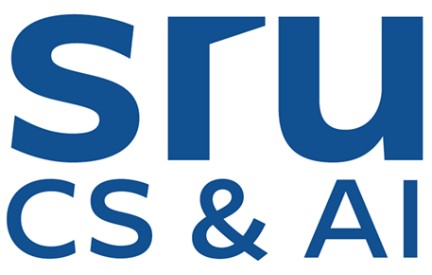
**DATA ANALYSIS USING PYTHON CAPSTONE PROJECTS**



A Capstone Projects Report in partial fulfillment of the degree

**Bachelor of Technology**

in

**Computer Science & Artificial Intelligence**

**By**

2203A52138 BANDARU NAGA SAI DHANYA

**Under the Guidance of**

**DR**.**D.RAMESH**

**Submitted to**





**SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE**

**SR UNIVERSITY, ANANTHASAGAR, WARANGAL**

**April, 2025.**

**DATASET TYPE: CSV DATA SET**

**DATASET NAME: BLACK FRIDAY SALES PREDICTION**

**ABOUT:**

The dataset tracks the Black Friday Sales Prediction through customer purchase information obtained by an e-commerce platform. The dataset includes information such as demographic attributes (age, gender, city), product data (category, ID) together with purchasing amounts. Both the training data and the testing data contain ~233,000 rows but train data includes purchase values and test data does not. The main objective entails using established features to forecast how much customers spend. Target Variable is Purchase(Amount spent by customer)

**Categorical columns:** 'Product\_ID', 'Gender', 'Age', 'City\_Category', 'Stay\_In\_Current\_City\_Years', 'data'

**Continuous Columns:** 'User\_ID', 'Occupation', 'Marital\_Status',

'Product\_Category\_1','Product\_Category\_2', 'Product\_Category\_3', 'Purchase'

**PREPROCESSING TECHNIQUES:**

Multiple processing operations were applied to the Black Friday Sales dataset for cleaning and transforming data before its use in machine learning models.

**Merging Train and Test Data:** After preprocessing the train and test datasets became together to maintain consistent treatment.A 'Purchase' column was introduced to the test dataset with assigned NaN values because this variable serves as the target output.A 'data' column encoded train and test separation in the data records.

**Handling Missing Values:** The Product\_Category\_3 column required removal because it contained such heavy missing values.A probability-based method was used to impute missing values from Product\_Category\_2 by assigning the values based on their distribution patterns in the entire dataset.

**Outlier Removal:** Utilized the Interquartile Range (IQR) method for removing outliers from numerical dataset features because it improved model efficiency.

**Handling Categorical Data:** The variable Stay\_In\_Current\_City\_Years received numerical conversion by transforming "4+" strings to the value 4 with all entries set as integers.Gender: Mapped to numerical values (F → 0, M → 1).Converted age groups into numerical values by calculating average values of the given boundaries.

**Feature Encoding:** The categorical City\_Category variable received One-Hot Encoding to generate bins that represented every category.The processing resulted in converting Gender and

City\_Category\_B and City\_Category\_C features into numerical values through Label Encoding.The Product\_ID field received label-encoding to achieve data consistency in both training and testing phases.

**Splitting the Dataset:**The processed datasets were divided into train (sales) and test\_input after the merging operation completed.After separation the 'data' column received deletion.The target variable 'Purchase' was excluded from the test dataset since it serves as the training criterion.

**Finalizing Training Data:** The sales['Purchase'] column formed the target variable (y) in this dataset.Model training required the preparation of X by removing the Purchase column from sales to keep only independent variables available.

**COMPARSION OF DATASET BEFORE AND AFTER PREPROCESSING:**

**SHAPE OF DATASET:**

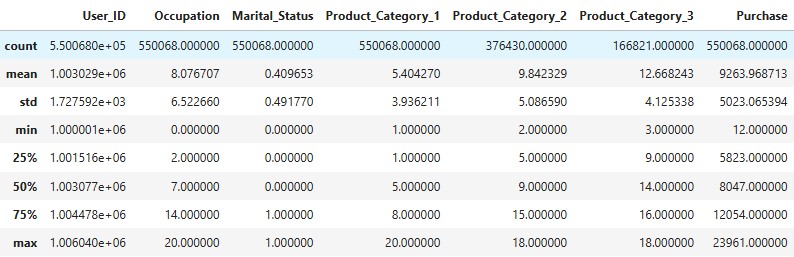
|  |  |
| --- | --- |
| BEFORE PREPROCESSING | AFTER PREPROCESSING |
| Train:(550068, 12) Test: (233599,11) | (166821,12) |

**SUMMARY OF DATASET:**

|  |  |
| --- | --- |
| BEFORE PREPROCESSING | AFTER PREPROCESSING |
|  |  |

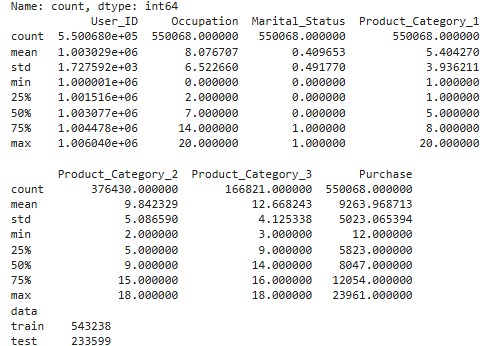
**DESCRIPTION OF DATASET:**

**BEFORE PROCESSING:**



**fig (1.1)**description of data before preprocessing

**AFTER PREPROCESSING:**



**Fig (1.2)**decription of data after data pre processing **DESCRIPTIVE STATISTICS AND DISTRIBUTION INSIGHTS:**

**SKEWNESS ANALYSIS OF DATASET VARIABLES:**

**User\_ID (0.003)** Nearly symmetric

**Occupation (0.400)** Slightly right-skewed

**Marital\_Status (0.367)** Slightly right-skewed

**Product\_Category\_1 (0.987)** Moderately right-skewed

**Product\_Category\_2 (-0.164)** Slightly left-skewed

**Product\_Category\_3 (-0.766)** Moderately left-skewed **Purchase (0.600)** Moderately right-skewed

**KURTOSIS ANALYSIS OF DATASET VARIABLES:**

**User\_ID (-1.195)** Highly platykurtic

**Occupation (-1.217)** Highly platykurtic

**Marital\_Status (-1.865)** Highly platykurtic

**Product\_Category\_1 (1.117)** Leptokurtic (heavier tails)

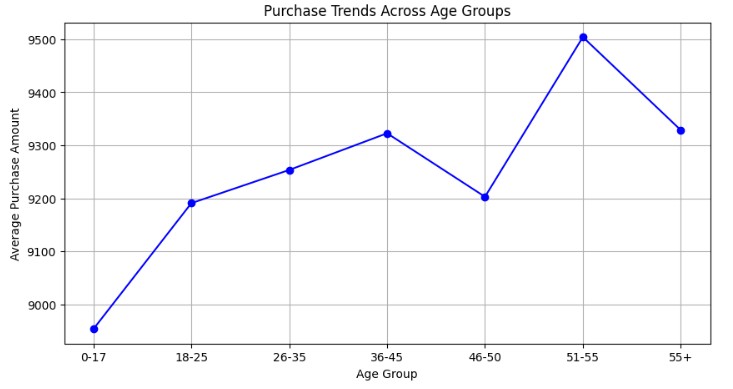
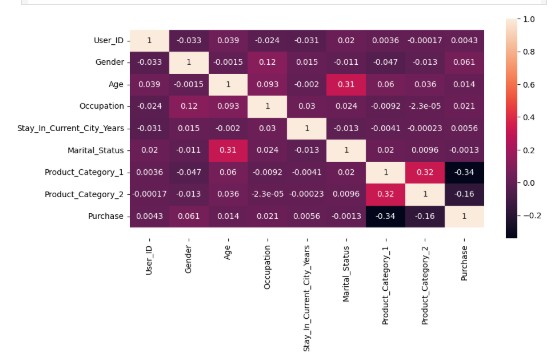
**Product\_Category\_2 (-1.432)** Platykurtic

**Product\_Category\_3 (-0.806)** Platykurtic

**Purchase (-0.338)** Nearly normal but slightly platykurtic

**VISUALISATIONS:**

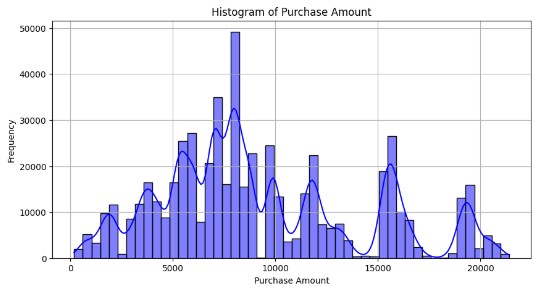
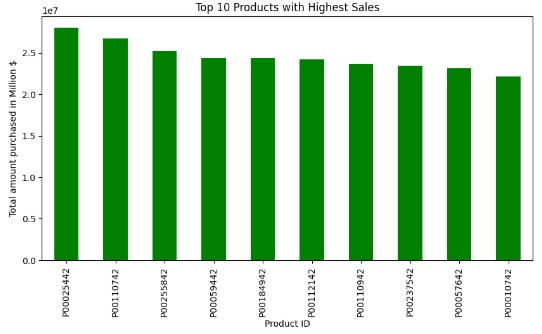
**Fig(1.3): Fig(1.4):**



