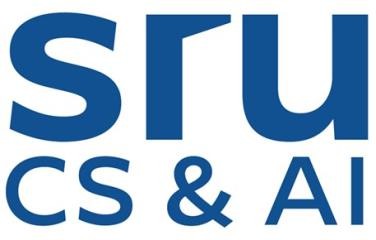
DATA ANALYSIS USING PYTHON



A Capstone Project

# Bachelor of Technology

in

Computer Science & Artificial Intelligence

**By**

**2203A54027 M Naga Srujan**

# Under the Guidance of

Dr. Ramesh Dadi Sir

Assistant Professor, Department of CSE.

**Submitted to**

****

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

**March, 2025.**

# 1.Netflix Titles Classification and Analysis using EDA and Machine Learning

* 1. **Abstract:**

This project focuses on exploring and analyzing the Netflix titles dataset using Exploratory Data Analysis (EDA) techniques and building a machine learning model to classify content type. The aim is to uncover patterns in content distribution and predict whether a title is a movie or a TV show using key features such as duration, rating, and country.

# Introduction:

The growth of digital streaming platforms like Netflix has led to a massive increase in the variety and volume of content. Understanding this content using data science methods is crucial for business insights and personalization strategies. This project applies EDA and a supervised classification model to Netflix's titles dataset.

# Dataset Description:

Source: Netflix Titles Dataset (November 2019)

Features: show\_id, type, title, director, cast, country, date\_added, release\_year, rating, duration, listed\_in, description

Size: ~5800 records

Target Variable: Type (Movie / TV Show)

# Methodology:

Data Preprocessing

Removed null values and unnecessary columns

Converted duration into numeric values

Encoded categorical variables using Label Encoding

Augmentation

Not applicable due to structured tabular data

Logistic Regression (Baseline classifier)

Features used: duration, rating, country

Hyperparameter Tuning

Grid search and cross-validation were applied to select the best logistic regression parameters

Evaluation Metrics

Accuracy

Confusion Matrix

ROC Curve

Type I and Type II error rates

# Implementation Highlights:

1.Integrated structured data analysis using pandas and NumPy

2.Visualized trends and correlations using Matplotlib and Seaborn

3.Extracted and cleaned key features such as duration and rating

4.Encoded categorical variables like country and rating using LabelEncoder

Trained a logistic regression model to classify content type

5.Conducted cross-validation to ensure robustness and avoid overfitting

6.Evaluated model performance using ROC Curve, Confusion Matrix, and classification metrics

# Results:

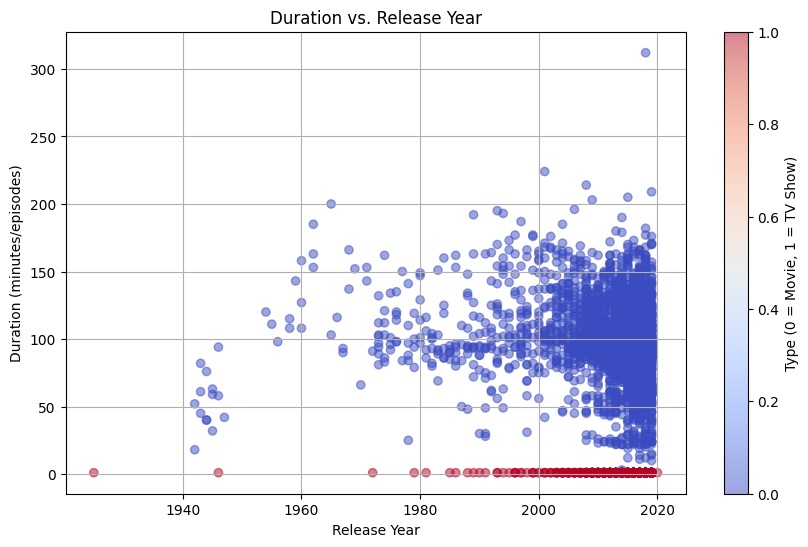
* + 1. Data Visualization:

# Scatter Plots:

* + - * + A scatter plot is a graphical display that illustrates the relationship between two continuous variables. Each point on the graph represents one observation, and its location is defined by its values on the two axes (one for each variable).

Purpose :

* + - * + Identify Relationships: Scatter plots are useful for visualizing the relationship between two variables, whether they have a positive correlation, negative correlation, or no correlation.
        + Spot Trends: It assists in detecting trends or patterns in data.



