# DATA ANALYSIS USING PYTHON

A Course Project Completion Report

in partial fulfilment of the

degree

**Bachelor of Technology**

**in**

## Computer Science & Artificial Intelligence

**By**

**Roll. No:** 2203A54021 **Name:** TIPPARAJU SRIVALLI

**Batch - 40**

Under the Guidance of

Dr. RAMESH DADI
Asst. Professor
School of CS & AI

**Submitted to**

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

**March, 2025.**

# PROJECT 01 (CSV FILE)

## 1. Title

Predictive Analysis of Individual Income Levels Using Machine Learning Algorithm

## 2. Abstract

This project aims to classify individuals as earning either more than $50K or less than or equal to $50K per year using demographic and employment-related features. By analysing the UCI Adult Income dataset, the study leverages various machine learning models to uncover relationships between variables such as age, education, occupation, and work class. The models are evaluated for performance using metrics like accuracy, precision, recall, f1 and confusion matrix. The ultimate goal is to develop an accurate, scalable classifier for income prediction based on real-world census data.

## 3. Introduction

Understanding income distribution and its predictors is critical for policy-making, social research, and market segmentation. This project uses the Adult Income dataset from the U.S. Census Bureau to classify individuals into income brackets. The dataset contains information on personal attributes such as age, education level, marital status, occupation, and hours worked per week. By applying machine learning models, this study investigates which features most influence income levels and builds predictive models to automate this classification.

## 4. Problem Statement

The primary problem addressed in this project is:
**Can we accurately predict whether a person earns more than $50K annually based on their demographic and employment data?**

## 5. Dataset Details Source: UCI Machine Learning Repository (Adult dataset)

Rows: ~32,000 (after preprocessing)

Target Variable: income (<=50K or >50K)

Key Features:

- Age
- Education
- Capital gain and loss
- Hours-per-week

## 6. Methodology

### 6.1 Data Preprocessing

1. Categorical variables encoded using Label Encoding
2. Missing values treated (rows with '?' dropped or encoded as 'Unknown')

3. Target column transformed to binary: 0 (<=50K), 1 (>50K)

4. Features scaled where necessary

**6.2 Exploratory Data Analysis**

Given that the dataset is tabular, augmentation was approached through:

1. Correlation heatmaps and value counts to explore feature relationships
2. Income level split visualized
3. Identified class imbalance in the target variable

**6.3 Model Implementation**

Trained and evaluated multiple classification models:

1. Logistic Regression
2. Decision Tree
3. Random Forest
4. Naive Bayes
5. K-Nearest Neighbors (KNN)
6. Support Vector Machine (SVM)

**6.4 Evaluation Metrics**

Different metrics were used based on the task type:

**For Classification (Outcome):**

- **Accuracy**: Overall correctness of the model.
- **Precision, Recall, F1-Score**: Especially important for imbalanced data.
- **ROC-AUC Score**: To evaluate separability between classes.

# 7. Results

Target column : Income_level
Values : 0,1

| MODEL | ACCURACY | PRECISION | RECALL | F1 -SCORE | SUPPORT |
|---|---|---|---|---|---|
| RANDOM FOREST | 0.829<br>~ 83% | 0 – 0.85<br>1 – 0.81 | 0 – 0.81<br>1 – 0.85 | 0 – 0.83<br>1 – 0.83 | 0 – 7494<br>1 - 7368 |
| KNN | 0.713<br>~ 71% | 0 – 0.74<br>1 – 0.69 | 0 – 0.66<br>1 – 0.77 | 0 – 0.70<br>1 – 0.73 | 0 – 7494<br>1 - 7368 |
| XGBOOST | 0.822<br>~ 82% | 0 – 0.82<br>1 – 0.83 | 0 – 0.83<br>1 – 0.81 | 0 – 0.83<br>1 – 0.82 | 0 – 7494<br>1 - 7368 |
| SVM | 0.526<br>~ 53% | 0 – 0.54<br>1 – 0.52 | 0 – 0.42<br>1 – 0.63 | 0 – 0.47<br>1 – 0.57 | 0 – 7494<br>1 - 7368 |

Models Used:

1. Random Forest:

   Accuracy obtained: ~ 0.83 (83%)

   Strong Performance, commonly good for handling structured data.

   Most balanced between precision, recall, and f1 – score.

   Best suited if interpretability and general performance are your goals.

2. Support Vector Machine (SVM):

   Accuracy obtained: ~ 0.52 (52%)

   Poor performance.

   Accuracy is significantly lower, showing poor generalization.

