

### AIM

To implement classifier using an open source dataset.

### OBJECTIVE:

⇒ To understand the fundamentals of supervised classifier.

⇒ To train and test a classifier using data.

⇒ To evaluate model performance using accuracy.

⇒ To visualize predictions and interpret results.

### PSEUDOCODE

1. Import necessary libraries
2. Load dataset from sklearn - (digits)
3. Split the dataset into features ( $x$ ) and labels ( $y$ )
4. Divide the data into training and testing sets.
5. Initialize KNN (K-Nearest Neighbours) classifier.
6. Training the classifier using training data.
7. Predict the labels for test data.
8. Evaluate the performance using accuracy and classification report
9. Visualize a sample predictions.

## OBSERVATION

the classifier trained successfully on the dataset.

It achieved high accuracy in classifying unseen data.

Visual inspection showed that predict values matched the actual labels in most cases.

## RESULT

~~Implement~~ classifier using an open source dataset.

10.1.38.19/user/ra2311047010011/lab/workspaces/auto-b/tree/DEEP%20LEARNING/L2.ipynb

Verify that it's you

File Edit View Run Kernel Tabs Settings Help

Filter files by name

/ DEEP LEARNING /

Name	Last Modified
Breast_canc...	next year
Exp2.ipynb	next year
L1.1.ipynb	next year
L1.ipynb	next year
L2.ipynb	next year
L3.ipynb	next year
Untitled.ip...	next year
Untitled1.jp...	6 days ago

L1.ipynb L3.ipynb L1.1.ipynb L2.ipynb

Notebook Python 3 (ipykernel)

```
[1]: from sklearn.datasets import load_digits

[2]: d=load_digits()

[34]: x=d.data
      y=d.target

[35]: x

[35]: array([[ 0.,  0.,  5., ...,  0.,  0.,  0.],
            [ 0.,  0.,  0., ..., 10.,  0.,  0.],
            [ 0.,  0.,  0., ..., 16.,  9.,  0.],
            ...,
            [ 0.,  0.,  1., ...,  6.,  0.,  0.],
            [ 0.,  0.,  2., ..., 12.,  0.,  0.],
            [ 0.,  0., 10., ..., 12.,  1.,  0.]], shape=(1797, 64))

[36]: y

[36]: array([0, 1, 2, ..., 8, 9, 8], shape=(1797,))

[37]: from sklearn.model_selection import train_test_split

[38]: x_train,x_test,y_train,y_test=train_test_split(x,y, test_size=0.2,random_state=42)

[39]: from sklearn.neighbors import KNeighborsClassifier

[40]: knn = KNeighborsClassifier()

[41]: from sklearn.linear_model import LogisticRegression

[42]: clf = LogisticRegression()

[20]: clf.fit(x_train, y_train)
```



← → ↺

Not secure 10.1.38.19/user/ra2311047010011/lab/workspaces/auto-b/tree/DEEP%20LEARNING/L2.ipynb

🔍 ☆ ⬇️ 🔗 Verify that it's you

File Edit View Run Kernel Tabs Settings Help

+

📁

📄

🔄

Filter files by name 🔍

/ DEEP LEARNING /

Name	Last Modified
Breast_canc...	next year
Exp2.ipynb	next year
L1.1.ipynb	next year
L1.ipynb	next year
L2.ipynb	next year
L3.ipynb	next year
Untitled.ip...	next year
Untitled1.ip...	6 days ago

L1.ipynb L3.ipynb L1.1.ipynb L2.ipynb +

Notebook Python 3 (ipykernel)

increase the number of iterations to improve the convergence (max\_iter=100).  
You might also want to scale the data as shown in:  
<https://scikit-learn.org/stable/modules/preprocessing.html>  
Please also refer to the documentation for alternative solver options:  
[https://scikit-learn.org/stable/modules/linear\\_model.html#logistic-regression](https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)  
n\_iter\_i = \_check\_optimize\_result

[20]: LogisticRegression  
Parameters

[21]: y\_pred=clf.predict(x\_test)  
y\_pred

[21]: array([[6, 9, 3, 7, 2, 1, 5, 2, 5, 2, 1, 9, 4, 0, 4, 2, 3, 7, 8, 8, 4, 3,  
9, 7, 5, 6, 3, 5, 6, 3, 4, 9, 1, 4, 4, 6, 9, 4, 7, 6, 6, 9, 1, 3,  
6, 1, 3, 0, 6, 5, 5, 1, 3, 5, 6, 0, 9, 0, 0, 1, 0, 4, 5, 2, 4, 5,  
7, 0, 7, 5, 9, 5, 5, 4, 7, 0, 4, 5, 5, 9, 9, 0, 2, 3, 8, 0, 6, 4,  
4, 9, 1, 2, 8, 3, 5, 2, 9, 0, 4, 4, 4, 3, 5, 3, 1, 3, 5, 9, 4, 2,  
7, 7, 4, 4, 1, 9, 2, 7, 8, 7, 2, 6, 9, 4, 0, 7, 2, 7, 5, 8, 7, 5,  
7, 5, 0, 6, 6, 4, 2, 8, 0, 9, 4, 6, 9, 9, 6, 9, 0, 5, 5, 6, 6, 0,  
6, 4, 3, 9, 3, 8, 7, 2, 9, 0, 6, 5, 3, 6, 5, 9, 9, 8, 4, 2, 1, 3,  
7, 7, 2, 2, 3, 9, 8, 0, 3, 2, 2, 5, 6, 9, 9, 4, 1, 2, 4, 2, 3, 6,  
4, 8, 5, 9, 5, 7, 8, 9, 4, 8, 1, 5, 4, 4, 9, 6, 1, 8, 6, 0, 4, 5,  
2, 7, 1, 6, 4, 5, 6, 0, 3, 2, 3, 6, 7, 1, 9, 1, 4, 7, 6, 5, 8, 5,  
5, 1, 5, 2, 8, 8, 9, 9, 7, 6, 2, 2, 2, 3, 4, 8, 8, 3, 6, 0, 9, 7,  
7, 0, 1, 0, 4, 5, 1, 5, 3, 6, 0, 4, 1, 0, 0, 3, 6, 5, 9, 7, 3, 5,  
5, 9, 9, 8, 5, 3, 3, 2, 0, 5, 8, 3, 4, 0, 2, 4, 6, 4, 3, 4, 5, 0,  
5, 2, 1, 3, 1, 4, 1, 1, 7, 0, 1, 5, 2, 1, 2, 8, 7, 0, 6, 4, 8, 8,  
5, 1, 8, 4, 5, 8, 7, 9, 8, 6, 0, 6, 2, 0, 7, 9, 8, 9, 5, 2, 7, 7,  
1, 8, 7, 4, 3, 8, 3, 5])