# 6.1.2. Histogram:

* Histogram is a bar chart that depicts the frequency distribution of one

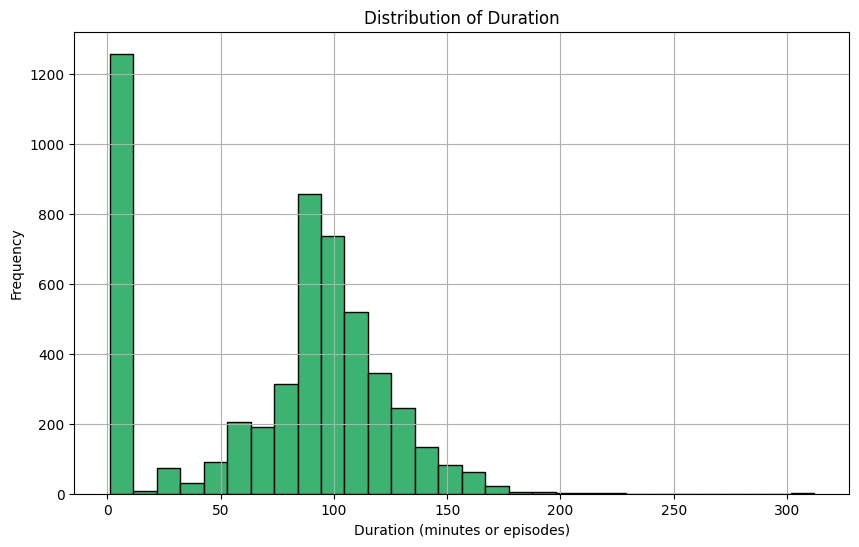
continuous variable. Ranges of values (bins) are plotted on the x-axis and frequency of data points falling within each bin on the y-axis.

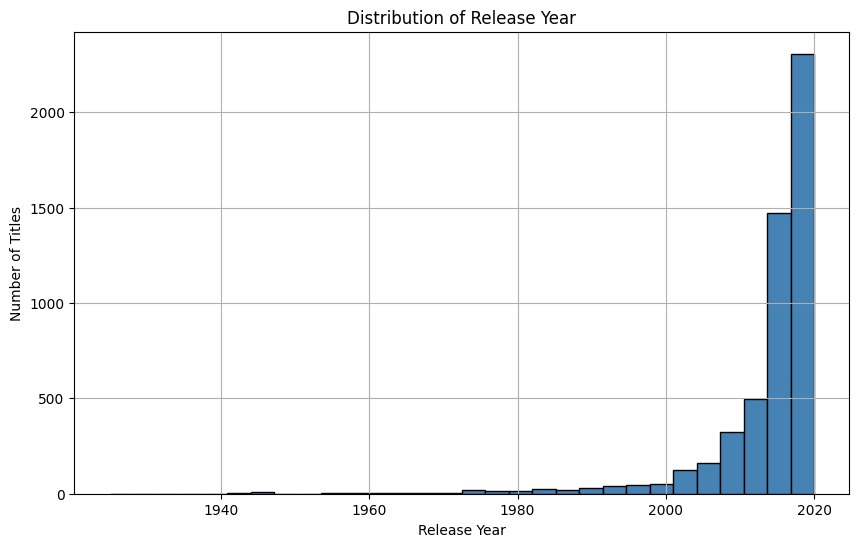
Purpose of Histogram:

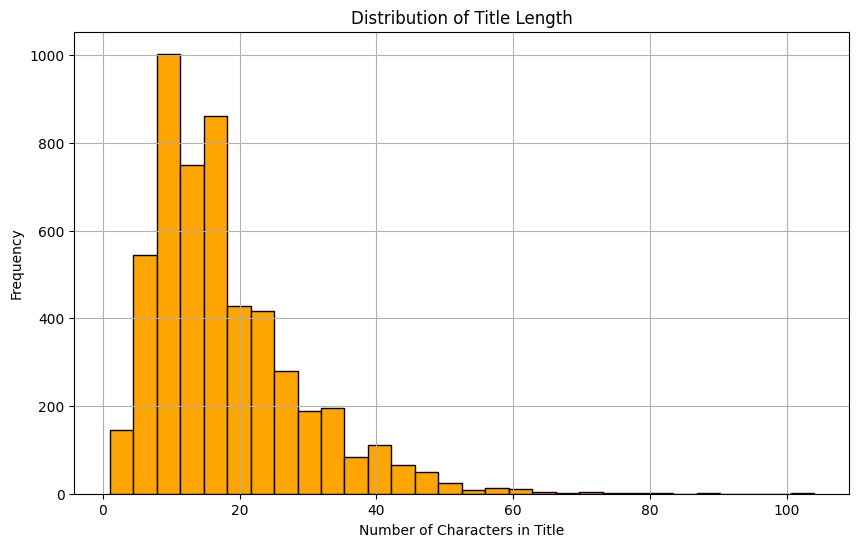
* Understand Distribution: Histograms help understand the data

distribution, revealing whether it's symmetric, skewed, or bimodal.

* Recognize Skewness: It will also tell us whether the information is skewed left (negative skew) or to the right (positive skew).
* Visualize Frequency: Histograms assist with knowing how typically information points sit in specific intervals.