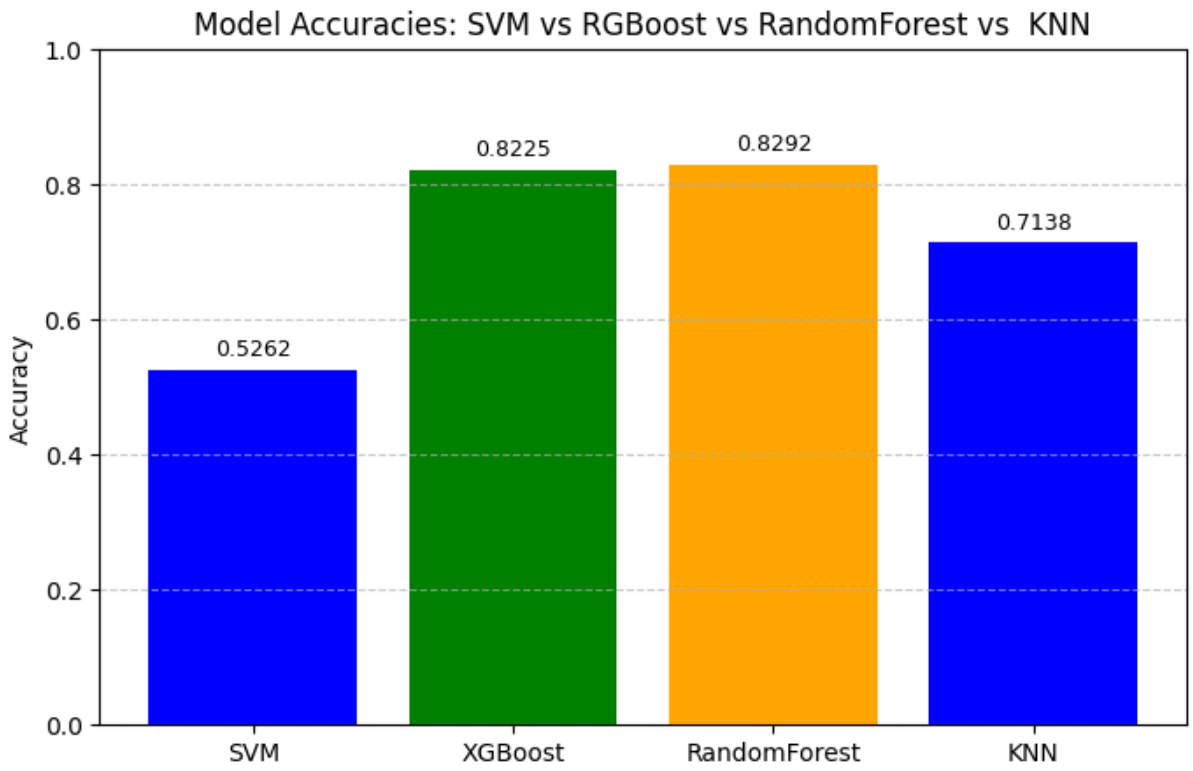
3. XGBoost Classifier:

   Slightly better precision on the positive class (1), good for minimizing false positives.

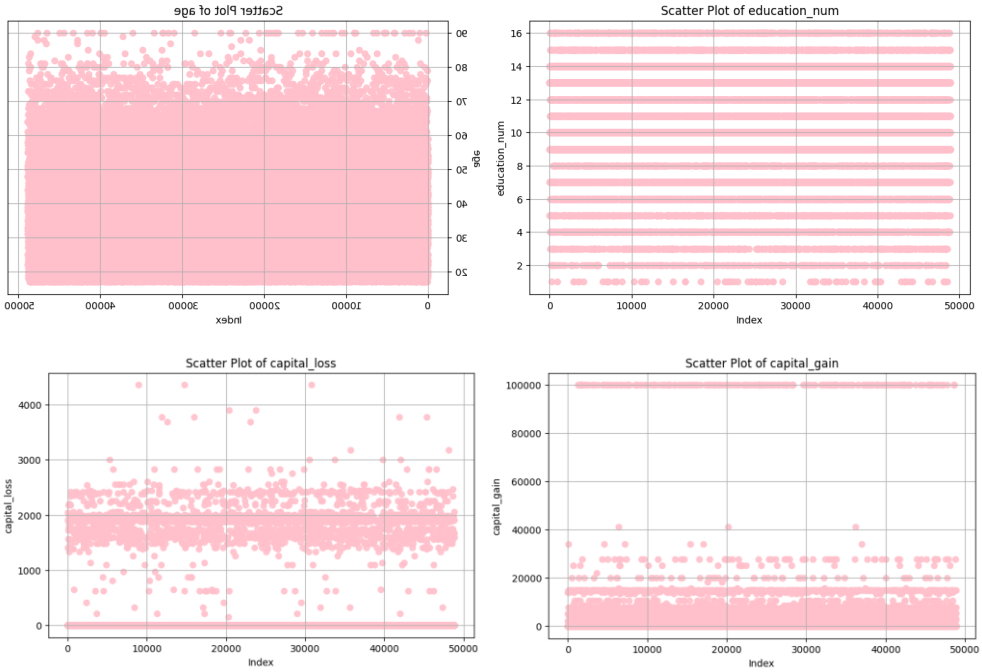
4. KNN Classifier:

