Sci-kit learn

1. from sklearn.model_selection import train_test_split # Import train_test_split
2. X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42) # Split data into training and testing sets
3. from sklearn.preprocessing import StandardScaler # Import StandardScaler
4. scaler = StandardScaler() # Create a StandardScaler object
5. X_train_scaled = scaler.fit_transform(X_train) # Fit and transform the training data
6. X_test_scaled = scaler.transform(X_test) # Transform the testing data
7. from sklearn.linear_model import LinearRegression # Import LinearRegression
8. model = LinearRegression() # Create a LinearRegression model
9. model.fit(X_train, y_train) # Fit the model to the training data
10. y_pred = model.predict(X_test) # Predict on the test data
11. from sklearn.metrics import mean_squared_error # Import mean_squared_error
12. mse = mean_squared_error(y_test, y_pred) # Calculate mean squared error
13. from sklearn.metrics import accuracy_score # Import accuracy_score
14. accuracy = accuracy_score(y_test, y_pred) # Calculate accuracy score
15. from sklearn.ensemble import RandomForestClassifier # Import RandomForestClassifier
16. clf = RandomForestClassifier() # Create a RandomForestClassifier model
17. clf.fit(X_train, y_train) # Fit the classifier to the training data
18. y_pred_class = clf.predict(X_test) # Predict on the test data
19. from sklearn.decomposition import PCA # Import PCA
20. pca = PCA(n_components=2) # Create a PCA object with 2 components

21. X_pca = pca.fit_transform(X) # Fit and transform the data with PCA
22. from sklearn.cluster import KMeans # Import KMeans
23. kmeans = KMeans(n_clusters=3) # Create a KMeans object with 3 clusters
24. y_kmeans = kmeans.fit_predict(X) # Fit and predict clusters