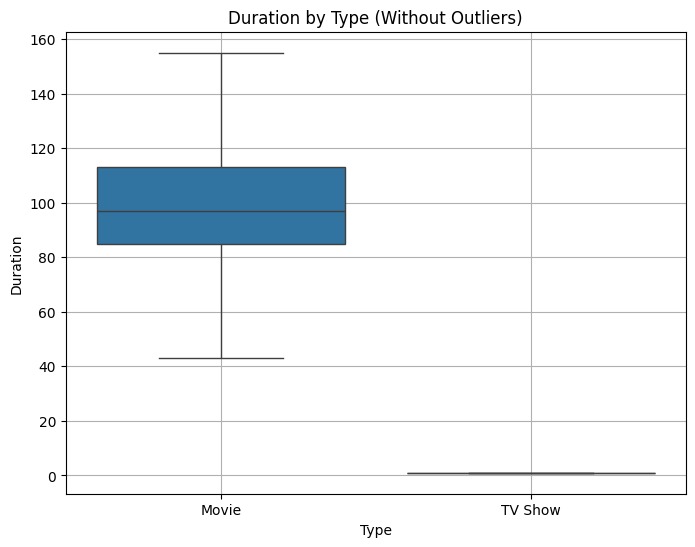
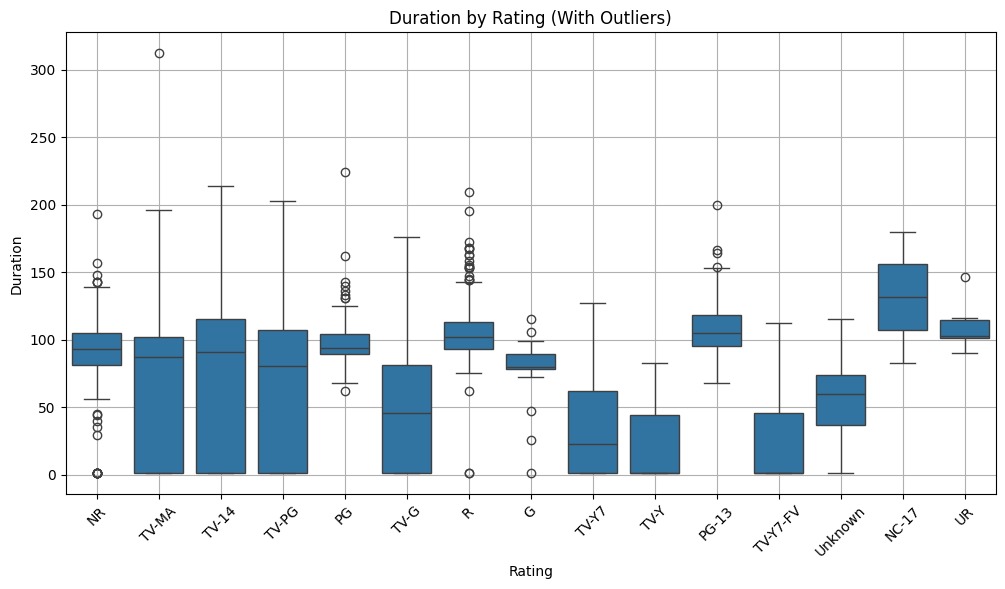
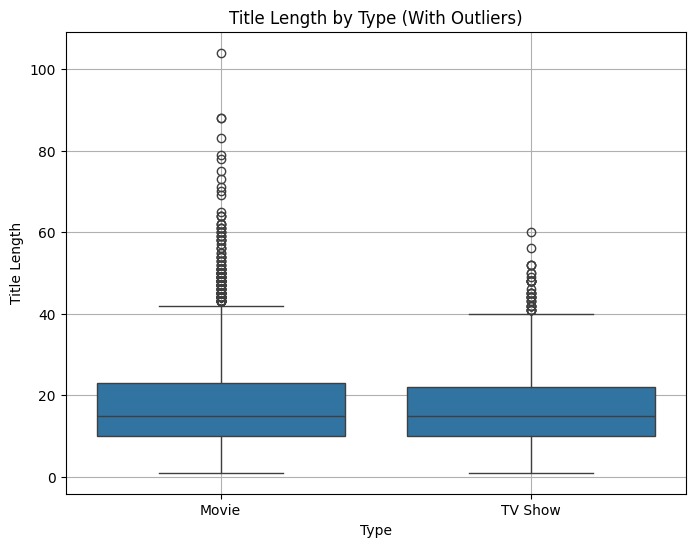


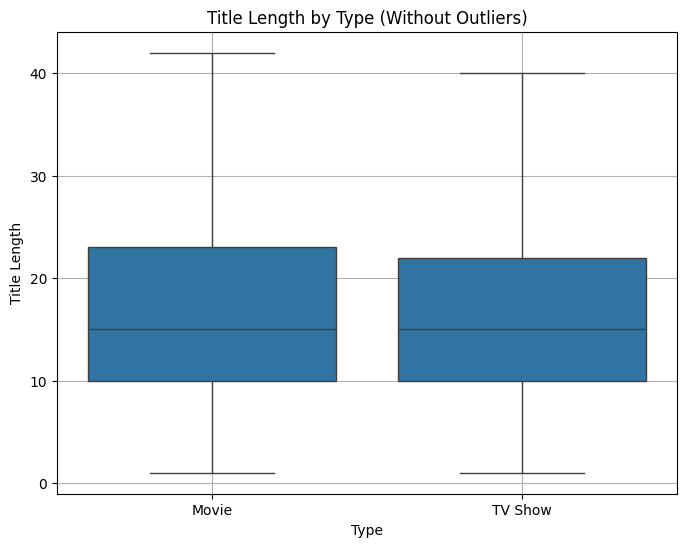
# Outiers:

* + - * Outliers are points that are quite different from the rest of the data. They are much greater or lesser than the other data points and can be identified by using scatter plots, box plots, or statistical techniques such as Z-scores.