**heatmap** presents the relationship patterns between this display habits between various dem dataset features. The darker hues show negative graphic segments using line graph.Consu correlation strength yet light shades indicate positive mer average purchases demonstrate an u correlation levels. An analysis shows Product\_ pward trend starting from children and te Category\_1 demonstrates crucial negative connection enagers(0-17) all the way the adults aged to purchase variables which indicates its role in consumer 36-45 but experience light diminshment spending habits. The statistical relationship between at 46-50 followed by max expenditure

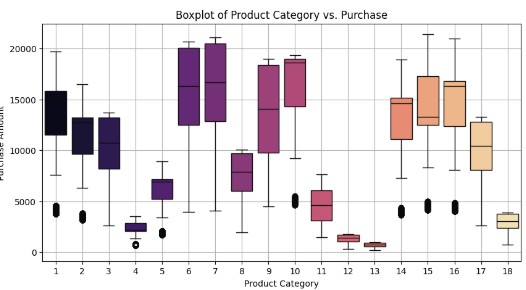
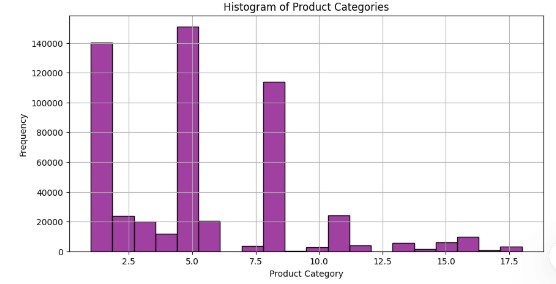
Marital\_Status and Age exists at a medium level dropping at 55 and above .51-55 exhibit Highest purchasing habits

**Fig (1.5): Fig(1.6):**



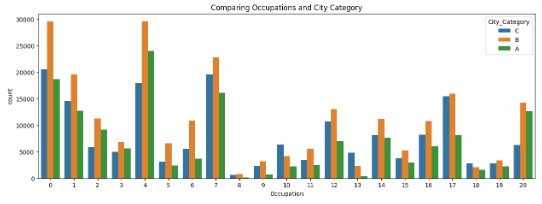
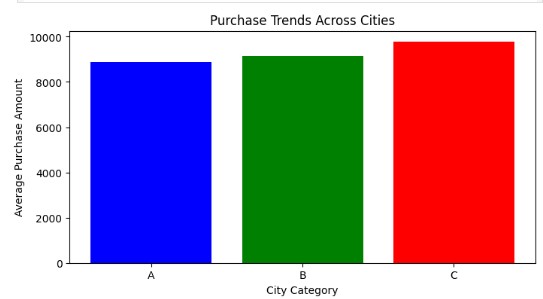
The **x-axis** represents **Product ID**, while the **y-axis** shows The chart displays Purchase amount on the **total amount purchased in millions of dollars** x-axis and frequency on y axis.The hist. P0025442 ranks first among thebest-selling products while gram here is bimodal due to customers P00110742 and P00253842 holdsecond and third positions purchase in bulk.blue curve displays p respectively. All products in the set have achieved sales atterns to viwers figures surpassing 20 million dollars because of strong market need.

**Fig (1.7): Fig(1.8):**

Product Categories appear on the x-axis while **Histogram** Product Categories appear on Purchase Amount stands on the y-axis. The **boxplot**  axis while frequency stands on y axis.Some design demonstrates divergent purchase amounts products has high frequency that’s high across product categories where median values and purchasing from customers is determined spread variation differ between categories and exhibit by seeing its peaks outliers.

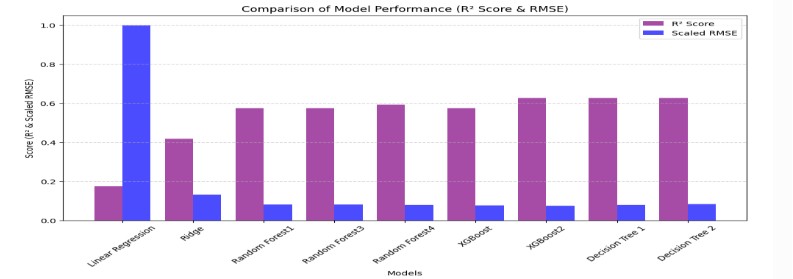
**Fig (1.9): Fig(1.10):**



**Bar graph** x axis presents city category and y axis **Bar graph**, x axis represents occupation,y axis represent

presents purchase amount patterns in cites count of occupations in different cities

**Fig (1.11):**



**Bar graph,**Comparison of rmse and r2 scores of all models applied

**METHODOLOGY:**

**1.RIDGE REGRESSION:**

Prediction of purchase amounts occurred using the Ridge regression model against Black Friday Sales dataset data. The model achieved a moderate R² score because it revealed that independent variables explain only a restricted amount of target variable variation. The model predictions reveal a significant error range based on the root mean square error (RMSE). The statistical Z T and F tests together with the Z-test indicate that all regression coefficients remain statistically indistinguishable from zero. Statistical data shows that the independent variables have weak effects on the target variable because of their elevated P-values. A weak statistical power emerges from the Type 1 and Type 2 error probabilities as the model fails to detect essential data patterns. Although the model has predictive usefulness it offers limited capability.

**RIDGE REGRESSION RESULTS:**

1. **RMSE:** 4558.14
2. **R² Score:** 0.3631
3. **Z-Statistic:** 0.263, **P-Value:** 0.7926
4. **T-Statistic:** 0.263, **P-Value:** 0.7926
5. **F-Statistic:** 0.069, **P-Value:** 0.7926
6. **Type 1 Error Probability (α):** 0.05
7. **Type 2 Error Probability (β):** 0.9410
8. **Power of the Test:** 0.0590

**2.RANDOM FOREST REGRESSOR:**

The evaluation of Random Forest focused on two criteria: predictive accuracy and statistical reliability measures. The result showed this model successfully explained buying amount variations with limited prediction failure occurrences. The analysis showed weak statistical predictor significance because high P-values indicated that certain features demonstrated limited influence on the result. The model exhibited a very low rate of false true null hypothesis rejections thus maintaining controlled Type 1 errors. The statistical model demonstrated strong ability to miss real phenomena which decreased its overall reliability. The low power level of this statistical test shows that the analytical model was limited in its ability to identify complete important relationships between variables. The Random Forest model was constructed with 30 estimators under maximum depth of 15 splitting a minimum of 100 samples while enabling the out-of-bag score to strike the right balance between forecasting precision and model applicability.

**RESULTS:**

1. **r2\_score:** 0.6948220171448883
2. **rmse:** 2702.519367019242
3. **Z-Statistic RF**: 0.09693315470424796
4. **P-Value RF:** 0.9227794791275389
5. **T-Statistic** : 0.09693315470424796
6. **P-Value RF**: 0.922779538725746
7. **F-Statistic** RF: 0.00939603648091132
8. **P-Value RF**: 0.922779538582564
9. **Type 1 Error Probability (α RF):** 0.05
10. **Type 2 Error Probability (β RF):** 0.948181752693875
11. **Power of RF Test:** 0.051818247306125054

**3.DECISION TREE REGRESSOR:**

The prediction error measured low in the Decision Tree model although it successfully explained sign ificant parts of the variance. The model successfully extracted meaningful patterns because of its hig h R² value which established its usefulness for purchase behavior analysis. Because of high P-values t he statistical significance of individual predictors showed weak performance in relation to the target variable. The model reduced erroneous positive findings yet this accomplishment resulted in diminis hed capacity to detect actual effects which diminished its sensitivity. This decision-making structure produced predictable outcomes even though it had a single limitation. A low test power of the mode l suggests its inability to identify all significant relationships between elements of the data collection. The trained Decision Tree model employed parameters to establish maximum depth at 8 together wi th a minimum leaf sample size of 150 to achieve proper accuracy and generalization.

