

```
1 from sklearn import datasets
2 from sklearn.linear_model import LinearRegression
3 from sklearn.metrics import mean_squared_error
4 from sklearn.model_selection import train_test_split
5 from sklearn.ensemble import RandomForestRegressor
6 from sklearn.linear_model import LogisticRegression
7 from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
8 from sklearn.naive_bayes import GaussianNB
9 from sklearn.neighbors import KNeighborsClassifier
10
11 iris=datasets.load_iris()
12 x_train,x_test,y_train,y_test=train_test_split(iris.data,iris.target,random_state=42)
13 # model=LinearRegression()
14 # model=RandomForestRegressor(n_estimators=100)
15 # model=LogisticRegression()
16 model=KNeighborsClassifier(n_neighbors=3)
17 # model=GaussianNB()
18 model.fit(x_train,y_train)
19 y_pred=model.predict(x_test)
20 # mse=mean_squared_error(y_test,y_pred)
21 # print(mse)
22 ax=accuracy_score(y_test,y_pred)
23 cf=confusion_matrix(y_test,y_pred)
24 cr=classification_report(y_test,y_pred)
25 cl=float(input("Enter cl:"))
26 cw=float(input("Enter cw:"))
27 pl=float(input("Enter pl:"))
28 pw=float(input("Enter pw:"))
29 prdcls=model.predict([[cl,cw,pl,pw]])
30 prdsp=iris.target_names[prdcls]
31 print(prdsp)
32 print(f"accuracy:{ax}")
33 print(f"cf:{cf}")
34 print(f"cr:{cr}")
```

```
1 from sklearn import datasets
2 from sklearn.linear_model import LinearRegression
3 from sklearn.metrics import mean_squared_error
4 from sklearn.model_selection import train_test_split
5 from sklearn.ensemble import RandomForestRegressor
6 from sklearn.linear_model import LogisticRegression
7 from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
8 from sklearn.naive_bayes import GaussianNB
9 from sklearn.neighbors import KNeighborsClassifier
10 from sklearn.tree import DecisionTreeClassifier
11
12 iris=datasets.load_iris()
```

```
13 x_train,x_test,y_train,y_test=train_test_split(iris.data,iris.targ
14 # model=LinearRegression()
15 # model=RandomForestRegressor(n_estimators=100)
16 # model=LogisticRegression()
17 # model=KNeighborsClassifier(n_neighbors=3)
18 # model=GaussianNB()
19 model=DecisionTreeClassifier()
20 model.fit(x_train,y_train)
21 y_pred=model.predict(x_test)
22 # mse=mean_squared_error(y_test,y_pred)
23 # print(mse)
24 ax=accuracy_score(y_test,y_pred)
25 cf=confusion_matrix(y_test,y_pred)
26 cr=classification_report(y_test,y_pred)
27 cl=float(input("Enter cl:"))
28 cw=float(input("Enter cw:"))
29 pl=float(input("Enter pl:"))
30 pw=float(input("Enter pw:"))
31 prdcls=model.predict([[cl,cw,pl,pw]])
32 prdsp=iris.target_names[prdcls]
33 print(prdsp)
34 print(f"accuracy:{ax}")
35 print(f"cf:{cf}")
36 print(f"cr:{cr}")
```