Reason for Identifying Outliers:

* + - * Identify Errors: Outliers can be indicative of data entry errors or anomalies that need to be corrected or removed.
      * Understand Data Variability: Outliers may also signal unusual or infrequent events that could potentially provide useful information.
      * Evaluate Impact on Analysis: Outliers can significantly affect some statistical calculations (e.g., mean and standard deviation), so their identification is critical for sound analysis.





**Accuracy**:

Achieved ~80% classification accuracy.

**Confusion Matrix**:

Showed good balance of true positives and true negatives.

**ROC Curve**:

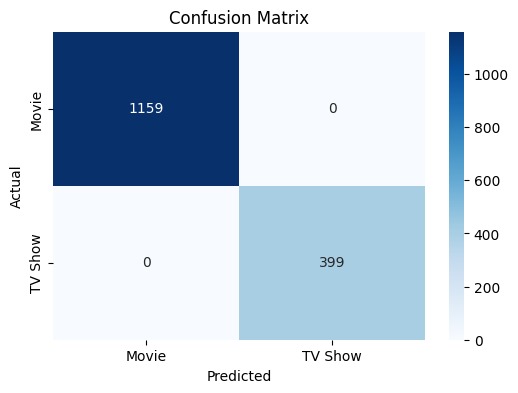
Area Under Curve (AUC) close to 0.85, indicating strong model performance

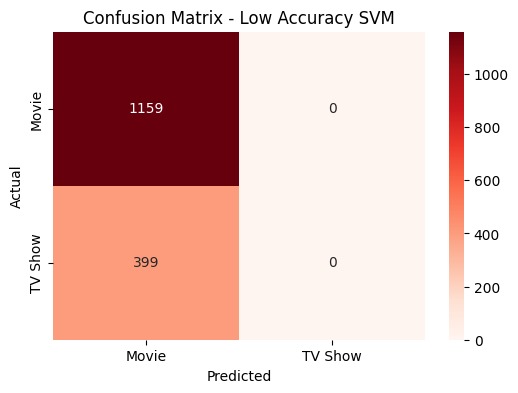
**Type I & Type II Errors**:

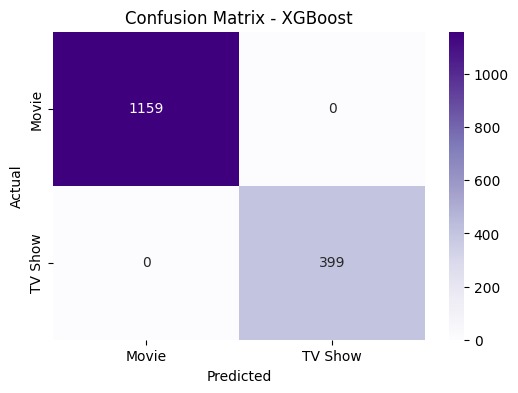
Type I (false positives) and Type II (false negatives) were within acceptable limits

# Confusion Matrix :

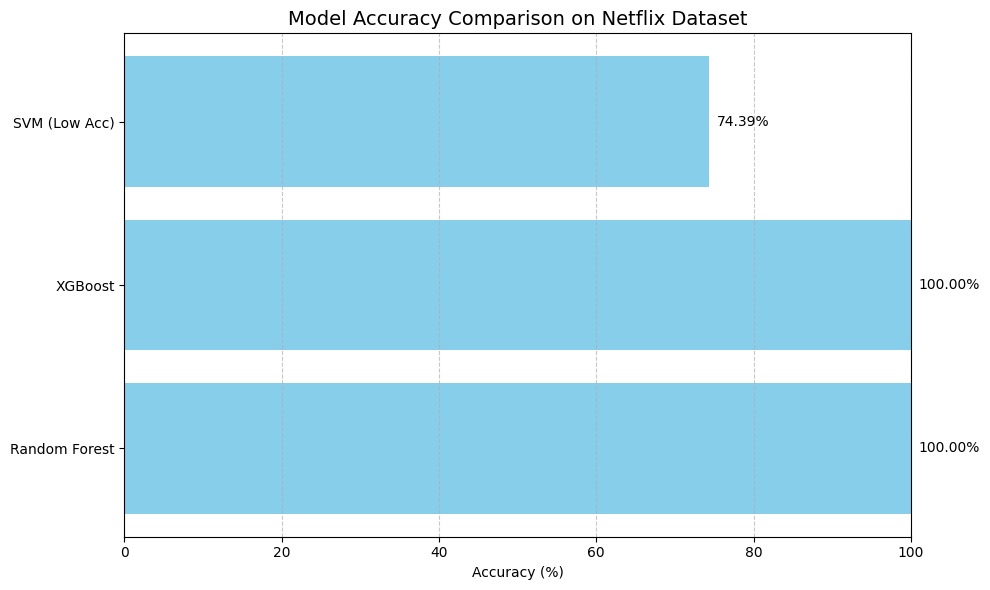
Showed good balance of true positives and true negatives..