[22]: from sklearn.metrics import accuracy\_score

[23]: accuracy\_score(y\_test,y\_pred)

[23]: 0.9694444444444444

[24]: from sklearn import metrics

[25]: confusion\_matrix=metrics.confusion\_matrix(y\_test,y\_pred)  
confusion\_matrix

Simple 0 7 Python 3 (ipykernel) Idle Mem: 1.31 GB

Mode: Command Ln 1, Col 11 L2.ipynb

← → ↺

Not secure 10.138.19/user/ra2311047010011/lab/workspaces/auto-b/tree/DEEP%20LEARNING/L2.ipynb

🔍 ☆ ⬇️ 🔒 Verify that it's you

File Edit View Run Kernel Tabs Settings Help

+

+

+

+

Filter files by name 🔍

/ DEEP LEARNING /

Name	Last Modified
Breast_canc...	next year
Exp2.ipynb	next year
L1.1.ipynb	next year
L1.ipynb	next year
L2.ipynb	next year
L3.ipynb	next year
Untitled.ip...	next year
Untitled1.ip...	6 days ago

L1.ipynb L3.ipynb L1.1.ipynb L2.ipynb

+

✂️

📄

📄

▶

⏏

Code

24

from sklearn import metrics

25

confusion\_matrix=metrics.confusion\_matrix(y\_test,y\_pred)

confusion\_matrix

25

array([[33, 0, 0, 0, 0, 0, 0, 0, 0, 0],

[ 0, 28, 0, 0, 0, 0, 0, 0, 0, 0],

[ 0, 0, 33, 0, 0, 0, 0, 0, 0, 0],

[ 0, 0, 0, 33, 0, 1, 0, 0, 0, 0],

[ 0, 1, 0, 0, 44, 0, 1, 0, 0, 0],

[ 0, 0, 1, 0, 0, 44, 1, 0, 0, 1],

[ 0, 0, 0, 0, 0, 1, 34, 0, 0, 0],

[ 0, 0, 0, 0, 0, 1, 0, 33, 0, 0],

[ 0, 0, 0, 0, 0, 1, 0, 0, 29, 0],

[ 0, 0, 0, 1, 0, 0, 0, 0, 1, 38]])

26

cm\_display=metrics.ConfusionMatrixDisplay(confusion\_matrix=confusion\_matrix,display\_labels=[0,1])

cm\_display

26

<sklearn.metrics.\_plot.confusion\_matrix.ConfusionMatrixDisplay at 0x720a307ade40>

27

pip install matplotlib

Defaulting to user installation because normal site-packages is not writeable

Collecting matplotlib

Downloading matplotlib-3.10.5-cp310-cp310-manylinux2014\_x86\_64.manylinux\_2\_17\_x86\_64.whl.metadata (11 kB)

Collecting contourpy>=1.0.1 (from matplotlib)

Downloading contourpy-1.3.2-cp310-cp310-manylinux\_2\_17\_x86\_64.manylinux2014\_x86\_64.whl.metadata (5.5 kB)

Collecting cycler>=0.10 (from matplotlib)

Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)

Collecting fonttools>=4.22.0 (from matplotlib)

Downloading fonttools-4.59.0-cp310-cp310-manylinux2014\_x86\_64.manylinux\_2\_17\_x86\_64.whl.metadata (107 kB)

Collecting kiwisolver>=1.3.1 (from matplotlib)

Downloading kiwisolver-1.4.8-cp310-cp310-manylinux\_2\_12\_x86\_64.manylinux2010\_x86\_64.whl.metadata (6.2 kB)

Requirement already satisfied: numpy>=1.23 in /home/jupyter-ra2311047010011/.local/lib/python3.10/site-packages (from matplotlib) (2.2.6)

Requirement already satisfied: packaging>=20.0 in /opt/tljh/user/lib/python3.10/site-packages (from matplotlib) (24.0)

Collecting pillow>=8 (from matplotlib)

Downloading pillow-11.3.0-cp310-cp310-manylinux\_2\_27\_x86\_64.manylinux\_2\_28\_x86\_64.whl.metadata (9.0 kB)

Collecting pyparsing>=2.3.1 (from matplotlib)

Downloading pyparsing-3.2.3-py3-none-any.whl.metadata (5.0 kB)

Requirement already satisfied: python-dateutil>=2.7 in /opt/tljh/user/lib/python3.10/site-packages (from matplotlib) (2.9.0.post0)

Requirement already satisfied: six>=1.5 in /opt/tljh/user/lib/python3.10/site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)

Simple 0 7 Python 3 (ipykernel) idle Mem: 1.31 GB

Mode: Command Ln 1, Col 11 L2.ipynb