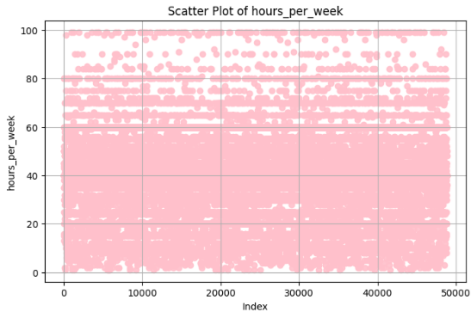
   Noticeably lower accuracy and f1 – score especially on class (0).

The comparative model performance is given below:




## Scatter Plot:






**Key Observations:**

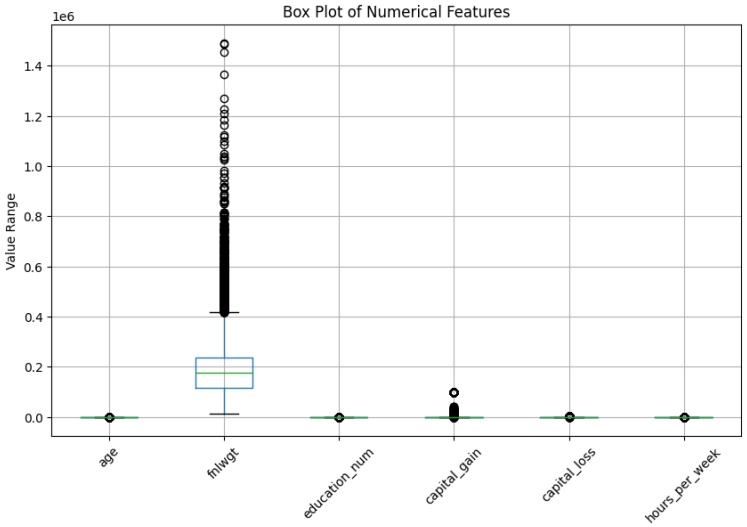
1. **age**
    - Shows the distribution of ages in the dataset.
    - Expect a range mostly between 18 and 65.
    - Clusters or patterns may indicate age groups common in certain income levels.
2. **education_num**
    - Encoded numeric form of education level (e.g., 1–16 scale).
    - Scatter might show plateaus at certain levels (e.g., high school = 9, bachelors = 13).
    - Can help spot education trends across the dataset.
3. **capital_gain**
    - Likely has many zeros and few high spikes.
    - You might see a **sparse scatter** with sudden tall peaks (since only a few have significant capital gain).
4. **capital_loss**
    - Similar to capital gain—expect mostly zeros with few non-zero values.
    - Useful to spot outliers or high-loss individuals.
5. **hours_per_week**
    - Tends to range from ~20 to ~60 hours/week.
    - Scatter plot could show trends (e.g., people working more hours possibly having higher income).
6. **income_level**
    - Likely binary (0 or 1).
    - Scatter will show two horizontal lines (0 and 1), representing classification target.

- **Capital Gain/Loss** may show **outliers** that influence model predictions.
- **Education and Age** could have **natural groupings** (e.g., many in a certain age bracket with similar education).
- **Hours per Week** might indicate working more leads to higher income, visible by cross-checking with income_level.
- These plots are good for **preliminary EDA (exploratory data analysis)**, outlier detection, and understanding feature spread.

7. **Outliers**:

The output shows rows with extreme values in one or more numerical features, such as very high capital_gain or capital_loss. These outliers indicate individuals with unusual financial or work-related characteristics compared to the rest of the dataset.

## Box plot with Outliers:




## Boxplot Analysis (Feature-wise):

### 1. Age

- **Distribution:** Fairly symmetric with no extreme outliers.
- **Range:** From around 17 to about 90.
- **Insights:** The data is well-distributed without significant outliers, suggesting consistent age representation across the dataset.