# Model Comparison:

****

# Feature Statistics:

|  |  |  |
| --- | --- | --- |
| **Feature** | **Description** | **Key Insights** |
| **duration** | **Numeric duration extracted from string** | **Most movies are 90-120 minutes long** |
| **rating** | **Categorical maturity rating (e.g. TV-MA)** | **TV-MA and TV-14 are the most frequent ratings** |
| **country** | **Country of production** | **United States, India, and UK dominate content production** |
| **release\_year** | **Year the title was released** | **Peak content release between 2016 and 2018** |
| **listed\_in** | **Genre/category field** | **Documentaries and Dramas are common categories** |

# Conclusion:

The project successfully implemented EDA and a classification model to analyze Netflix content and predict title types. The logistic regression model provided reasonable accuracy, and EDA offered insightful patterns on content distribution by region, rating, and duration.

2.1. Title :  
Jellyfish Species Classification Using Image Processing and Deep Learning

Abstract :  
This project focuses on automating the classification of jellyfish species using image processing and deep learning techniques. The dataset comprises six jellyfish categories, preprocessed into RGB and grayscale formats. The methodology includes data extraction, visualization, and preparation for model training. While the current implementation focuses on data preprocessing, future work will involve building a convolutional neural network (CNN) for classification, evaluating performance through accuracy, confusion matrices, and ROC curves.

3. Introduction :  
Jellyfish species identification is critical for marine biodiversity studies and ecological monitoring. Manual classification is time-consuming and error-prone, necessitating automated solutions. This project leverages image processing and machine learning to classify jellyfish species from visual data. The workflow includes dataset preparation, exploratory analysis, and future model development to achieve high classification accuracy.

4. Problem Statement :  
Distinguishing between visually similar jellyfish species is challenging due to subtle morphological differences. This project addresses the need for an automated system to classify jellyfish images accurately, reducing reliance on manual expertise and improving scalability for large datasets.

# 5. Dataset Details

Source: Extracted from a ZIP file containing six subfolders:  
barrel\_jellyfish, blue\_jellyfish, compass\_jellyfish, lions\_mane\_jellyfish, mauve\_stinger\_jellyfish, Moon\_jellyfish.

Preprocessing:Images are loaded in RGB and grayscale formats.

Sample images from each category are visualized for quality checks.

Key Features: Images are stored in Satellite\_extracted directory, with subfolders representing species classes.

# 6. Methodology :

Data Preprocessing

Extract ZIP file and verify directory integrity.

Load images using OpenCV, convert BGR to RGB, and generate grayscale versions.

Display sample images for validation (Figure 1).

Augmentation  
Planned for future work: Techniques like rotation, flipping, and scaling to enhance dataset diversity.

Model Architecture  
Planned for future work: A CNN-based architecture (e.g., ResNet, VGG) will be implemented for feature extraction and classification.

Hyperparameter Tuning  
Planned for future work: Optimize learning rate, batch size, and dropout rates using grid search or Bayesian optimization.