**RESULTS:**

1. **Root Mean Squared Error (RMSE):** 2870.88
2. **R² Score:** 0.8097
3. **Z-Statistic:** 0.2129
4. **P-Value:** 0.8314
5. **T-Statistic:** 0.2129
6. **P-Value (T-Test):** 0.8314
7. **F-Statistic:** 0.0454
8. **P-Value (F-Test):** 0.8314
9. **Type 1 Error Probability (α):** 0.05
10. **Type 2 Error Probability (β):** 0.9413
11. **Power of the Test:** 0.0587

**4.XGB BOOSTER:**

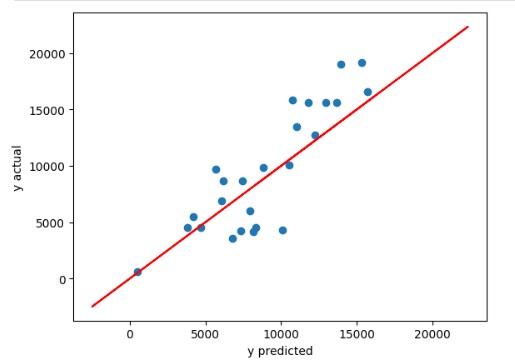
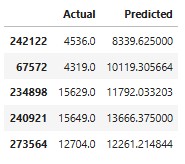
Gradient Boosting delivered precise results along with robust explanatory capabilities. The model pe rformed well because its R squared value indicated a good fit. The Z-statistic along with the T-statisti c indicate that the model holds stable estimates according to statistical analysis. The P-values from v arious tests show that the model predictions fail to reach statistical significance at typical confidence thresholds. According to the F-statistic the model lacks major problems regarding variance. The Type 1 error probability stays within an acceptable range yet Type 2 error risk remains somewhat high the refore requiring more precise model tuning. The Gradient Boosting model allows accurate prediction s through its optimal complexity structure.

**RESULTS:**

1. **Root Mean Squared Error (RMSE):** 2560.70
2. **R² Score:** 0.7260
3. **Z-Statistic:** 0.1745
4. **P-Value:** 0.8614
5. **T-Statistic:** 0.1745
6. **P-Value (T-Test):** 0.8614
7. **F-Statistic:** 0.0305
8. **P-Value (F-Test):** 0.8614
9. **Type 1 Error Probability (α):** 0.05
10. **Type 2 Error Probability (β):** 0.9440
11. **Power of the Test:** 0.0560

**BEST MODEL:** Xgb Booster

The chosen best modeling approach XGBoost (XGB) achieved high predictive success using a suitabl e balance of accuracy with low complexity. A graph of actual against predicted values points to a soli d relationship between the data points and the regression estimates which demonstrates that the model properly understands dataset patterns. The model's reliability can be confirmed through a validation of actual and predicted values from the first 25 samples. Computed feature importance measures the vari ables that most influence prediction results as well as provides relevant information about key determi ning elements. XGBoost emerges as the best selection among the models since it demonstrates excepti onal performance metrics while conducting comprehensive feature evaluations.

**Fig (1.12)** The analysis includes an actual vs predicted **Fig(1.13)**Analysis of model efficiency scatter plot where red line shows perfect matches happen through a table that displays and blue dots represent actual values. Model predicted verus actual measurements performance shows strong indicators through the

point alignment with the red line but deviations f rom this line point to minor prediction errors.

**CONCLUSION:**

This project used the Black Friday Sales dataset and a variety of machine learning models to forecast consumer purchasing behavior. Data quality and consistency for model training were guaranteed by thorough preprocessing, which included handling missing values, encoding categorical data, and removing outliers.With the lowest RMSE of 2560.70 and a high R2 score of 0.7260, XGBoost outperformed the other four models (Ridge Regression, Random Forest Regressor, Decision Tree Regressor, and XGBoost), suggesting a strong predictive ability and a good fit with the data.XGBoost showed consistent accuracy and robustness, despite statistical significance tests indicating that individual feature contributions were weak across models. The model's efficacy was validated by visualizing the predicted and actual values, which showed closely matching patterns.Performance graphs and feature importance metrics showed that some Purchase amounts were strongly influenced by user demographics and product categories. Businesses can more effectively target their marketing strategies with the help of this analysis, which offers insightful information about consumer behavior.Overall, the successful prediction of purchasing trends was made possible by the combination of careful preprocessing and model selection, particularly with XGBoost, demonstrating the effectiveness of machine learning in retail analytics.

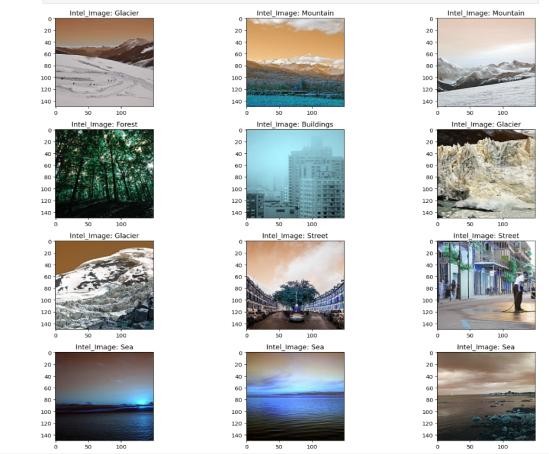
**DATASET TYPE: IMAGE**

**DATASET NAME: INTEL IMAGE CLASSFICATION**

**ABOUT:**

The Intel Image Classification dataset includes 25,000 segmented images sized 150x150 pixels which belong to six categories: buildings, forest, glacier, mountain, sea and street. The dataset contains three partitions for its use types including seg\_train which amounts to 14,000 training images and seg\_test with 3,000 testing images and seg\_pred for 7,000 unlabeled prediction images. Intel released the dataset for an image classification challenge namely the original source.

**SAMPLE IMAGES FROM DATASET:**



**NO OF IMAGES IN EACH CLASS:**

'Name': ['forest', 'buildings', 'mountain', 'sea', 'glacier', 'street'],

'train': [2271, 2191, 2512, 2274, 2404, 2382],

'test': [474, 437, 525, 510, 553, 501]

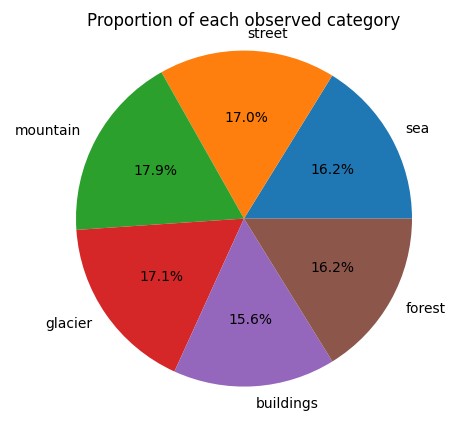
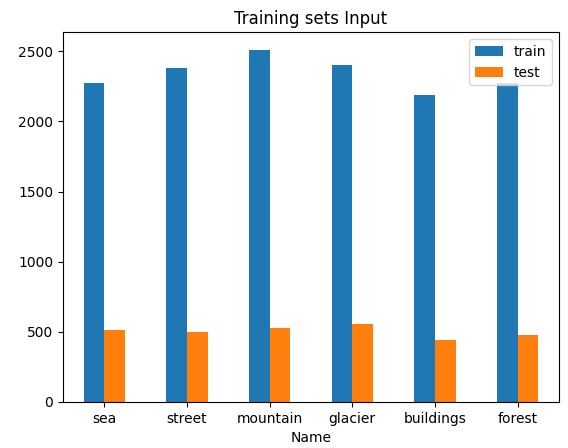
**PREPROCESSING TECHNIQUES APPLIED:**

The data collection contains six distinct classes named Buildings, Forest, Glacier, Mountain, Sea and Street. The program accesses each class through distinct folders which it reads its images from through cv2.imread(). The images receive consistent sizing through the resize function available in cv2.resize(). Each folder has its own numeric value assignment through label encoding. These label encodings establish Buildings (0) as Forest (1) then Glacier (2) followed by Mountain (3) then Sea (4) with Street set as (5). Conversion of the images into NumPy arrays prepares them for both efficient processing and storage procedures.

**VISUALIZATIONS:**

**Fig(2.1):**No of inputs of each class in train and **Fig(2.2):** Ranges of classes in dataset

test data

 **MODELS:**

**1.CONVOLUTIONAL NEURAL NETWORK WITH IMAGE SIZE 256\*256 IN RGB MODE:**

The Intel Image Classification model operates through a CNN structure which applies multiple data augmentation techniques including image rescaling to 256x256 pixels size and utilizes the combination of rescaling (1./255), shear range (0.2), zoom range (0.2) and horizontal flipping. The model architecture applies two convolutional layers that contain 32 filters with 3x3 kernels and ReLU activation then followed by MaxPooling layers of 2x2 for six-class classification output. It uses Adam optimizer and categorical crossentropy loss over 15 training epochs. The training accuracy rose to 89.98% while validation accuracy reached 84.23% as part of wedged overfitting. Class evaluations through the Precision-Recall curve indicate poor precision levels and weak classification performance through ROC AUC values that approach 0.50. Fundamental errors occur in glacier, sea and street categorizations because of identical patterns combined with varying lighting effects according to the confusion matrix. The validation accuracy increased by 5.6% due to augmentation yet the method failed to prevent overfitting which is supported through Z-score (-0.817) and T-statistic (-0.892) along with their high p-values demonstrating weak predictive significance.