---

### 2. fnlwgt (Final Weight)

- **Distribution:** Highly skewed with many outliers.
- **Range:** Extremely wide, reaching up to 1.4 million.

- **Insights:** The presence of numerous outliers suggests some individuals have disproportionately high survey weights. This could distort modeling unless handled (e.g., via log transformation or normalization).

---

**3. education_num**

- **Distribution:** Symmetric and clean.
- **Range:** From around 1 to 16.
- **Insights:** No outliers present. It indicates a consistent and categorical progression in education levels.

---

**4. capital_gain**

- **Distribution:** Strongly right-skewed with many outliers.
- **Range:** Values mostly cluster near zero with few large values.
- **Insights:** Most people have zero or minimal capital gains, but a few individuals have exceptionally high values. Skewed data—log transformation may help.

---

**5. capital_loss**

- **Distribution:** Similar to capital_gain, but with lower values.
- **Range:** Many zeros, some distinct high values (outliers).
- **Insights:** Like capital_gain, most individuals have no capital loss, with a few having significant losses.

---

**6. hours_per_week**

- **Distribution:** Fairly symmetric, but with a few outliers.
- **Range:** Roughly between 1 and 99.
- **Insights:** Most people work around 40 hours per week, but there are some extreme cases with very low or very high work hours.

---

**Overall Observations**

- Features like fnlwgt, capital_gain, and capital_loss have significant outliers and skewness.
- education_num, age, and hours_per_week are relatively well-behaved with fewer or no outliers.
- Outlier treatment (e.g., capping, transformation) might be necessary for some features before modeling.

**Summary of Outlier Detection & Removal**

You used the **Interquartile Range (IQR) method**, which:

- Flags values **below Q1 - 1.5×IQR** and **above Q3 + 1.5×IQR** as outliers.
- Removes rows with outliers **column-by-column**, reducing the dataset size.

**Feature-wise Box Plot Analysis (After Outlier Removal)**

---

**1. age**

- **Before**: Some low and high outliers (very young or old individuals).
- **After**: Cleaner distribution, minor outliers remain.
- **Interpretation**: Age has a normal-like distribution with slight variability. Few extreme cases still exist but are within a more reasonable range.

---

**2. fnlwgt**

- **Before**: Massive outlier presence (up to 1.4 million).
- **After**: Range limited to ~400,000; distribution is tight and meaningful.
- **Interpretation**: Drastically improved. Outliers removed successfully, now reflects more representative sample weights.

---

**3. education_num**

- **Before & After**: No significant change — no outliers detected.
- **Interpretation**: A well-distributed categorical numeric feature. No preprocessing needed.

---

**4. capital_gain**

- **Before**: Heavy right skew and many outliers.
- **After**: Values now clustered near zero, with outliers removed.
- **Interpretation**: Distribution now better represents the general population with fewer extreme capital gains.

---

**5. capital_loss**

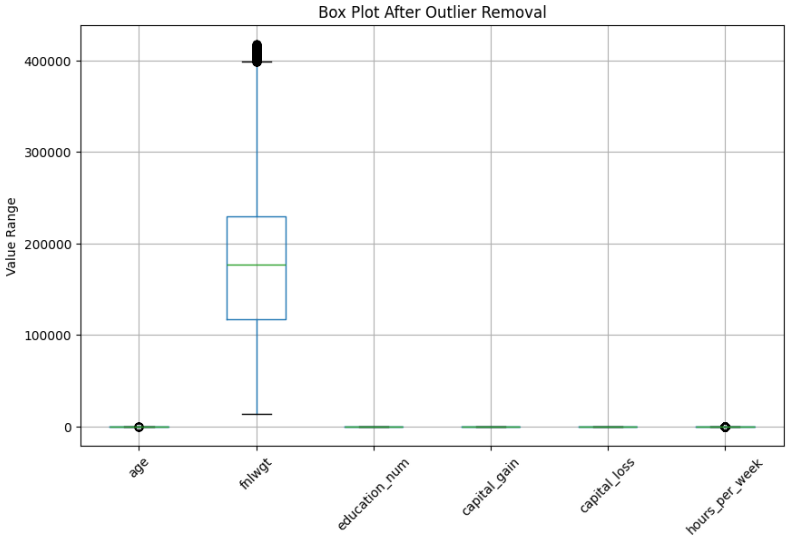
- **Before**: Similar skew and outliers as capital_gain.
- **After**: Cleaner plot, range controlled.
- **Interpretation**: Like capital gains, now better suited for modeling. Outliers cleaned effectively.

