Machine Learning

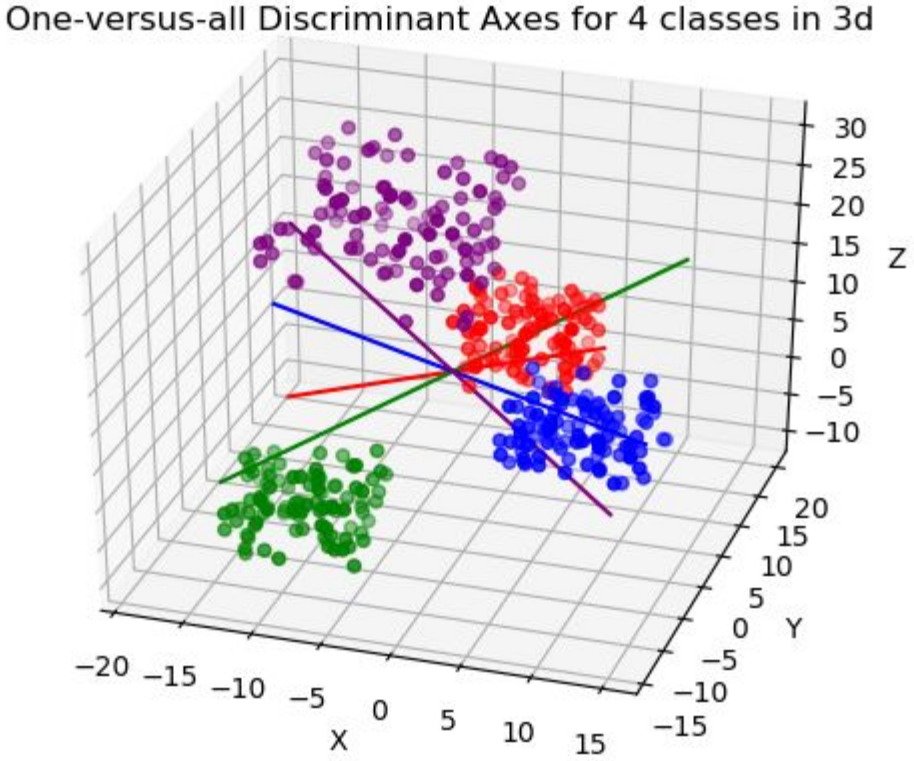
By EvolkAI

Model Name: Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a statistical algorithm used for classification tasks. It aims to find linear decision boundaries to separate different classes of data.

In simple terms, LDA works by calculating the mean and covariance of each class from the training data. It then uses this information to find a line (or hyperplane in higher dimensions) that maximally separates the classes while minimizing the within-class variability.

When presented with new data, LDA projects it onto this line or hyperplane and assigns it to the class with the closest projected value. LDA assumes that the data follows a Gaussian distribution and that the classes have equal covariance.LDA is a popular algorithm for classification because it is simple, computationally efficient, and performs well when the classes are well-separated. It is widely used in various fields, such as pattern recognition, image classification, and bioinformatics.


One-versus-all Discriminant Axes for 4 classes in 3d
30
25
20
15
10
5
0
−5
−10
Z
20
15
10
5
0
−5
−10
−15
Y
−20
−15
−10
−5
0
5
10
15
X

```
import pandas as pd
data = pd.read_csv('iris_dataset.csv')
data.info()
```

In First line we Import pandas library as pd, then we read iris_dataset.csv file using the read_csv() function, and prints information about the data using the info() method.

```
feature = ['sepal_length', 'sepal_width', 'petal_length', 'petal_width']
predection_class = ['species']
X = data[feature].values
y = data[predection_class].values
```

This defines the ‘feature’ and ‘predection_class’ variables, which specify the columns of the data to use as ‘features’ and the column to use as the prediction target.
The code then creates ‘X’ and ‘y’ arrays containing the values of these columns from the data DataFrame.

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X,y,test_size=0.30)
```

This imports the train_test_split function from the sklearn.model_selection module and uses it to split the data into training and testing sets. The test_size parameter specifies that 30% of the data should be used for testing.

```
print(f"Shape of X_test is {X_test.shape}")
print(f"Shape of X_train is {X_train.shape}")
print(f"Shape of Y_test is {Y_test.shape}")
print(f"Shape of Y_train is {Y_train.shape}")
```

These lines print the shapes of the training and testing data arrays. This output totally depends on the test size we took while train_test_split.

OUTPUT

```
Shape of X_test is (45, 4)
Shape of X_train is (105, 4)
Shape of Y_test is (45, 1)
Shape of Y_train is (105, 1)
```

```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
clf = LinearDiscriminantAnalysis()
clf.fit(X_train, Y_train)
Y_pred = clf.predict(X_test)
```

This imports the *LinearDiscriminantAnalysis* module from the sklearn discriminant_analysis library, creates a *LinearDiscriminantAnalysis* classifier object, fits the classifier with the training data using the *fit()* method, and use the *predict()* method to generate predictions for the testing data.

```
from sklearn import metrics
print("Accuracy", metrics.accuracy_score(Y_test, Y_pred)*100)
```

This imports the metrics module from sklearn and uses the accuracy_score() function to calculate the accuracy of the model on the testing data. The result is printed in the console.

OUTPUT

```
Accuracy 97.77777777777777
```

Conclusion

In conclusion, Linear Discriminant Analysis (LDA) is a widely used and effective statistical algorithm for classification tasks. By finding optimal linear decision boundaries, LDA can successfully separate different classes of data. Its simplicity, computational efficiency, and assumption of equal class covariances make it a popular choice. LDA performs well when the classes are well-separated, making it suitable for various applications such as pattern recognition and image classification.

While LDA assumes a Gaussian distribution and linear decision boundaries, it remains a valuable tool in situations where these assumptions hold true. Its straightforward approach and reliable performance make Linear Discriminant Analysis a practical and accessible algorithm for classification problems.

Thank You