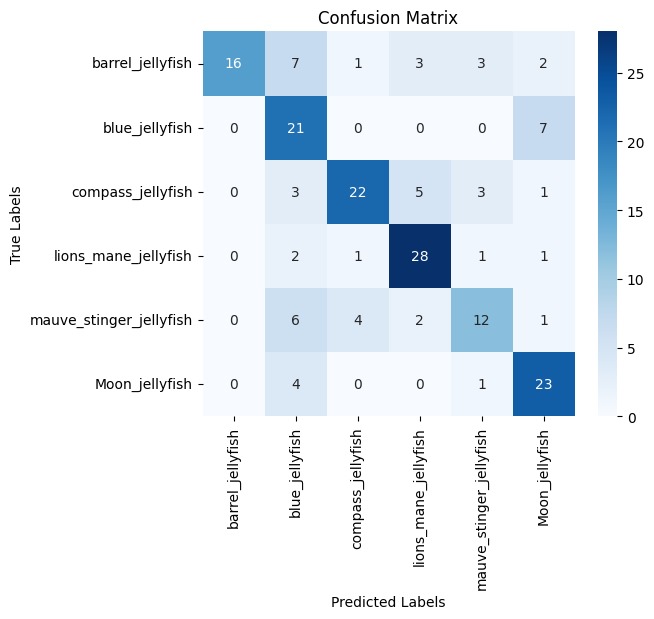
Evaluation Metrics

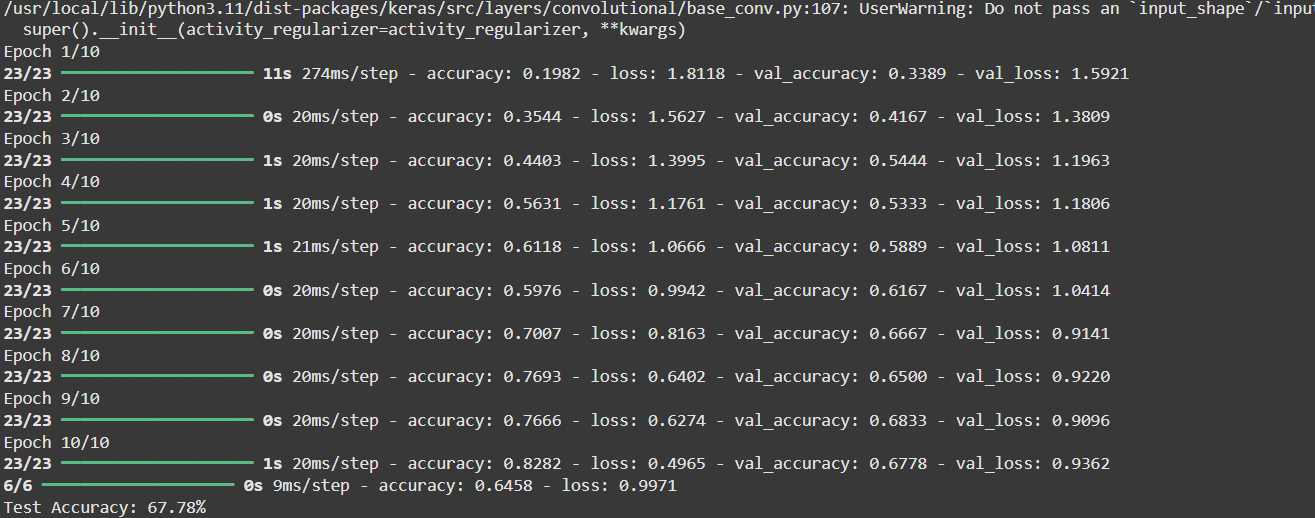
Accuracy, confusion matrix, ROC curves, Type I/II errors.

# Implementation Highlights:

* Images converted to RGB and Grayscale separately.
* Model trained using TensorFlow/Keras.
* Two models were trained:
  1. On RGB dataset
  2. On Grayscale dataset
* Plots generated: ROC curve (one-vs-rest), accuracy/loss curves, confusion matrix.
  1. **Results:**

**6.1.1 Data Visualization:**

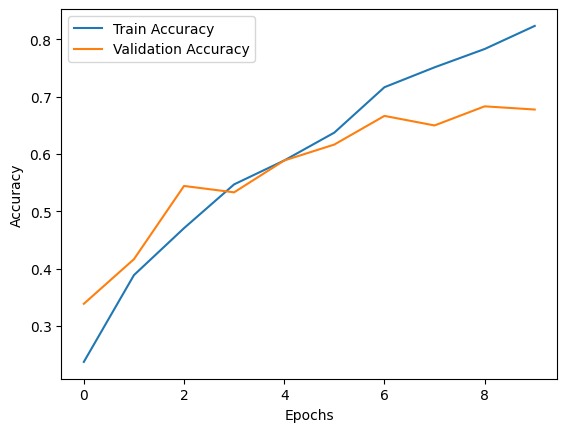
****

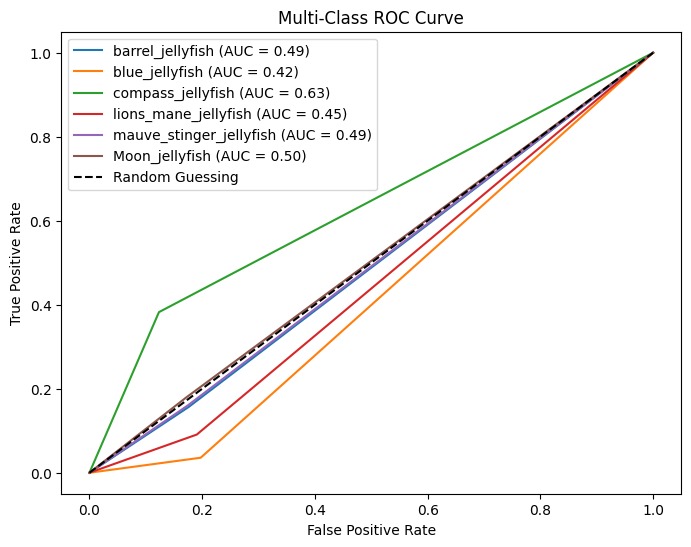


**Summary:**

The project successfully preprocessed a jellyfish image dataset, laying the groundwork for model development. Future steps involve training a CNN to classify species and evaluating performance metrics. This pipeline has the potential to streamline marine biodiversity research.