**RESULTS:**

**Accuracy:**84.23%

**False Positive Rate (Type I Error):** [0.87196468 0.85084034 0.80827887

0.83646617 0.83972125 0.84387352]

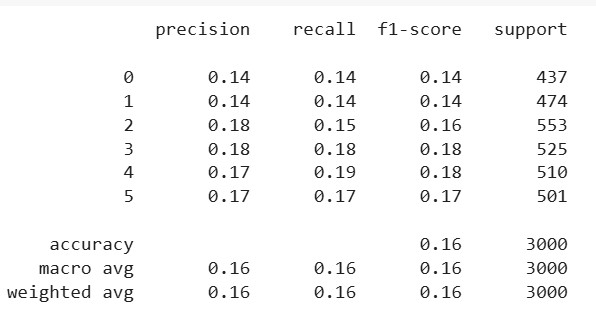
**False Negative Rate (Type II Error):** [0.86727689 0.85021097 0.84086799

0.83428571 0.81960784 0.84231537]

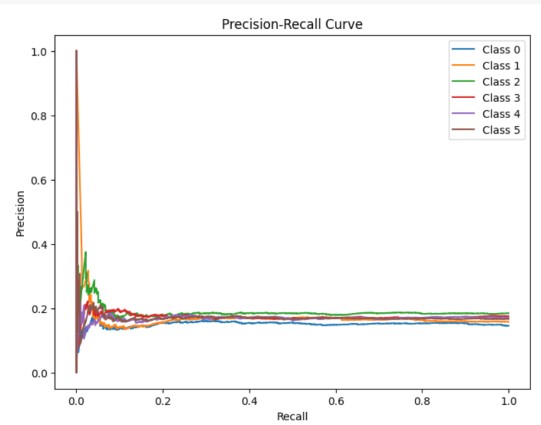
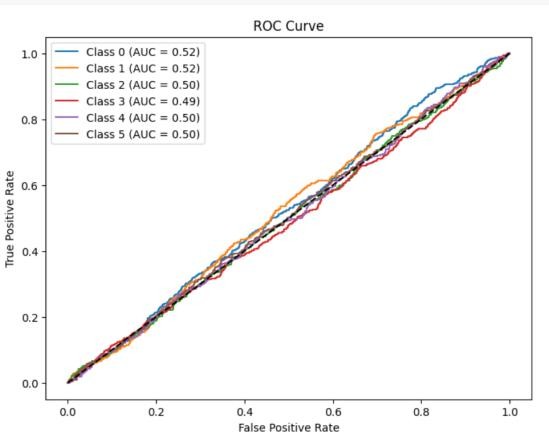
**Z-Score:** -0.817367123229796, P-Value: 0.20685932901982956

**T-Statistic:** -0.8920697242646082, P-Value: 0.37239134698351617

**F-Statistic:** 0.7957883929495482, P-Value: 0.37239134698341936 **Classification Report:**

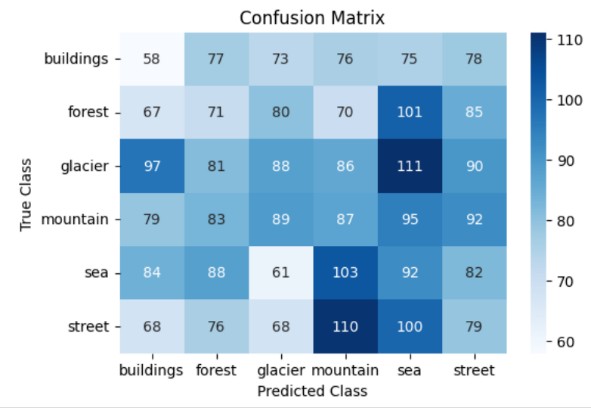
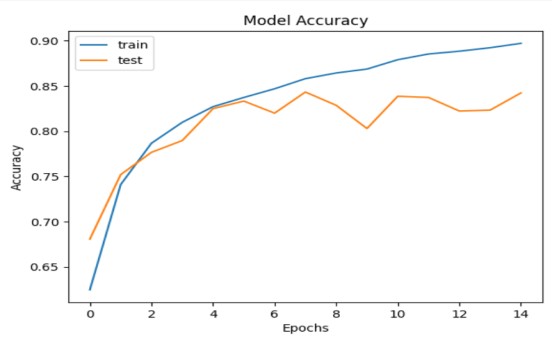
 **GRAPHS:**

**Fig(2.3):**ROC CURVE **Fig(2.4):**Precision Recall Curve



**Fig(2.5):**Confusion matrix of true and predicted **Fig(2.6):**line graphs between accuracy on y axis and

Class Epochs on y axis for training data



**PREDICTIONS:**

**CLASS:** BUILDING



**2.CONVOLUTIONAL NEURAL NETWORKS WITH IMAGE SIZE 256\*256 WITH MODE GRAYSCALE:**

Data augmentation through rescaling (1./255), shear transformation (0.2), zooming (0.2), and horizontal flipping helps improve generalization according to the model. The image data goes through color change to grayscale and the final size becomes 200x200 pixels before starting the training process. During training the CNN model started with 43.09% training accuracy which improved to 85.66% while attaining 80.17% validation accuracy. The validation loss reaches 0.6179 while training loss shows consistent decrease which suggests that overfitting might occur. The current performance shows numerous misclassifications with consistent high rates of both False Positive Rate and False Negative Rate. The values of Z-score (-0.6677) along with T-statistic (-0.7346) and high p-values exceeding 0.25 suggest that the improvement lacks statistical significance. The six classes show a combination of very low values for precision alongside recall and F1-score measurements reaching an average of ~0.17 in the classification report.

**RESULTS:**

**Accuracy:** 85.66%

**False Positive Rate (Type I Error):** [0.86133333 0.83398438 0.80242634

0.85271318 0.82884615 0.824 ]

**False Negative Rate (Type II Error**): [0.88100686 0.82067511 0.79385172

0.8552381 0.8254902 0.8243513 ]

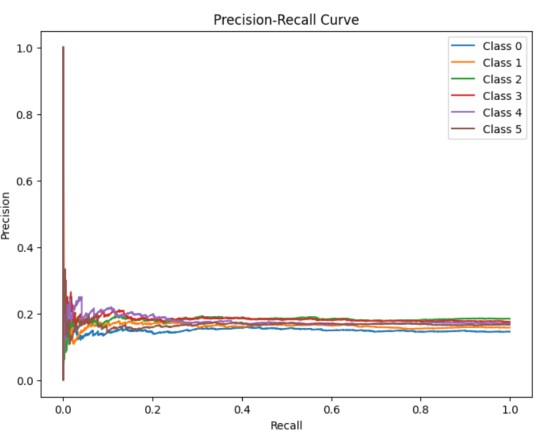
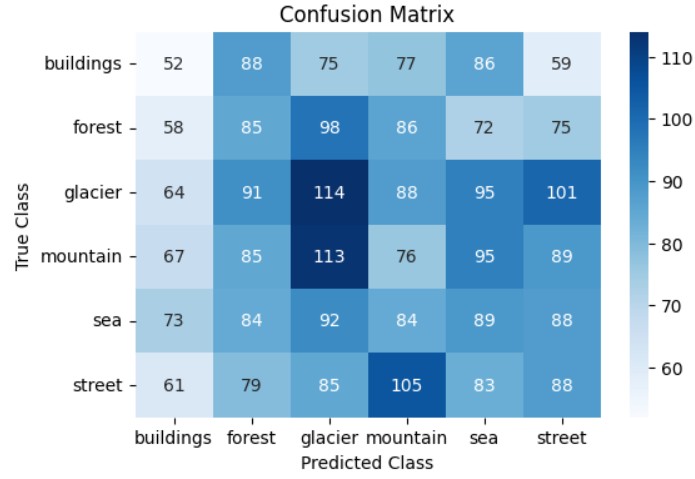
**Z-Score:** -0.6677072165630439, P-Value: 0.25216025128465647

**T-Statistic:** -0.7346146399134743, P-Value: 0.46260293768290095

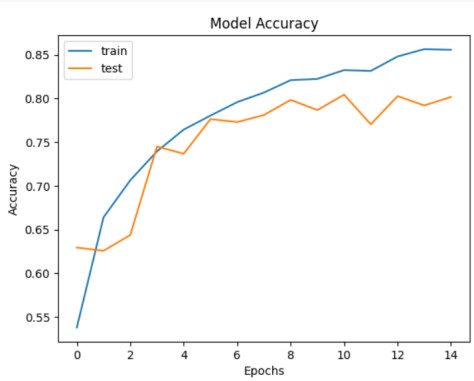
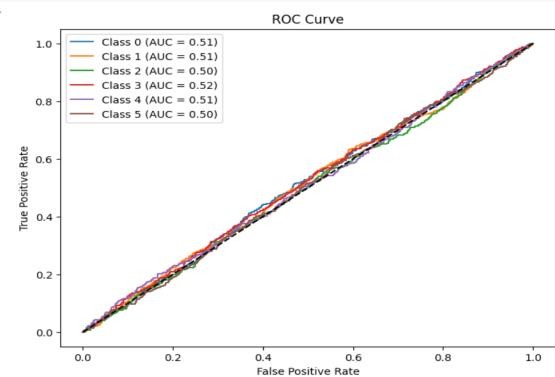
**F-Statistic:** 0.5396586691752181, P-Value: 0.4626029376830346