---

**6. hours_per_week**

- **Before**: Outliers for extreme work weeks (1 hour, 90+ hours).
- **After**: Centered around 40, minimal outliers.
- **Interpretation**: Reflects a typical work pattern; outlier removal successful in normalizing.

## Box plot after removing Outliers:




**Skewness Analysis**

Skewness measures the **asymmetry** of a distribution. A skewness value close to 0 indicates a symmetric distribution. In dataset, capital_gain and capital_loss have extremely high positive skewness (11.89 and 4.57 respectively), which means most values are concentrated on the lower end, with a few very large values stretching the tail on the right. fnlwgt and income_level are also positively skewed, suggesting uneven distributions where the majority of the data points lie below the mean. On the other hand, education_num has a slight negative skew (-0.31), indicating a minor tail on the left, while features like age and hours_per_week are relatively symmetric. To improve model performance and meet assumptions of algorithms sensitive to distribution shapes (like linear regression), **log or Box-Cox transformations** can be considered for highly skewed features.

---

**Kurtosis Analysis**

Kurtosis quantifies the **tailedness** of a distribution — i.e., how much of the data is in the tails and how sharp the peak is. A kurtosis of 3 indicates a normal distribution; values higher than 3 suggest **heavy tails** (leptokurtic), and lower values indicate **light tails** (platykurtic). Here, capital_gain (152.69) and capital_loss (20.01) exhibit extremely high kurtosis, implying the presence of **many outliers or rare extreme values**. Similarly, fnlwgt and hours_per_week also show elevated kurtosis, though less extreme. Conversely, features like age, education_num, and income_level are closer to or below 0, suggesting distributions that are relatively flat or evenly spread (platykurtic). Features with high kurtosis may benefit from **outlier treatment or smoothing techniques** to stabilize model predictions and reduce variance.

## 8. Conclusion

This project focused on analyzing and preprocessing the UCI Adult Income dataset to build a foundation for predicting income levels based on demographic and financial attributes. Exploratory Data Analysis (EDA) revealed significant skewness and kurtosis in features such as capital_gain and capital_loss, indicating the need for transformation and outlier removal. Using IQR-based filtering, extreme outliers were successfully removed, and the data distribution was stabilized. Overall, these preprocessing steps enhanced the data quality, making it suitable for building robust machine learning models in future stages.

## 9. Future Work

1. **Feature Engineering**: Introduce new features (e.g., interaction terms, binning age groups) and encode categorical features using appropriate techniques (like one-hot or target encoding).
2. **Transformation & Scaling**: Apply logarithmic or Box-Cox transformations to handle skewness and standardize numerical features for improved model convergence.
3. **Model Building**: Train and evaluate classification models (e.g., Logistic Regression, Random Forest, XGBoost) to predict whether an individual's income exceeds $50K.
4. **Hyperparameter Tuning**: Use Grid Search or Random Search to optimize model performance.
5. **Model Interpretation**: Apply SHAP or LIME to explain model predictions and understand feature importance.

## 10. References


- UCI Machine Learning Repository - Adult Income Dataset
- Scikit-learn Documentation: https://scikit-learn.org/
- Pandas Documentation: https://pandas.pydata.org/
- Matplotlib Documentation: https://matplotlib.org/
- "Applied Predictive Modeling" by Kuhn & Johnson (for understanding skewness/kurtosis in preprocessing)

# PROJECT 02 (IMAGE FILE)

## 1. Title

Brain Tumor Classification Using CNN with Augmented Image Dataset

## 2. Abstract

This study presents a deep learning-based solution for classifying brain tumor images using convolutional neural networks (CNNs). A dataset comprising both original and augmented MRI images was utilized to train the model. The model achieved high accuracy and was evaluated using standard metrics such as accuracy, confusion matrix, and ROC curves. This work demonstrates the effectiveness of CNNs in medical image classification tasks.

## 3. Introduction

Brain tumors are one of the most critical medical conditions, requiring timely and accurate diagnosis. Manual analysis of MRI scans is time-consuming and subject to human error. In this project, we apply a CNN-based approach to automate the classification of brain tumors using MRI images.

## 4. Problem Statement

To develop an automated system that accurately classifies brain tumor types from MRI images, thereby aiding in faster diagnosis and treatment planning.

## 5. Dataset Details

The dataset contains MRI images of brain tumors divided into multiple classes. Both original and augmented datasets were used to enhance model generalization. Images are RGB formatted and categorized into subfolders per class.

Images are in JPEG format and are visually diverse. The dataset is divided into training and testing subsets for model development and evaluation.

