Task-12

GO_STP_5247

Train SVM classifier using sklearn digits dataset

- 1. Measure accuracy of your model using different kernels such as rbf and linear.
- 2. Tune your model further using regularization and gamma parameters and try to come up with highest accurancy score 3.Use 80% of samples as training data size
- import pandas as pd import sklearn from sklearn.datasets import load digits
 - from sklearn.model selection import train test split
 - from sklearn.metrics import confusion matrix, accuracy score

[] import numpy as np

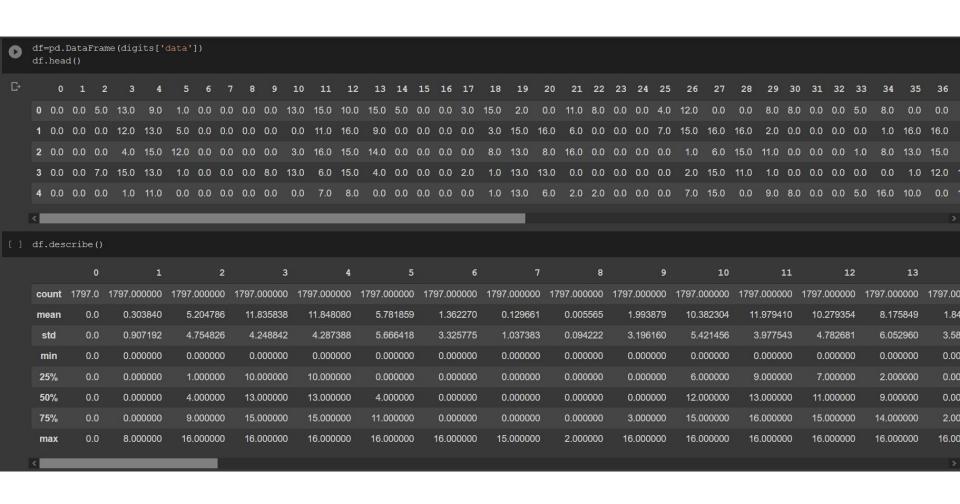
- [] digits = load digits()
- [] digits.keys()
- print(digits['DESCR'])
- [] digits.target names Os completed at 1:26 PM











```
df['target']=digits.target
    df.head()
       digits target
[ ] X=df.drop(['target'],axis='columns')
```

[] X_train, X_test, Y_train, Y_test=train_test_split(X, Y, test_size=0.2, random_state=5)

```
[ ] mymodel = SVC()
    SVC(C=1.0, break ties=False, cache size=200, class weight=None, coef0=0.0,
        decision function shape='ovr', degree=3, gamma='scale', kernel='rbf',
        max iter=-1, probability=False, random state=None, shrinking=True,
[ ] mymodel.score(X test,Y test)
[ ] pred=mymodel.predict(X test)
[ ] cm=np.array(confusion matrix(Y test,pred))
    array([[35, 0, 0, 0, 0, 0, 0, 0, 0, 0],
```

Tuning

model_C.fit(X_train,Y_train)

```
model C.score(X test, Y test)
model C.fit(X train, Y train)
model C.score(X test, Y test)
```

model_C.score(X_test,Y_test)

```
[ ] model kernel=SVC(C=3, kernel='linear')
    model kernel.fit(X train, Y train)
    model kernel.score(X test, Y test)
```

```
[ ] model kernel=SVC(C=3, kernel='rbf')
```

```
model kernel.fit(X train, Y train)
```

```
model kernel.score(X test, Y test)
```

```
[ ] model g=SVC(C=3,kernel='poly',degree=5, gamma='auto')
    model g.score(X test, Y test)
[ ] finalModel=SVC(C=3)
    finalModel.fit(X train, Y train)
    finalModel.score(X test, Y test)
[ ] pred=finalModel.predict(X test)
    cm=np.array(confusion matrix(Y test,pred))
    array([[35, 0, 0, 0, 0, 0, 0, 0, 0],
```