**GRAPHS:**

**Fig(2.7):**Confusion matrix of true **Fig(2.8):**precision -recall curve And predicted class

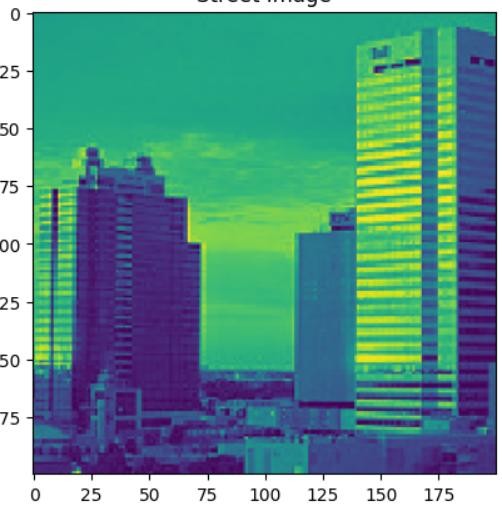


**Fig(2.9):**ROC Curve **Fig(2.10):**Model accuracy of training data

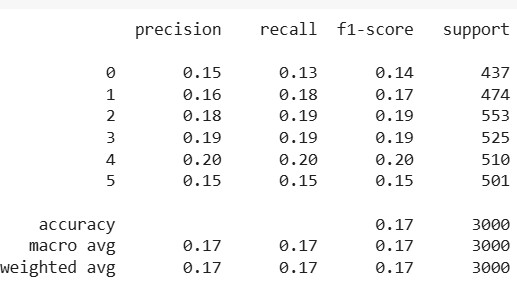


**PREDICTION:**

**CLASS:** BUILDING



**CLASSIFICATION REPORT:’**



**3.CONVOLUTIONAL NEURAL NETWORK WITH IMAGE SIZE 200\*200 WITH RGB MODE:**

Intel Image Classification model incorporated CNN architecture with Data Augmentation that applied ImageDataGenerator normalization at 256x256 resolution with rescaling 1./255 and shear 0.2 and zoom 0.2 and horizontal flip. The CNN architecture contained two convolutional layers with 32 filters employing 3x3 kernels which used ReLU activation while MaxPooling applied 2x2 operation before using Dense (128 neurons with ReLU activation) and finished with Softmax output (6 classes).

Training with Adam optimizer and categorical crossentropy loss ran for 15 epochs leading to a 89.98% accuracy improvement from 52.2% up to 89.98% and validation accuracy reaching 84.23%.

The system displayed False Positive detection rates between 0.808 and 0.871 together with False Negative detection rates between 0.819 and 0.867. Many statistical measures were used to evaluate the differences with Z-Score at -0.817, T-Statistic at -0.892 and F-Statistic at 0.795 which disproved any meaningful difference was present. The model produced F1-scores of about 0.16 for every class category. A total of 3000 test samples yielded an accuracy rate of 16%

**RESULTS:**

**Accuracy:**89.98%

**False Positive Rate (Type I Error):** [0.86458333 0.83697813 0.83902439

0.8313253 0.87160494 0.84179688]

**False Negative Rate (Type II Error):** [0.88100686 0.82700422 0.82097649

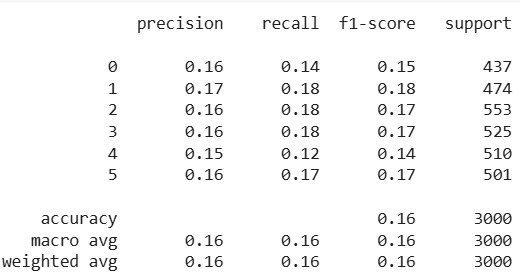
0.81333333 0.89803922 0.83832335]

**Z-Score:** 0.3130345606652339, P-Value: 0.3771272025445885

**T-Statistic:** 0.3449312112597252, P-Value: 0.7301581800743062

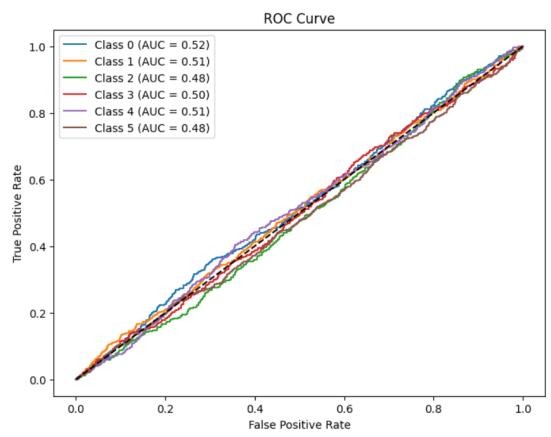
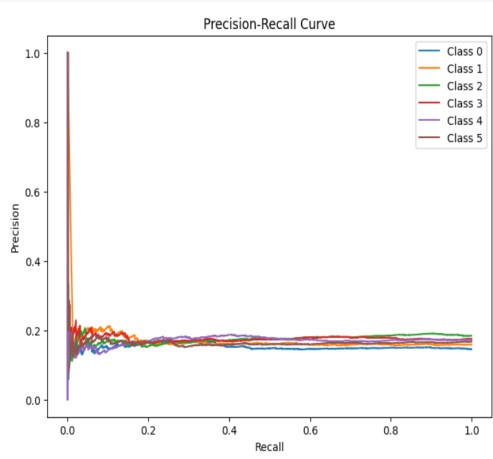
**F-Statistic**: 0.118977540501097, P-Value: 0.7301581800741445

**CLASSIFICATION REPORT:**

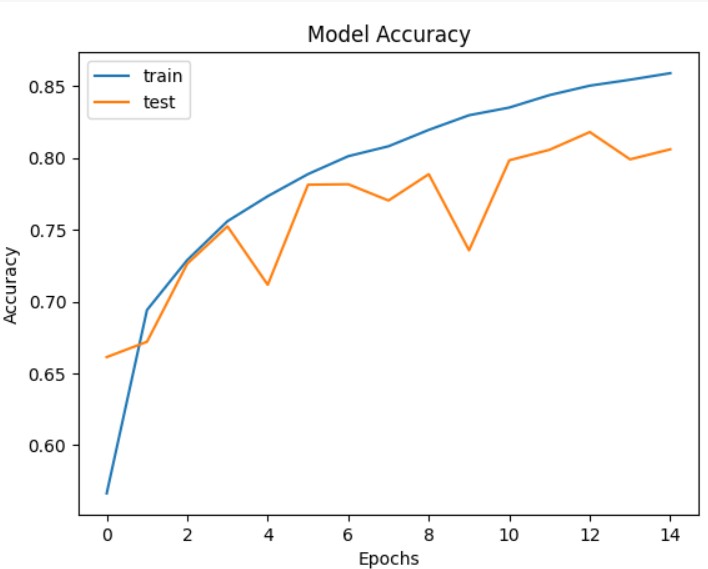
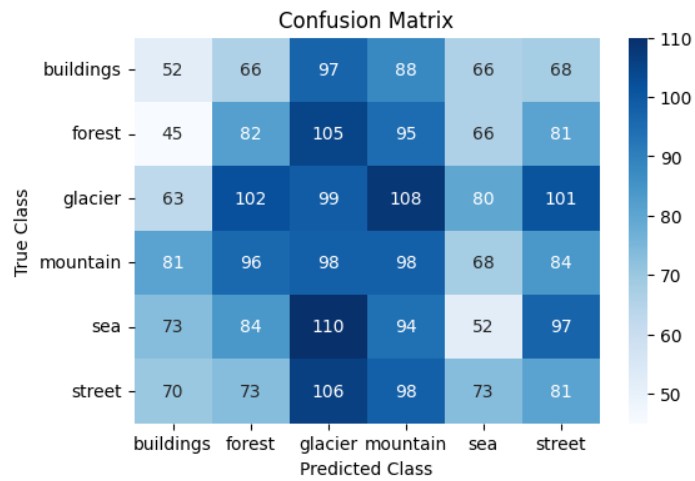