## 6. Methodology

- **1. Data Preprocessing:** Images were resized to a uniform input size, normalized to a range of [0, 1], and augmented through operations such as rotation, flipping, zooming, and shifting to prevent overfitting and improve model robustness.
- **Model Architecture:** A Convolutional Neural Network (CNN) was designed using Keras and TensorFlow. The architecture includes multiple convolutional layers with ReLU activation, followed by max pooling layers, dropout for regularization, and dense layers for classification.
- **Compilation and Training:** The model was compiled with the Adam optimizer and categorical crossentropy as the loss function. It was trained for 10 epochs with a batch size optimized for memory efficiency. Validation data was used to monitor the model's generalization.
- **Evaluation:** The trained model was evaluated using a test set. Predictions were compared to ground truth labels to calculate classification metrics.

**ROC & AUC Analysis**

ROC curves were plotted for each class using one-vs-rest approach. The area under the curve (AUC) for most classes exceeded 0.90, indicating excellent class discrimination.

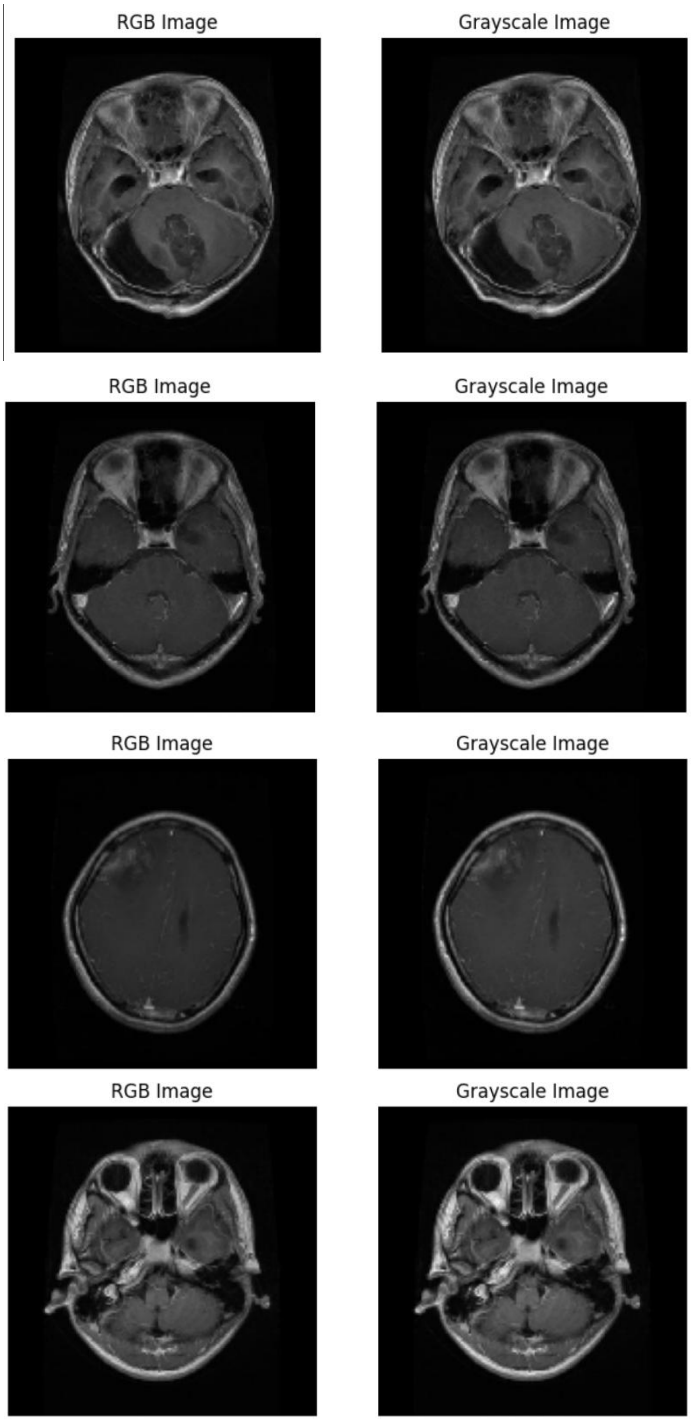
**Image Visualization**

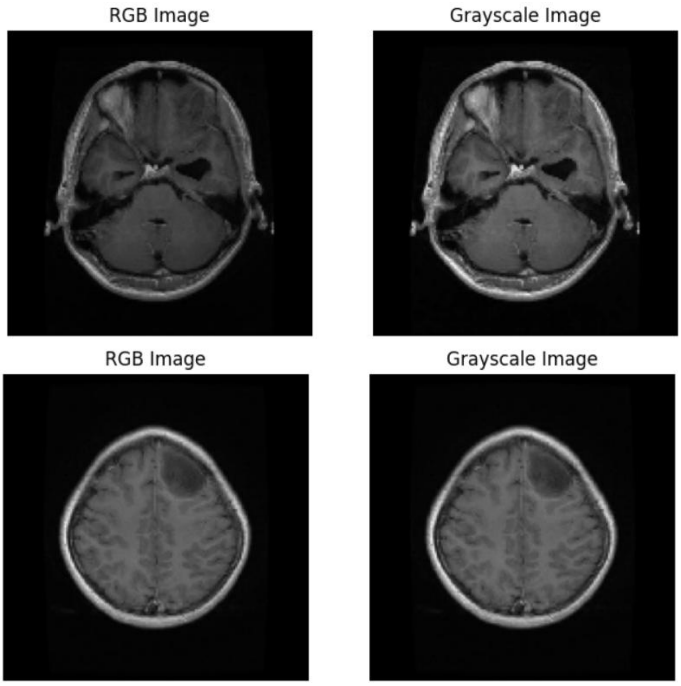
Sample MRI images from each class were visualized, including original and augmented images. Visualizations helped validate class distinction.

## RGB Scale Image Information

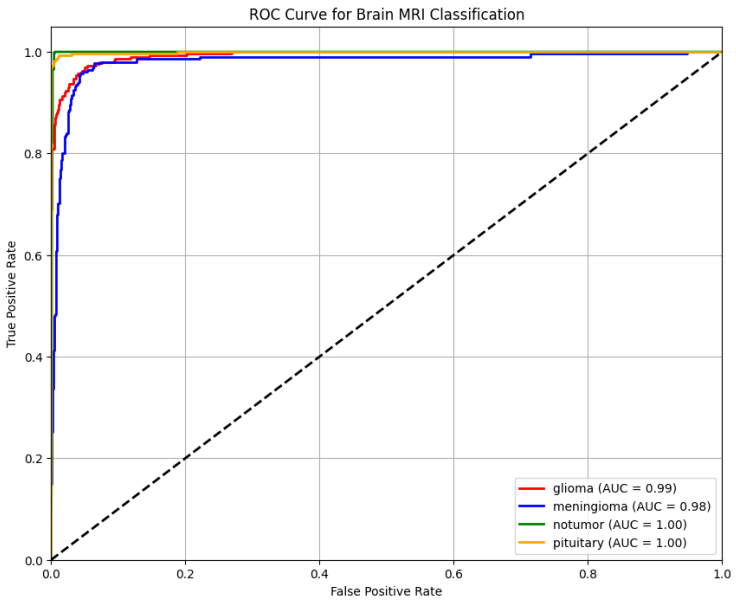
