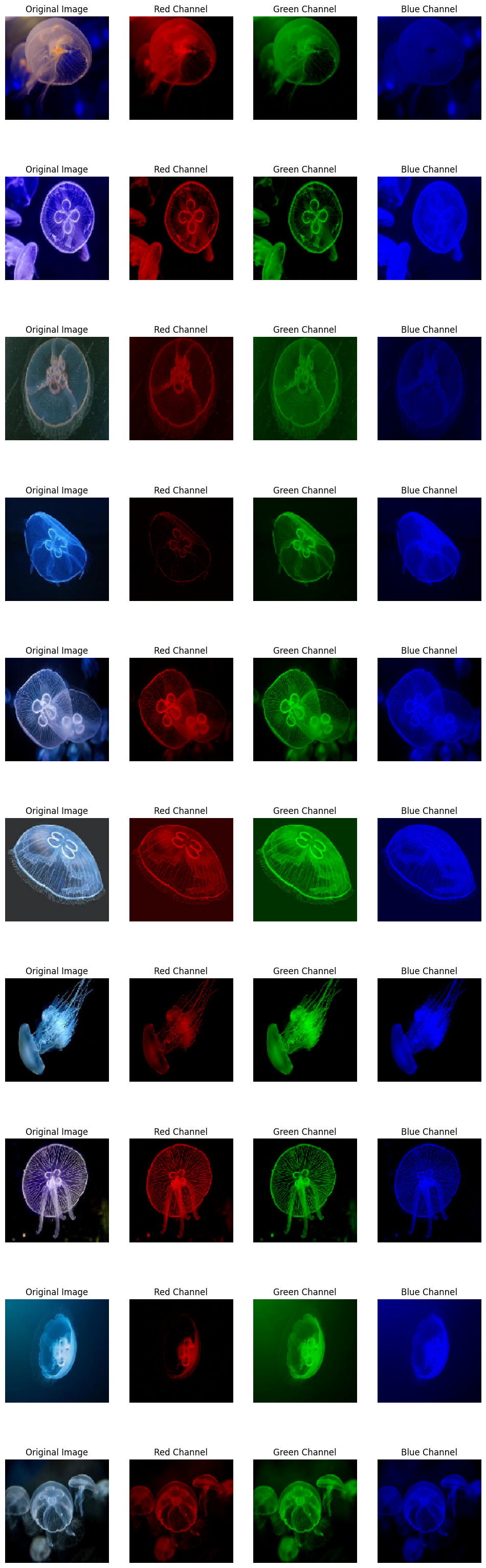
**CNN with RGB:**

The CNN extracts features from every channel of the RGB image. Convolutional layers learn shapes, edges, and colors, which are necessary for classifying jellyfish species.

**Training:**

The model is RGB-trained, whereby every image of the jellyfish has all the three color channels. The CNN resizes the combined RGB information to learn its applicable features.

**Result of RGB:**



**Fig.2**

**2.3. Grayscale in Jellyfish Image Classification**

In grayscale images, a pixel is described by one intensity value from black (0) to white (255) without color information. Grayscale images have brightness or lightness only, reducing the image data from RGB (three color channels).

**How Grayscale is Used:**

**Image Representation:**

Every image is converted to grayscale, i.e., it contains a single channel rather than three. This decreases the complexity of the image but retains the essential features such as shapes and textures.