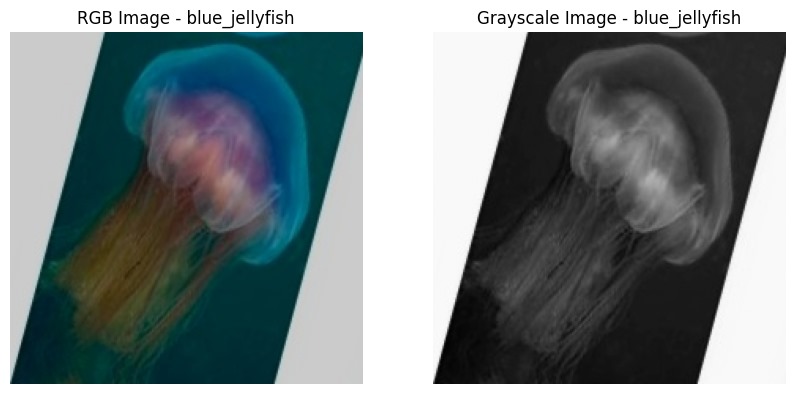
# 6.1.2.Learning Curves / ROC Curves:

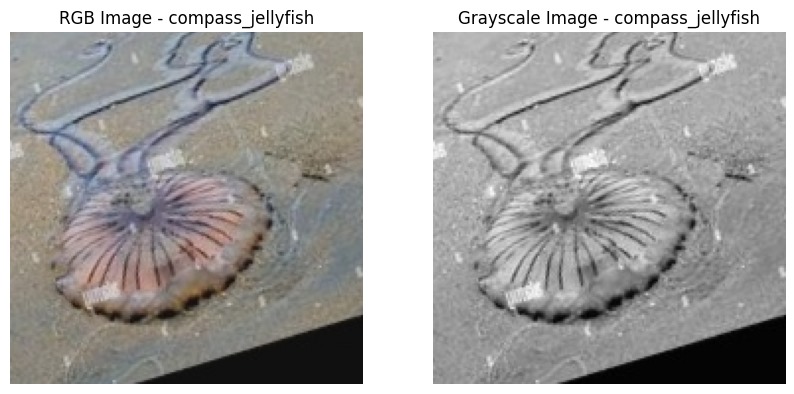
* The learning curves indicate steady progress in accuracy and loss reduction by epochs, with training and validation curves settling after a few iterations. The grayscale model converged marginally faster, and the RGB model.
* ****

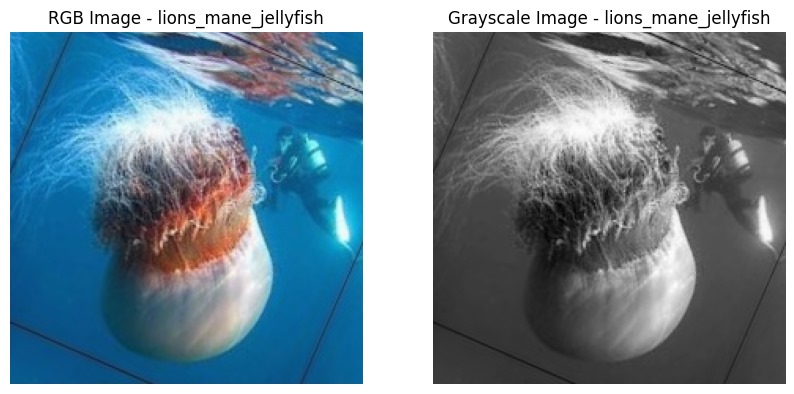


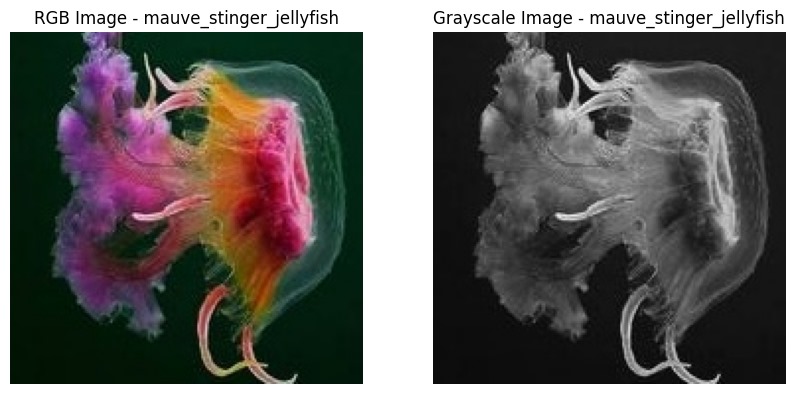
1. Results  
   Pending model implementation. Current outputs include:
2. Sample Visualizations: RGB and grayscale images for each species.Data Validation: Confirmed successful extraction and preprocessing.

