week3 about:srcdoc

```
In [1]: from sklearn.datasets import load_iris
        from sklearn.model_selection import train_test_split
        from sklearn.linear_model import LogisticRegression
        from sklearn.neighbors import KNeighborsClassifier
        from sklearn.metrics import accuracy_score, classification_report
        iris =
        load_iris() X =
        iris.data y =
        iris.target
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_
        lr =
        LogisticRegression(max_iter=200)
        lr.fit(X_train, y_train) lr_pred =
        lr.predict(X_test)
        print("Logistic Regression:")
        print("Accuracy:", accuracy_score(y_test, lr_pred))
        print(classification_report(y_test, lr_pred))
        knn =
        KNeighborsClassifier(n_neighbors=3)
        knn.fit(X_train, y_train) knn_pred =
        knn.predict(X_test)
        print("\nK-Nearest Neighbors:")
        print("Accuracy:", accuracy_score(y_test, knn_pred))
        print(classification_report(y_test, knn_pred))
```

```
Logistic Regression:
     precision recall f1-score
     support
            1.00
                    1.00
                            1.00
                                      16
            1.00
                    0.94
                            0.97
                                      18
            2
                  0.92
                           1.00
                                   0.96
                                             11
         accuracy
                                      0.98
         macro avg
                       0.97
                                0.98
                                        0.98
     45 weighted avg
                        0.98
                                0.98
                                        0.98
     K-Nearest Neighbors:
     precision recall f1-score
     support
     0
            1.00
                    1.00
                            1.00
                                      16
                    0.94
            1.00
                            0.97
                                      18
                  0.92
                           1.00
                                   0.96
                                             11
         accuracy
                                      0.98
          macro avg
                        0.97
                                0.98
                                        0.98
     45 weighted avg
                        0.98
                                0.98
                                        0.98
In [ ]:
```

1 of 1

a open-source dataset based on accuracy, precision, recall and F1-score.

procedure:

1) Import the required libraries.

2) Load the Pris dataset (builth).

3.) split the dataset hito training and test sets.

4) Traily two classifiers:

- Logistic Regression

- K- Nearest Meighbours

- s) use both models to make predictions.
- 6.) Evaluate each model using:

→ Accuracy → classification report.

7) print and compare the results.

code:

from skleam. datasets import load-ing.

from skleam. model-selection import train-test-split.

from skleam. librear-model import Logistic Regression

from skleam. neighbors import kaleighbors classifier

from sk leam. metrics import accuracy-score, classification

-report

ms = (oad-msc)

x = Pris-data

Y = Ms. target

x-train, x-test, y-train, y-test = train-test-split (x, y, test-size=0.3, random-state=0)

delinged strong doors (4

lr = Logistic Regression (max_iter = 200)
lr. fit (x-train, y-train)
lr-pred = Ir. predict (x-test)

Print ("Logistic Rogression:"),

print ("Accuracy;", accuracy_score (Y thit, br-pred))

print (classification_report (Y test, br-pred))

KMM = KNeighborsclassifier (n_neighbors=3)

kMM dit (X-train, Y-train)

Khn- pred = Khn. predict (x test)

print (" In k- Mearest Meighbors:")

print ("Accuracy: " accuracy_score (y-test, knn pres))
print (classification_report (y-test, knn-pred))

output :

Logistic regression:

Accuracy = 0.977

		and the			
		precision	recall	F1-score	support
	0	1.00	1.00	1.00	t.e.
	1	4.00	0.94	6,97	18
	7	0 . 92	1.00	0,96	11
accuracy				89.9	45
macro arg		0.97	0.98	0.48	45

K-Nearest Neighbors :

Accuracy : 0.9777

		precision	reall	FI-Score	support
	D	1.00	1.60	1.00	16
	1	1.00	0.94	6.45	18
	2	0-9 0002	1.00	0.46	u
accuracy				89.0	45
macro avg		0.97	0,98	0.98	45

Total positive of Total Megative Accuracy = Totalesave TP (How many are actual possible precision = THE INTER SOUS CLOSE STORE STATE OF TP (How many actual positive Recall = teter that the term of precision and Recall. Periodelian Propinty of 10 the FI Score = result: In this experiment I have successfully imple -mented the companing of different classification, algorithms using open source dataset. Ladistic Regression FEB. O - MOCHION = 2001/101/1 - 1200/1 - 4 PEELO : POLINO, Y CON P-3