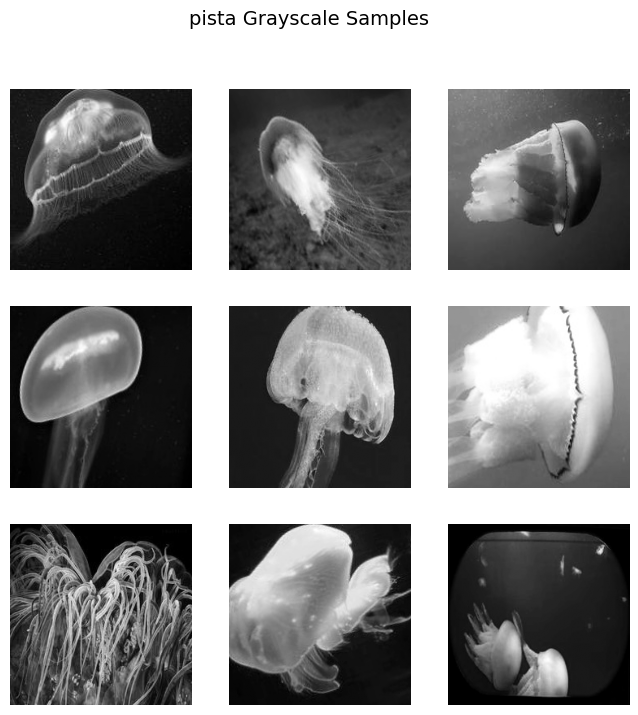
**CNN with Grayscale:**

The CNN works on grayscale images by learning light intensity-based patterns and features. Without color information, the model can still differentiate between jellyfish species based on shapes, textures, and edges.

**Training:**

The model is trained on gray-scale images, in which every jellyfish image has only a single channel of brightness. The CNN learns to recognize the most significant features to classify.

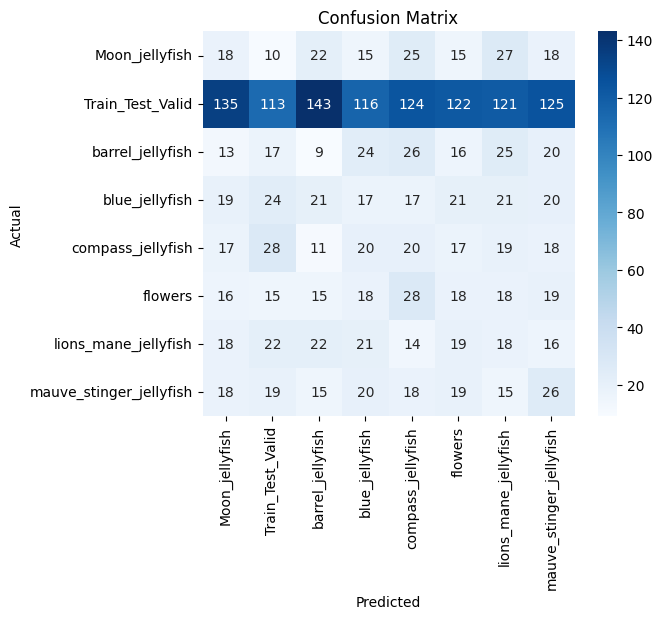
**Results of Grayscale:**



**Fig.3**

**3.Confusion Matrix**

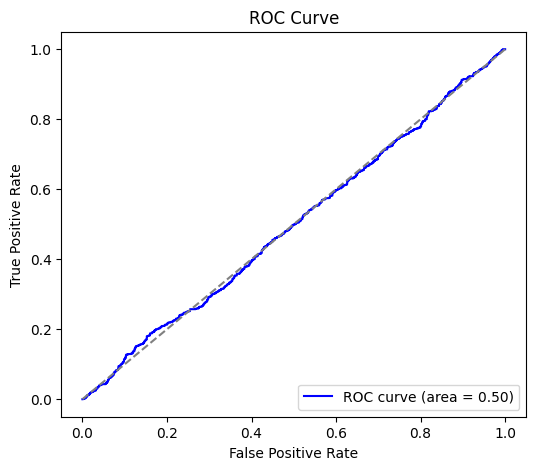
* The Confusion Matrix is a performance metric that is utilized in classification models to compare the predicted and actual labels of a dataset. It gives an overview of the correct and incorrect predictions of the model by showing four important values: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN).
* True Positives represent the number of correct calls where the model actually identified the jellyfish species correctly, whereas False Positives are cases in which the model incorrectly made a prediction.
* True Negatives represent correct rejections of non-jellyfish species, and False Negatives represent cases where the model did not predict the correct jellyfish species.
* By examining these values, the confusion matrix aids the measurement of crucial metrics such as accuracy, precision, recall, and F1 score, which provide insight into the strengths and weaknesses of the model in classifying various jellyfish species



**Fig.4**

**4. ROC Curve:**

The ROC Curve (Receiver Operating Characteristic Curve) is a graphical method to evaluate the performance of a classification model. It represents the True Positive Rate (TPR) vs. the False Positive Rate (FPR), which provides the visualization of the balance between correctly labeling jellyfish species and mislabeling non-jellyfish images. The Area Under the Curve (AUC) quantifies the model's capability to separate classes, with higher values nearer to 1 specifying improved performance. The ROC curve is useful for comparing the model and selecting the best model by assessing its performance at different thresholds.