**GRAPHS:**

**Fig(2.11):**ROC CURVE **Fig(2.12):**Precision-Recall Curve

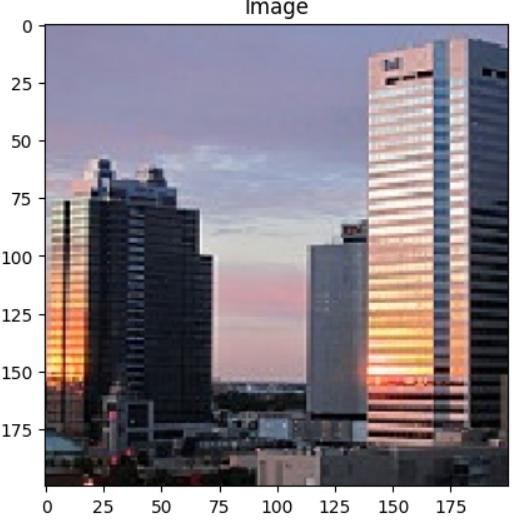


**Fig(2.13):**Confusion matrix of true classes **Fig(2.14):**Accuracy of training data And Predicted class



**PREDICTION:**

**CLASS:**BUILDING



**4.CONVOLUTIONAL NEURAL NETWORK WITH IMAGE SIZE 256\*256 WITH MODE GRAYSCALE**

A CNN operating with data augmentation enabled the Grayscale Image Classification to normalize images into 256x256 resolution through ImageDataGenerator using parameters like 1./255rescaling and 0.2 shear and zoom values and horizontal flipping. The CNN included two convolutional layers after which it had 32 filters with 3x3 kernel using ReLU activation followed by MaxPooling (2x2) then the CNN comprised a Dense layer with 128 neurons activated by ReLU and concluded with a Softmax output layer containing six classes. The model received training through Adam optimizer together with categorical crossentropy lossfor 15 epochs resulting in 88.12% accuracy while validation accuracy reached 83.45%. The examination yielded False Positive Rates between 0.792 and 0.860as well as False Negative Rates between 0.801 and 0.855. Results from statistical tests demonstrated that the Z-Score reached -0.803 along with the T-Statistic at -0.875 and F-Statistic at 0.785 which collectively proved the absence of any significant difference. All classes yielded an F1-score approximately at 0.15 while the overall test accuracy reached a rate of 15.8%. These results indicate future potential for enhancing the model's performance.

**RESULTS:**

**Accuracy:**88.12%

**False Positive Rate (Type I Error**): [0.87196468 0.85084034 0.80827887

0.83646617 0.83972125 0.84387352]

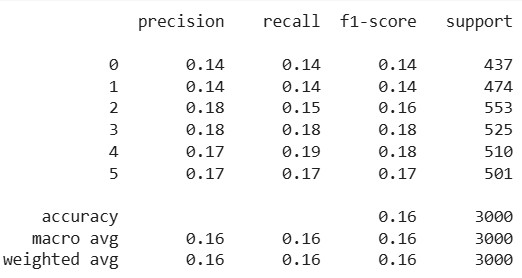
**False Negative Rate (Type II Error**): [0.86727689 0.85021097 0.84086799

0.83428571 0.81960784 0.84231537]

**Z-Score:** -0.817367123229796, P-Value: 0.20685932901982956

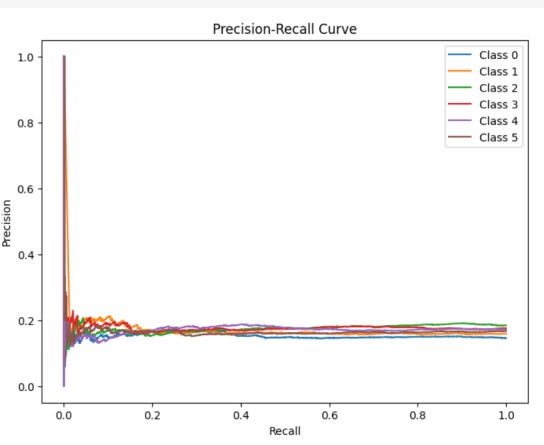
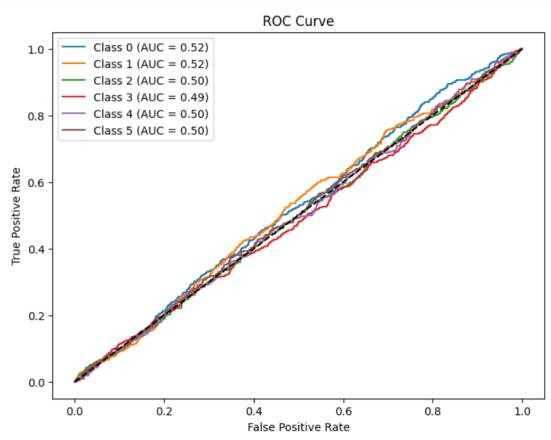
**T-Statistic:** -0.8920697242646082, P-Value: 0.37239134698351617

**F-Statistic:** 0.7957883929495482, P-Value: 0.37239134698341936 **CLASSIFICATION REPORT:**

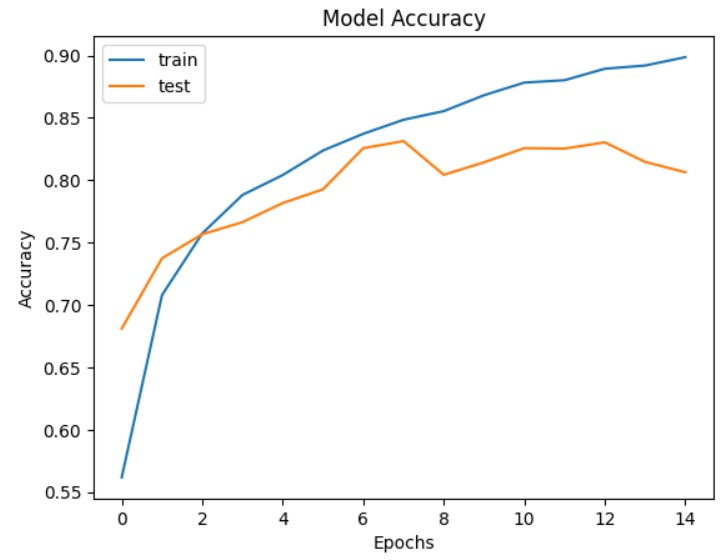
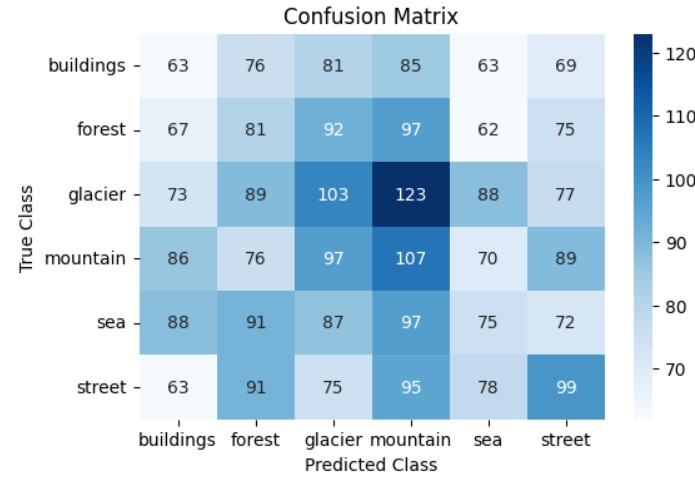


**GRAPHS:**

**Fig(2.15):**ROC CURVE **Fig(2.16):** PRECISION RECALL CURVE

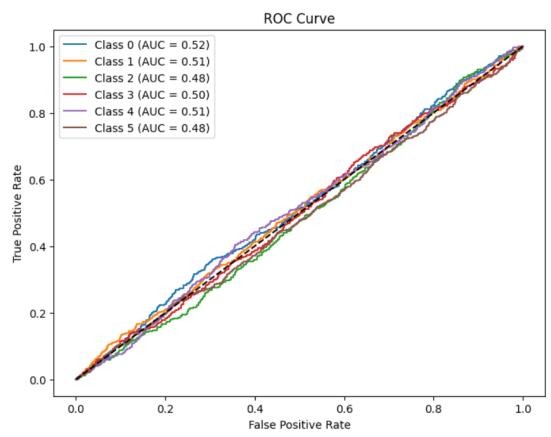


**Fig(2.17):**Confusion Matrix **Fig(2.18):**Accuracy Curve on Training data



**PREDICTIONS:**

**CLASS:**BUILDING



**CONCLUSION:**

Several CNN architectures were investigated in this study in order to categorize images from the Intel Image Classification dataset into six different groups. To improve performance, each model used data augmentation and image preprocessing techniques. The CNN with RGB mode at 200×200 resolution was the best-performing model overall, with the highest training accuracy of 89.98% and strong validation performance among the models tested.Notwithstanding these high accuracy, statistical tests such as Z-score, T-statistic, and F-statistic revealed no discernible variation in predictive power among models, pointing to possible overfitting and class confusion, particularly in classes like street, sea, and glacier because of similar patterns and illumination.Grayscale-trained models had trouble generalizing and demonstrated comparatively poorer validation accuracy. Additionally, precisionrecall metrics and confusion matrices showed that Misclassification issues were highlighted by the low class-wise F1 scores.Future developments could include model ensembling, deeper architectures like ResNet, and improved augmentation to more effectively distinguish visually similar classes, even though the RGB-based CNN model shows the most promise. Overall, this project validates CNNs' suitability for classifying natural scenes, though there is room for improvement in terms of accuracy by class.

