LDA Assignment

support@intellipaat.com

+91-7022374614

US: 1-800-216-8930 (Toll-Free)

Problem Statement:

You have been provided with a multi dimensional data that contains information on certain images. Using machine learning, you should be able to predict the images on the new set of data using the model that you have trained on the existing data.

Dataset Information:

Each point in the data is an 8x8 image.

Classes	10
Samples per class	~180
Samples total	1797
Dimensionality	64
Features	integers 0-16

Note: Load the dataset from sklearn.datasets.load_digits()

Q1. What will be the output of the following code?

```
from sklearn import dataset

digits = datasets.load_digits()
```

1. Digits data from the sklearn module
2. Import error
3. Value error
4. Digits data in a pandas dataframe

Q2. If we split the data in a ratio of 80% training and 20% testing data, what will be the correct code for the same?

1. X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=20, random_state=42)
2. X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=80, random_state=42)
3. X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
4. X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=80:20, random_state=42)

Q3. In the train_test_split, if we keep the random_state = 1, what does it mean for our training and testing data?

1. Everytime the new random values are generated in the test and train sets
2. The values will be the same every time the code is executed in the testing and training sets.
3. None of the Above

4. Both 1 and 2.

Q4. In the code below, where we standardize the data, we have used the fit_transform() for the training sample, and transform() for the testing sample, why?

```
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

1. We use the same mean and variance calculated on the training data to fit the test data
2. The methods distinguish between the variance of each class
3. The methods distinguish between mean and standard deviation of each class.
4. None of the above

Q5. Find the mistake in the code below?

```
lda = LinearDiscriminantAnalysis(n_components=9)

X_train = lda.fit_transform(X_train)

X_test = lda.transform(X_test)
```

1. Fit_transform() must include the y_train.
2. transform() must include the y_train.
3. None of the above
4. Both 1 and 2

Q6. What is the shape of the data after standardizing the training and testing data?

1. (1437,64)
2. (1797,64)
3. (1437,9)
4. (1437,)

Q7. What is the mistake in the code below?

```
lda = LinearDiscriminantAnalysis(n_components=9)

X_train = lda.fit_transform(X_train, X_test)

X_test = lda.transform(X_test)
```

1. X_test instead of y_train in fit_transform()
2. X_test instead of y_train in transform()
3. n_components = 9 is incorrect
4. X_test instead of y_test in transform()

Q8. How do you decide the n_components in the LinearDiscriminantAnalaysis()?

1. Correlation coefficient
2. variation inflation factor

3. explained_variance_ratio
4. None of the above

Q9. If we keep the n_components as 15 in the LDA, what will be the shape of the data?

1. (1797,15)
2. (15,15)
3. (15,)
4. (1437, 15)

Q10. After performing LDA on the standardized data, with n_components= 9, Create a random forest classifier to fit the new data with n_estimators= 100, and random_state same as used in train_test_split. After the above operation, what will be the accuracy score of the model?

1. 0.75
2. 0.85
3. 0.95
4. 0.98

Q11. Identify the mistake in the code below.

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(n_estimators=100, random_state=42)

rf.fit(X_train, X_test)
```

1. X_test instead of y_test
2. X_test instead of y_train
3. X_test instead of n_components = 9
4. Missing parameter - random_state=42

Q12. What percentage of positive cases was the model able to catch for class 6?

1. 100
2. 97
3. 35
4. 99

Q13. What percentage of the predictions were true for class 5?

1. 96
2. 47
3. 98
4. 0.98

14. What percentage of positive predictions were correct for class 3?

1. 34
2. 92
3. 97

4. 94