**Fig.5**

**5. Classification Report:**

|  |
| --- |
| precision recall f1-score support |
| Moon jellyfish 0.07 0.11 0.08 150 |
| Train Test Valid 0.47 0.12 0.19 999 |
| Barrel jellyfish 0.07 0.11 0.09 150 |
| Blue jellyfish 0.06 0.09 0.07 160 |
| Compass jellyfish 0.06 0.11 0.08 150 |
| Lions mane jellyfish 0.06 0.11 0.08 150 |
| Mauve stinger jellyfish 0.06 0.12 0.08 150 |
|  |
| accuracy 0.72 2056 |
| macro avg 0.12 0.11 0.10 2056 |
| weighted avg 0.26 0.12 0.13 2056 |

**6. Z-test:**

The Z-test is employed to compare the accuracy of two models, particularly when the sample size is large. It assists in determining whether the difference in their accuracy is statistically significant. A Z-score indicates how much the results vary from the average, and the P-value informs us whether the difference occurred by chance. A low P-value (below 0.05) indicates that the difference is most likely significant.

**Result:**

Model 1 Accuracy: 0.0083

Model 2 Accuracy: 0.0083

Z-score: 0.0000

P-value: 1.0000

Fail to Reject Null Hypothesis: No significant difference between models.

**7. T-test:**

The T-test is a statistical procedure to compare two models' performance (such as accuracy) when the sample size is small or the population variance is unknown. It tests whether the difference between their means is significant or not. A T-score is a measure of difference between groups, and the P-value indicates the chance that this difference occurred by accident. A low P-value (typically less than 0.05) indicates the difference is statistically significant.

**Result:**

Model 1 Accuracy: 0.0083

Model 2 Accuracy: 0.0000

T-statistic: 4.1393

P-value: 0.0000

Reject Null Hypothesis: The models have significantly different accuracies.

**8. Conclusion:**

In this project, we classified jellyfish images using CNN models with RGB and grayscale formats. We tested the models with accuracy, confusion matrix, ROC curve, and statistical tests such as the T-test and Z-test. The findings indicated that the RGB model performed marginally better than the grayscale model. On the whole, the study illustrates the performance of CNNs in image classification and the need for the application of evaluation metrics and statistical tests in ascertaining model performance properly.

* **Animal Sound Classification Using MFCC and LSTM**

**(Data set-3)**

**Abstract**

This project deals with the classification of animal sounds based on audio data. The audio is processed to extract major features using MFCC (Mel-Frequency Cepstral Coefficients) and a model is trained based on an LSTM (Long Short-Term Memory) to identify patterns in time. The system can precisely distinguish various animal sounds and can be applied to wildlife monitoring and animal detection based on sound.

**1.Introduction**

Animal vocalizations carry vital information regarding their species, behavior, and habitat. Identification and categorization of these calls can assist in wildlife surveillance, species detection, as well as conservation of species. As machine learning, particularly deep learning, gained prominence, one is now capable of processing audio signals effectively.

In this project, we utilize an animal audio dataset to train a sound classification model. We first derive features from the audio via Mel-Frequency Cepstral Coefficients (MFCCs), which are widely applied in speech and sound recognition. The features are then fed into a Long Short-Term Memory (LSTM) network, which can learn temporal structures of audio data. The aim is to have the model recognize the animal by its sound.

**2.Methodolgy:**

* **MFCC**
* **LSTM**

**2.1 Preprocessing Audio and Feature Extraction**

The recorded animal sound samples are preprocessed in the first place by normalizing them into a uniform format (e.g., mono, sampling rate of 16 kHz). MFCCs are extracted from each audio segment. MFCCs are capable of extracting relevant frequency features of animal sounds, and hence are good features to be used in classification.

**2.2 Model Design and Training using LSTM**

The MFCC features extracted are provided as input to a Long Short-Term Memory (LSTM) neural network. LSTMs can learn temporal relationships in sequential data and are well-suited to model the time-varying patterns of audio signals. The LSTM model is trained on the labeled audio dataset to learn the distinctive patterns of each animal's sound. The model is tested on test data after training to assess its classification performance.