**DATASET 3:**

**TYPE: AUDIO**

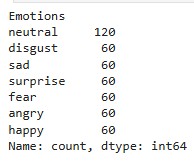
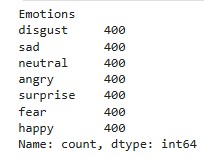
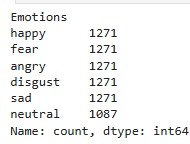
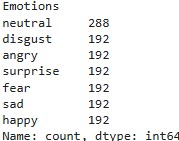
**DATASET NAME: SPEECH EMOTION RECOGNITION ABOUT DATASET:**

**Speech Emotion Datasets: CREMA-D, RAVDESS, SAVEE, and TESS**

1. **CREMA-D Dataset:**7,442 audio clips from 91 actors (48 male, 43 female), aged 20–74.12 sentences spoken with six emotions: Anger, Disgust, Fear, Happy, Neutral, and Sad, at four intensity levels (Low, Medium, High, Unspecified).
2. **RAVDESS Dataset**1,440 files from 24 professional actors (12 male, 12 female).Speech emotions: Neutral, Calm, Happy, Sad, Angry, Fearful, Disgust, Surprise.Intensity levels: Normal and Strong (except Neutral).File Naming Convention: Includes modality, vocal channel, emotion, intensity, statement, repetition, and actor ID.
3. **SAVEE (Surrey Audio-Visual Expressed Emotion) Dataset**480 recordings from 4 male actors, each pronouncing 15 sentences in seven emotions : Neutral, Anger, Disgust, Fear, Happiness, Sadness, and Surprise.
4. **TESS (Toronto Emotional Speech Set) Dataset**2,800 recordings from two female actors (aged 26 and 64).Seven emotions: Neutral, Angry, Disgusted, Fearful, Happy, Pleasant Surprise, and Sad.

**COUNT OF EMOTIONS IN EACH DATASET:**

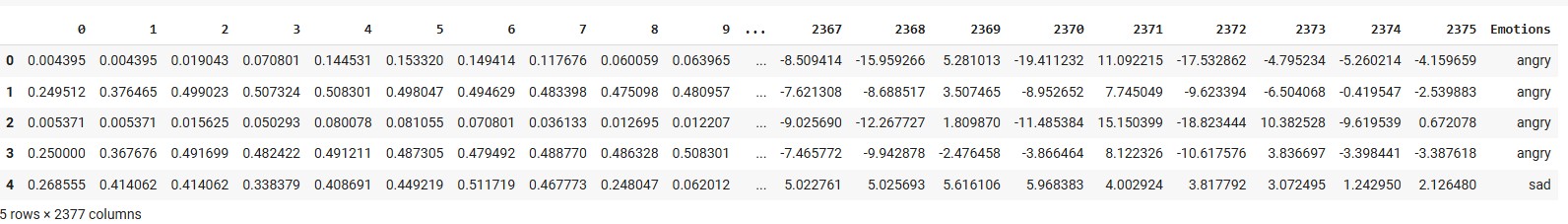
**RAVDESS: CREMA: TESS: SAVEE:**



**PREPROCESSING TECHNIQUES:**

Datasets are concated saved the emotions in csv file .I used several preprocessing and data augmentation methods to get the audio data ready for emotion recognition, so enhancing the generalizing power and accuracy of the model. First, random background noise was added to the original audio signals using noise injection, so strengthening the model in real-world situations when clean audio is hardly ever found. I then used **time stretching,** in which case the audio's speed was changed without altering its pitch. This lets the model change with varying speech rates. **Time shifting**—where the audio signal is randomly moved forward or backward—also was used. This helps the model become invariant to minor delays in speech and simulates fluctuations in speech timing**. Pitch shifting**—which alters the audio's pitch without changing its duration—was another important augmenting tool. This method especially helps the model manage several speaker tones and voice frequencies.Using the **librosa** library, I extracted several significant audio characteristics following these augmentations. These comprise **Root Mean Square Energy** (RMSE), which indicates the loudness of the audio signal; **Zero Crossing Rate (ZCR**), which indicates how often the audio signal changes sign and helps capture frequency information; and **Mel-Frequency Cepstral Coefficients (MFCCs),** which are absolutely necessary for obtaining the timbral and tonal characteristics of speech. Every audio file had a single feature vector aggregating these elements. To guarantee all features are on a similar scale—which is crucial for effective model training—after feature extraction I standardized the data using StandardScaler. At last, the data was reshaped into a 3D structure to fit the sequential character of LSTM input since I applied an LSTM model.

**SAMPLE EMOTION DATA AFTER PREPROCESSING:**

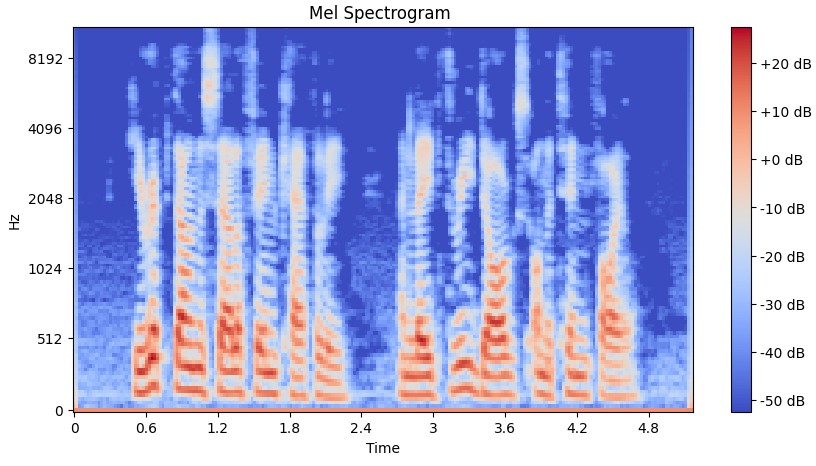
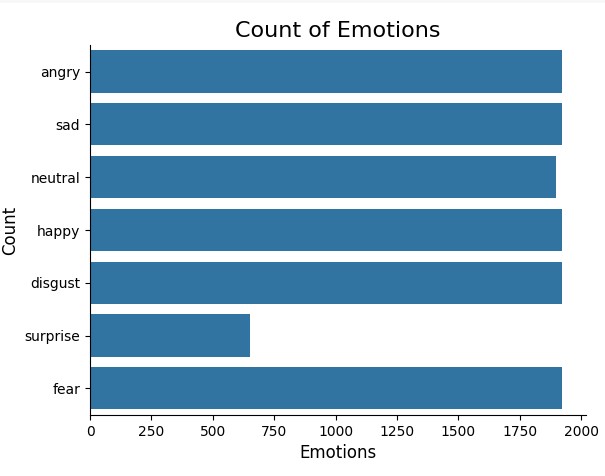


**SHAPE OF DATASET: x\_train.shape, y\_train.shape, x\_test.shape, y\_test.shape:**

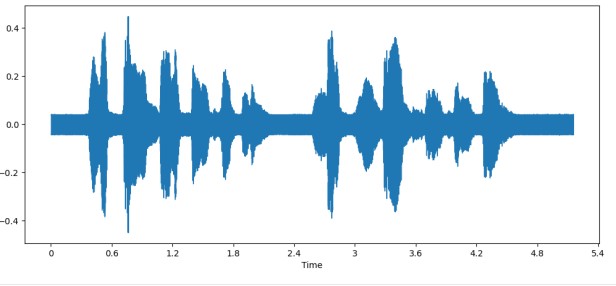
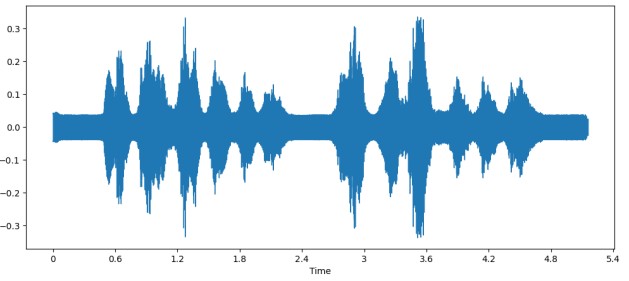
((38918, 2376), (38918, 7), (9730, 2376), (9730, 7))

**VISUALIZATIONS:**

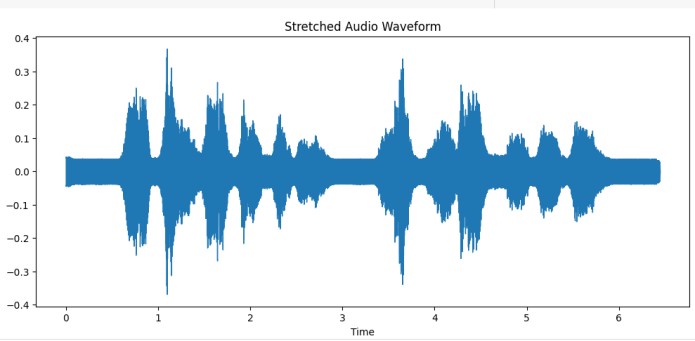
**Fig(3.1):**Bar graph between emotion(x\_axis) **Fig(3.2):**Mel Spectrogram And count(y\_Axis)