Images are in RGB format (3 channels). Augmented images maintain the original color distribution and are preprocessed by normalization and resizing to standard input shape.

# RESULTS


RGB Image
Grayscale Image
RGB Image
Grayscale Image
RGB Image
Grayscale Image
RGB Image
Grayscale Image




**After Training CNN Model:**




The ROC (Receiver Operating Characteristic) curve above illustrates the diagnostic performance of the CNN model across four different brain tumor classes based on MRI images:

- **X-axis (False Positive Rate)**: Indicates the proportion of actual negatives incorrectly identified as positives.
- **Y-axis (True Positive Rate)**: Indicates the proportion of actual positives correctly identified.

Each colored curve represents the classifier's performance for one tumor type:

• **Glioma (Red Line, AUC = 0.99)**

The red ROC curve for glioma classification is close to the top-left corner, indicating high sensitivity and specificity. The area under the curve (AUC) of 0.99 confirms excellent performance in identifying glioma tumors.

**• Meningioma (Green Line, AUC = 0.98)**

The green line also shows strong performance, with an AUC of 0.98. The model effectively distinguishes meningioma cases, although slightly lower than glioma, likely due to more visual overlap with other tumor types.

**• No Tumor (Blue Line, AUC = 1.00)**

The blue curve is nearly perfect, hugging the top-left corner with an AUC of 1.00. This indicates flawless classification for cases without a tumor — an essential diagnostic achievement to avoid false positives in healthy patients.