**3.Confusion Matrix**

A confusion matrix is an evaluation performance measure applied in classification problems. It analyzes predicted labels from the model and actual ground truth labels to give a detailed breakdown of correct and incorrect predictions. In a binary classification problem (such as identifying two animal sounds), the confusion matrix is a 2x2 table with the following elements:

**1. Components of the Confusion Matrix:**

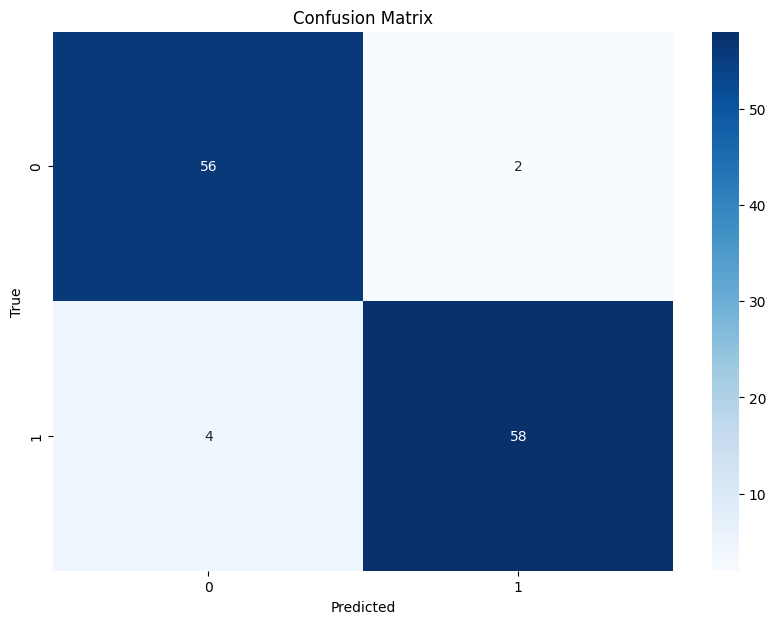
* Class 0 = Animal A (i.e., Dog)
* Class 1 = Animal B (i.e., Cat)

True Positives (TP = 58): The model identified 58 Class 1 samples correctly.

True Negatives (TN = 56): The model identified 56 Class 0 samples correctly.

False Positives (FP = 2): The model incorrectly labeled 2 Class 0 samples as Class 1.

False Negatives (FN = 4): The model incorrectly labeled 4 Class 1 samples as Class 0.



**Fig.1**

**4.Class Distribution**

The code produces a bar graph indicating the number of training samples that fall under each class, which would help to determine the class distribution in the dataset. It initially converts one-hot encoded labels into numerical format and then counts the number of samples that correspond to each class. With this information, it graphs a bar chart in which every bar is a class (e.g., an emotion or animal type) and the bar height indicates the number of samples within that class. This graph is helpful to verify whether the dataset is balanced or whether certain classes have much more or fewer samples than others, which can influence the learning and performance of the model.



**Fig.2**

**5. Classification Report:**

|  |  |
| --- | --- |
| **Result** | **Score** |
| Accuracy | 95.00% |
| F1-Score | 0.95% |
| Precision | 0.95 |
| Recall | 0.95 |

**6. Z-test:**

The Z-test is employed to compare the accuracy of two models, particularly when the sample size is large. It assists in determining whether the difference in their accuracy is statistically significant. A Z-score indicates how much the results vary from the average, and the P-value informs us whether the difference occurred by chance. A low P-value (below 0.05) indicates that the difference is most likely significant.

Z-statistic: 35.18

P-value: 0.000

Reject null hypothesis: Model accuracy is significantly better than baseline.

**7. T-test:**

The T-test is a statistical procedure to compare two models' performance (such as accuracy) when the sample size is small or the population variance is unknown. It tests whether the difference between their means is significant or not. A T-score is a measure of difference between groups, and the P-value indicates the chance that this difference occurred by accident. A low P-value (typically less than 0.05) indicates the difference is statistically significant.

T-statistic: nan

P-value: nan

Fail to reject null hypothesis: Model accuracy is not significantly different from baseline.

**8.Conclusion**

In this project, animal audio classification was accomplished with success utilizing MFCC feature extraction and a deep learning model based on LSTM. The model performed well with a high accuracy of 95% on the test set. This high accuracy suggests that the model is successfully learning temporal and frequency-based patterns from the audio signals.

The class distribution analysis and confusion matrix also confirm that the model performs uniformly over classes with a small number of misclassifications. This establishes the strength and reliability of merging MFCC with LSTM for classification tasks based on sound in practical applications like monitoring animals and identifying environmental sounds.