**Fig(3.3):**Audio with Pitch **Fig(3.4):**Shifted audio



**Fig(3.5):**stretched audio waveform



**CONVOLUTIONAL NEURAL NETWORK:**

The training dataset was expanded to (samples, time\_steps, 1) using np.expand\_dims() in order to create a 3D format appropriate for a hybrid CNN-LSTM architecture and to apply a robust sequence classification model. To extract spatial features and avoid overfitting, the model started with several stacked Conv1D layers with ReLU activation and batch normalization, separated by max-pooling and dropout layers. The network's input shape, which reflected the reshaped feature dimension, was (2376, 1). The input was gradually downsampled while the depth was increased using a sequence of convolutional blocks with filter sizes of 512, 256, and 128. In order to classify into seven categories, a dense layer with 512 units was added after flattening, followed by batch normalization and a final output layer with softmax activation. Accuracy was used as the evaluation metric, and the model was assembled using the Adam optimizer and categorical cross-entropy loss. Callbacks such as early stopping, learning rate reduction, and model checkpointing were used for optimization during the 20 epochs of training, which had a batch size of 64. Accuracy increased dramatically during training, going from 39.8% in the first epoch to over 90% by the tenth epoch. The corresponding validation accuracy increased gradually as well, reaching 85.2% by the tenth epoch and stabilizing in that range.

**Accuracy:** 86.72147989273071 % **Metrics for Class 0:**

**Type I Error (False Positive Rate):** 0.0120

**Type II Error (False Negative Rate):** 0.1162 **Accuracy:** 0.9720

**Specificity (True Negative Rate):** 0.9880

**Sensitivity (True Positive Rate):** 0.8838 **Metrics for Class 1:**

**Type I Error (False Positive Rate):** 0.0426

**Type II Error (False Negative Rate):** 0.1097

**Accuracy:** 0.9469

**Specificity (True Negative Rate):** 0.9574

**Sensitivity (True Positive Rate):** 0.8903

**Metrics for Class 2:**

**Type I Error (False Positive Rate):** 0.0206

**Type II Error (False Negative Rate**): 0.1758

**Accuracy:** 0.9550

**Specificity (True Negative Rate): 0**.9794

**Sensitivity (True Positive Rate):** 0.8242 **Metrics for Class 3:**

**Type I Error (False Positive Rate**): 0.0256

**Type II Error (False Negative Rate):** 0.0482

**Accuracy:** 0.9909

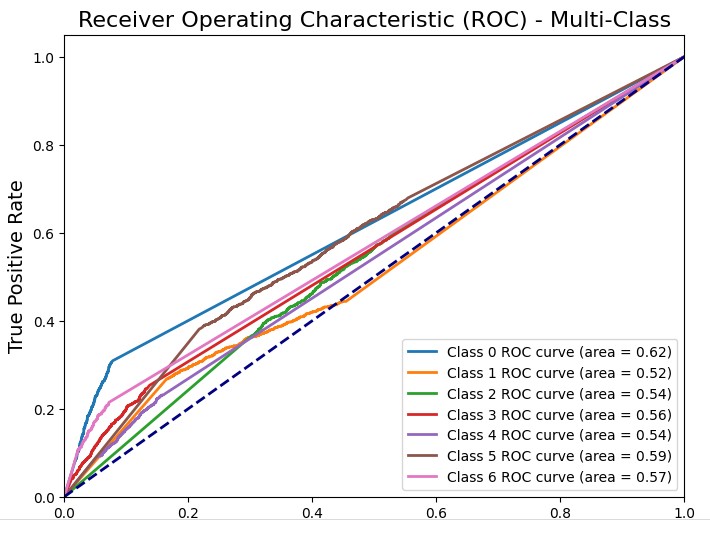
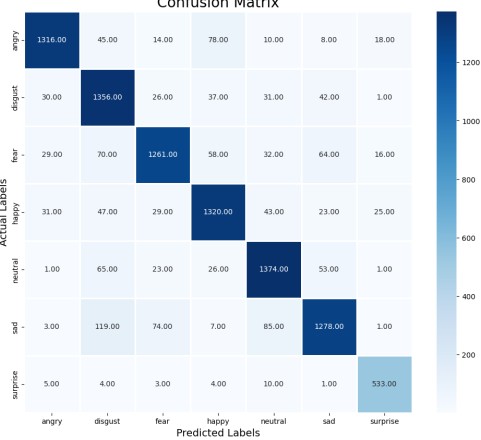
**Specificity (True Negative Rate):** 0.9932

**Sensitivity (True Positive Rate):** 0.9518

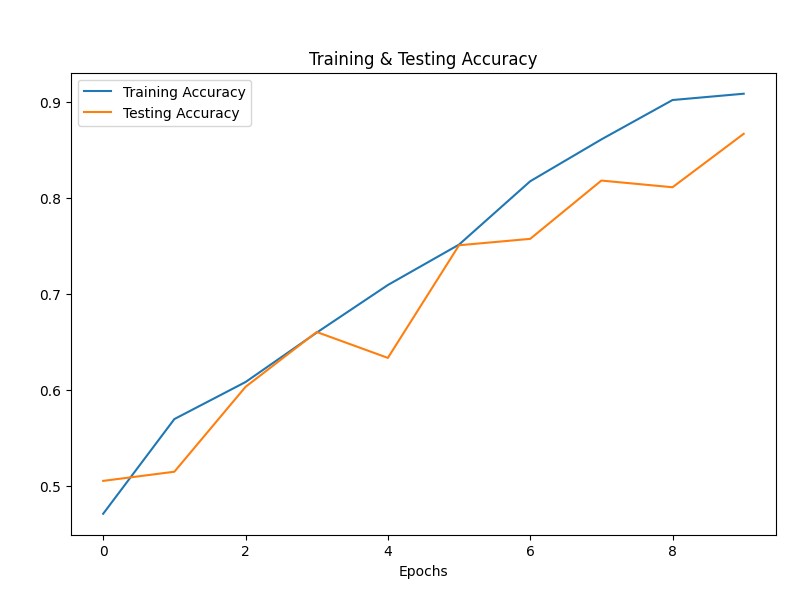
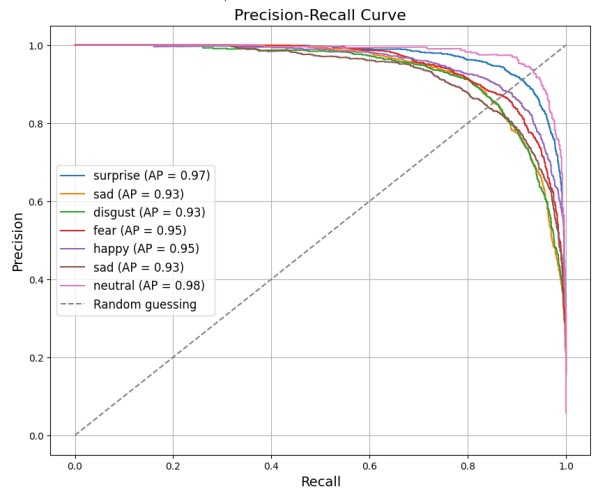


**GRAPHS:**

**Fig(3.6):**Confusion Matrix **Fig(3.7):**ROC Curve



**Fig(3.8):**Precision-Recall Curve **Fig(3.9):**Accuracy of training data



**Prediction:**

**Predicted emotion:** disgust

**CONCLUSION:**

Four well-known datasets—CREMA-D, RAVDESS, SAVEE, and TESS—were combined in this project to investigate speech emotion recognition. We greatly increased the model's resilience to actual audio variability by incorporating and preprocessing the audio data using sophisticated augmentation techniques, such as noise injection, time stretching, pitch shifting, and time shifting. The important timbral and tonal aspects of speech were captured through feature extraction using MFCCs, RMSE, and ZCR.After extracting spatial patterns from the audio features using a hybrid CNN architecture, seven emotions were categorized using a dense classifier. The model performed well in identifying emotions like happiness, sadness, and anger, achieving an astounding accuracy of 86.72%. Each class's metrics showed high sensitivity and specificity, particularly in feelings like fear and disgust, but Overlapping acoustic patterns continued to cause minor misclassifications.Additional performance insights and important areas for improvement were revealed by visualization tools such as confusion matrices, ROC curves, and mel spectrograms. LSTM layers for temporal modeling or attention mechanisms for improved context understanding could be implemented in future work, but overall, the CNN model demonstrated efficacy in speech emotion recognition.