**The conclusions and results of this code are as follows:**

1. Logistic Regression: The code is implementing one-vs-rest logistic regression (multi\_class='ovr') parameter in LogisticRegression()

It treats each class (0-9) as a separate binary classification problem and trains a separate logistic regression model for each class.

1. Confusion Matrix: The code prints the confusion matrix, which shows the number of true positives, true negatives, false positives, and false negatives for each class. It provides insights into the model's performance in terms of correctly and incorrectly classified instances for each class.

Confusion Matrix:

[[54 0 0 0 0 2 2 0 1 0]

[ 0 48 3 2 2 2 4 3 1 1]

[ 0 2 45 1 0 0 2 0 2 0]

[ 0 0 2 43 0 1 0 0 1 1]

[ 0 3 0 0 51 1 0 1 5 3]

[ 0 0 0 2 0 58 0 0 2 9]

[ 0 1 0 0 0 0 51 0 2 1]

[ 0 0 0 2 3 0 0 49 4 2]

[ 0 8 2 3 0 1 5 1 39 1]

[ 3 2 2 2 0 2 1 0 2 44]]

1. Classification Report: The code also prints the classification report, which includes precision, recall, F1-score, and support for each class. It provides a comprehensive evaluation of the model's performance, including accuracy metrics for individual classes and overall performance.

Precision is calculated as TP / (TP + FP)

Recall is calculated as TP / (TP + FN)

F1-score is calculated as 2 \* (precision \* recall) / (precision + recall).

Accuracy calculated as (TP + TN) / (TP + TN + FP + FN)

Classification Report:

precision recall f1-score support

0.0 0.95 0.92 0.93 59

1.0 0.75 0.73 0.74 66

2.0 0.83 0.87 0.85 52

3.0 0.78 0.90 0.83 48

4.0 0.91 0.80 0.85 64

5.0 0.87 0.82 0.84 71

6.0 0.78 0.93 0.85 55

7.0 0.91 0.82 0.86 60

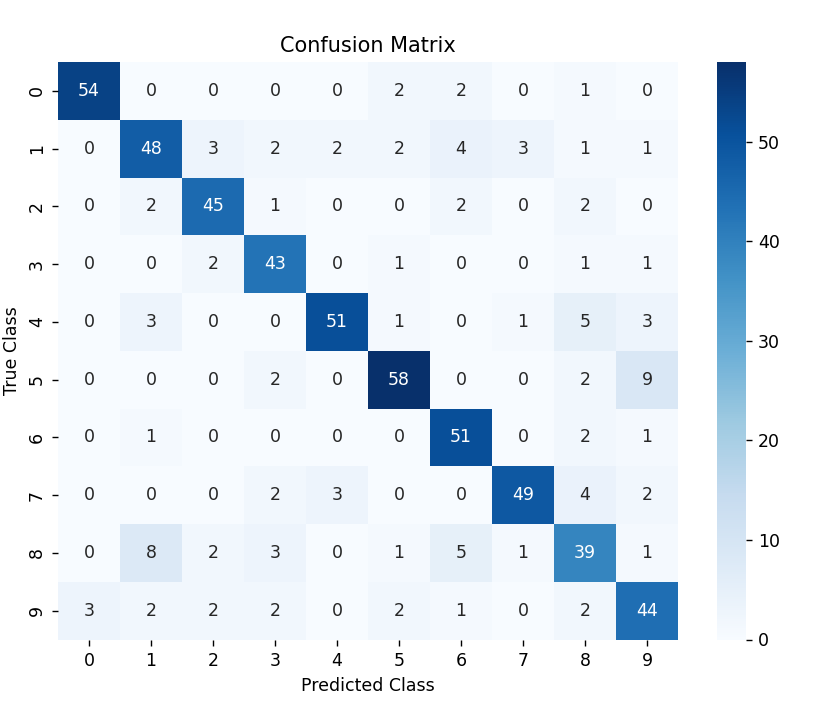
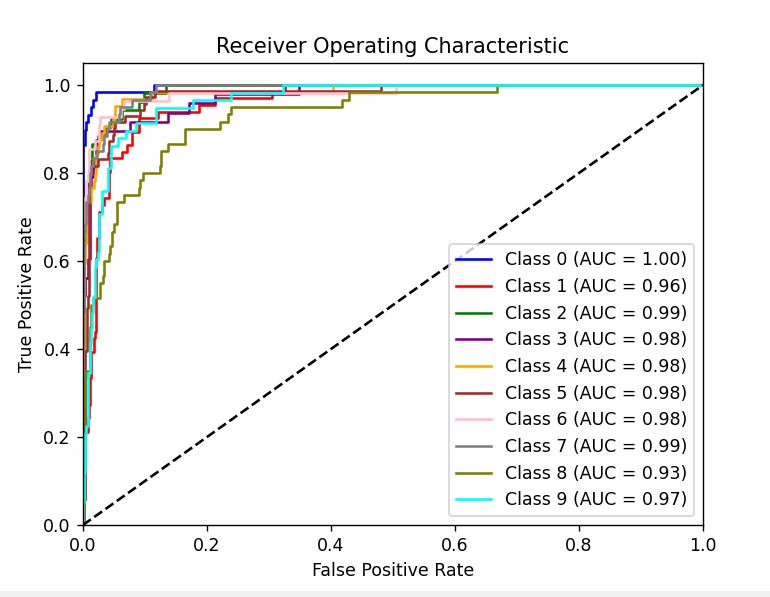
8.0 0.66 0.65 0.66 60

9.0 0.71 0.76 0.73 58

accuracy 0.81 593

macro avg 0.82 0.82 0.81 593

weighted avg 0.82 0.81 0.81 593

1. Confusion Matrix Visualization: The code visualizes the confusion matrix using a heatmap generated by seaborn. The heatmap provides a color-coded representation of the confusion matrix, making it easier to identify patterns and discrepancies in the model's predictions.
2. ROC Curve: The code calculates the probabilities and constructs the ROC curve for each class. The ROC curve illustrates the trade-off between true positive rate (sensitivity) and false positive rate (1 - specificity) at different classification thresholds. The AUC (Area Under the Curve) is also calculated for each class, indicating the overall performance of the model in distinguishing between positive and negative instances.