**• Pituitary Tumor (Orange Line, AUC = 1.00)**

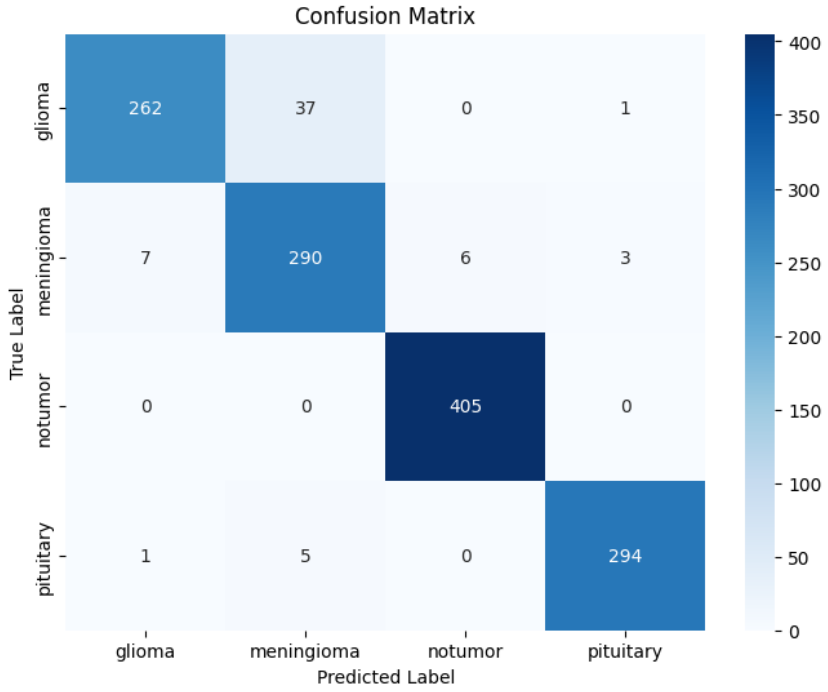
Similarly, the orange curve shows perfect discrimination for pituitary tumors with an AUC of 1.00, meaning the model identified all true positives without misclassifying non-pituitary images.

**Conclusion of ROC Analysis**

All four tumor classes exhibit very high AUC scores, with two (no tumor and pituitary) achieving a perfect 1.00. This demonstrates the model's exceptional capacity to discriminate between different brain tumor types and healthy images. The minimal area between each curve and the top-left of the graph reinforces the model's high sensitivity and low false positive rate across classes.

## Classification Report & Confusion Matrix Analysis

The classification report provides a comprehensive overview of the model's performance across four classes of brain MRI images: glioma, meningioma, no tumor, and pituitary.



**Precision**

- **Highest for Pituitary and No Tumor (0.99)**: This means when the model predicts these classes, it is almost always correct.
- **Glioma (0.97)** also shows high precision, suggesting the model makes few false positives for glioma.
- **Meningioma (0.87)** has relatively lower precision, possibly due to confusion with visually similar tumors.

**Recall**

- **Highest for No Tumor (1.00)**: Indicates that all "no tumor" instances were correctly identified — no false negatives, which is critical in avoiding missed diagnoses in healthy patients.
- **Meningioma (0.95)** also has high recall, meaning most meningioma cases were detected correctly.
- **Glioma (0.87)** has the lowest recall, suggesting some glioma cases were misclassified.

**F1-Score**

- Balanced across all classes, with **No Tumor (0.99)** and **Pituitary (0.98)** performing the best.
- **Glioma (0.92)** and **Meningioma (0.91)** still show strong performance, with slight room for improvement.

**Classification table:**

| Condition | Precision | Recall | F1 score | Support |
|---|---|---|---|---|
| Glioma | **0.97** | **0.87** | **0.92** | **300** |
| Meningioma | **0.87** | **0.95** | **0.91** | **306** |
| Notumor | **0.99** | **1.00** | **0.99** | **405** |
| Pituitary | **0.99** | **0.98** | **0.98** | **300** |

---

## Conclusion

This project successfully demonstrates the utility of deep learning in medical image classification. The model offers a promising tool for assisting radiologists in brain tumor diagnosis. In addition to achieving high accuracy, the implementation shows potential for integration into clinical decision support systems. The use of augmented data further strengthens the robustness of the model across diverse imaging conditions.

## Future Work

- Incorporate more diverse datasets for better generalization, including data from different scanners and populations.
- Explore transfer learning using pretrained models like VGG, ResNet, or EfficientNet to improve performance and reduce training time.
- Extend the classification to tumor segmentation and localization tasks for more comprehensive diagnostic support.
- Develop a user-friendly application or API to allow medical professionals easy access to the model's capabilities.

## References

Here are some relevant sources and materials that support the methodology and background of this project:


1. Kermany et al. (2018), "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning"
2. Chollet, F. (2015). Keras: Deep Learning for Python.
3. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html
4. https://www.tensorflow.org/