Viva Questions and Answers (with Code Questions)

Q1: What is classification in machine learning?

A: Classification is a supervised learning technique where the model learns to predict discrete labels or categories based on input features.

Q2: Which algorithm did you use in this practical?

A: I used Logistic Regression, which is a widely used algorithm for binary and multiclass classification problems.

Q3: What is Logistic Regression?

A: Logistic Regression is a statistical method used for binary classification. It uses the logistic (sigmoid) function to model the probability of belonging to a particular class.

Q4: How does Logistic Regression differ from Linear Regression?

A: Linear Regression predicts continuous values, while Logistic Regression predicts probabilities and class labels.

Q5: Why do we standardize the data before applying Logistic Regression?

A: Standardization ensures that features have a mean of zero and unit variance, which helps the model converge faster and perform better.

Q6: What is a confusion matrix?

A: A confusion matrix is a table used to evaluate the performance of a classification algorithm by comparing actual vs. predicted labels.

Q7: What are precision, recall, and F1-score?

A: Precision is the ratio of true positives to total predicted positives, recall is the ratio of true positives to total actual positives, and F1-score is the harmonic mean of precision and recall.

Q8: What does 'random_state' do in train_test_split?

A: It ensures reproducibility by controlling the shuffling of data before splitting into train and test sets.

Q9: Can Logistic Regression be used for multiclass classification?

A: Yes, Logistic Regression can handle multiclass classification using techniques like One-vs-Rest (OvR) or Softmax regression.

Q10: What is overfitting and how can it be avoided?

A: Overfitting occurs when a model learns noise in the training data. It can be avoided using techniques like cross-validation, regularization, and using more data.

Q11: Why do we use train_test_split in this code?

A: train_test_split is used to split the dataset into training and testing sets so that we can train the model on one set and evaluate it on unseen data.

Q12: What is StandardScaler doing in this code?

A: StandardScaler standardizes features by removing the mean and scaling to unit variance, which helps models like Logistic Regression perform better.

Q13: What does model.fit() do in this code?

A: model.fit() trains the Logistic Regression model using the training data.

Q14: How are predictions made in this code?

A: Predictions are made using model.predict() on the scaled test data.

Q15: Why do we check y_train.astype(int) in this code?

A: We convert y_train to integers to ensure the target labels are discrete classes, as Logistic Regression expects categorical targets.

Q16: How is the confusion matrix generated in this code?

A: The confusion matrix is generated using confusion_matrix(y_test, y_pred) after making predictions on the test data.

Q17: What happens if we skip scaling before applying Logistic Regression?

A: The model might perform poorly or converge slowly because features with different scales can bias the learning process.

Q18: What is the role of random_state=42 in this code?

A: random_state ensures that the train-test split and synthetic data generation are reproducible every time the code runs.

Q19: Can we change the test_size parameter? What will happen?

A: Yes, changing test_size changes the proportion of data used for testing. A higher value gives more test data but less training data.

Q20: Why do we use plt.scatter() in Step 8?

A: plt.scatter() is used to visualize the classification results by plotting test data points colored by their predicted class.