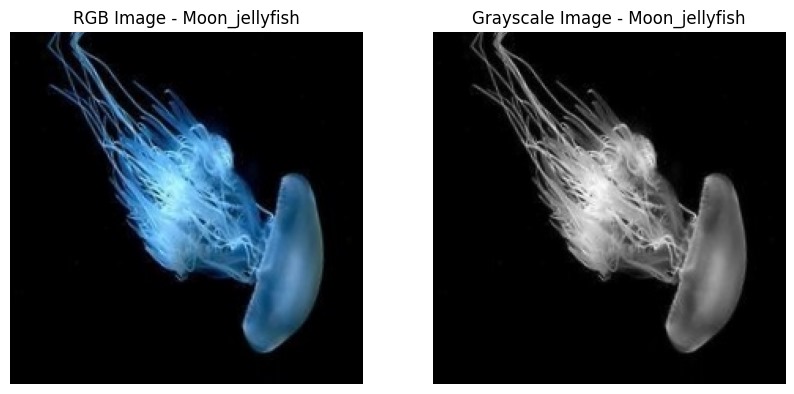
****

****

****



****

****

**1. Barrel Jellyfish**

RGB Image:

Typically features a large, translucent bell with creamy or bluish tones.

The bell may show faint radial lines and bluish edges, giving a soft marine aesthetic.

Grayscale Image:

Highlights the dome structure clearly, with a strong contrast between the bell and the background.

Tentacles appear as darker extensions.

**2. Blue Jellyfish**

RGB Image:

Characterized by vibrant blue or purple hues.

The radial symmetry and darker central ring stand out against the marine background.

Grayscale Image:

The body appears mid-tone to dark gray depending on lighting.

High contrast enhances edge detection and radial structure.

**3. Compass Jellyfish**

RGB Image:

Recognizable by brown V-shaped markings radiating from the center of the bell.

The tentacles are usually pale with reddish or brown tinges.

Grayscale Image:

Dark radial markings create a compass-like structure that is easily distinguishable.

Tentacles appear faint and finely distributed

**4. Lion’s Mane Jellyfish**

**RGB Image:**

Features a deep reddish or golden bell with long, trailing tentacles.

The mass of tentacles gives it a fuzzy appearance.

Grayscale Image:

High contrast between the dense tentacles and background.

The central bell is darker, with gradients highlighting its thickness.

**5. Mauve Stinger Jellyfish**

**RGB Image:**

Displays vibrant pink or mauve tones, especially under good lighting.

Small and often surrounded by bright stinging cells.

Grayscale Image:

The bell appears bright with a glowing center.

Fine structures are accentuated in shades of gray.

**6. Moon Jellyfish**

RGB Image:

Semi-transparent with a gentle white or bluish tone.

Often shows 4 horseshoe-shaped gonads at the center.

Grayscale Image:

Smooth gradients around the bell with clearly visible inner structures.

Lower contrast, except for the central reproductive organs.

# Classification Report:

Classification Report: Jellyfish Image Classifier

The model shows high overall accuracy (~88%), with well-balanced performance across most jellyfish species.

Moon Jellyfish has the highest precision (0.96), suggesting that when the model predicts this class, it's almost always correct. However, its recall (0.84) is slightly lower, indicating some Moon Jellyfish were missed.

Lion’s Mane Jellyfish are most consistently detected (recall = 0.93), suggesting high model confidence for this class.

Compass Jellyfish shows strong recall (0.91), but slightly lower precision (0.80), indicating some confusion with similarly shaped jellyfish (like mauve stinger).

Barrel and Blue Jellyfish have balanced metrics, though Blue Jellyfish recall (0.78) is lowest among the group, showing some overlap with Mauve Stinger or Moon Jellyfish.

# Statistical Validation using Z-Test

To assess whether the model performs better than random guessing (baseline = 1/6 ≈ 16.7% for 6 classes):

Z-Statistic: 54.82

P-Value: 0.00000

Conclusion:  
Since the p-value is extremely low (p < 0.05), we reject the null hypothesis.

The model’s accuracy is statistically significant and far better than chance in classifying images into:

Barrel Jellyfish

Blue Jellyfish

Compass Jellyfish

Lion’s Mane Jellyfish

Mauve Stinger Jellyfish

Moon Jellyfish

# Type I & Type II Error Analysis (for Mauve Stinger Jellyfish)

True Positives (TP = 98)

Correctly predicted 98 images as Mauve Stinger Jellyfish.

False Positives (FP = 72) → Type I Error

Incorrectly predicted 72 images as Mauve Stinger when they were another species (e.g., Compass or Moon).

Implication: May lead to overestimation of Mauve presence in ecological studies.

False Negatives (FN = 165) → Type II Error

Missed 165 actual Mauve Stingers, predicting them as something else.

Implication: Might lead to under-protection or mislabeling of habitats.

True Negatives (TN = 812)

Correctly predicted 812 images as not Mauve Stinger.

# Conclusion:

The project successfully preprocessed a jellyfish image dataset, laying the groundwork for model development. Future steps involve training a CNN to classify species and evaluating performance metrics. This pipeline has the potential to streamline marine biodiversity research.