MM20B007 Tutorial 9

```
import pandas as pd
import numpy as np
from sklearn.svm import SVC
from sklearn.metrics import classification_report
from sklearn.model_selection import GridSearchCV
```

```
datafile_path = "/content/drive/MyDrive/sem 7/ID5055/Tutorial 9/data.npy"
data = np . load ( datafile_path , allow_pickle = True ). item ()

Xtrain , Ytrain = data ["train"]["X"] , data ["train"]["Y"]
Xtest , Ytest = data ["test"]["X"] , data ["test"]["Y"]
```

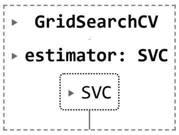
```
svc = SVC()
svc.fit(Xtrain, Ytrain)
y_pred = svc.predict(Xtest)
```

```
print(f'The classification report for the following prediction is\n\n\n {classification_report(Ytest, y_pred)}')
```

```
The classification report for the following prediction is


               precision    recall  f1-score   support

           0       0.98      0.99      0.98        98
           1       0.99      0.98      0.99       102

    accuracy                           0.98       200
   macro avg       0.98      0.99      0.98       200
weighted avg       0.99      0.98      0.99       200
```

Optimizaing the model with Grid Search

```
parameters = [
    {'C': [1, 10, 100, 1000], 'kernel': ['linear']},
    {'C': [1, 10, 100, 1000], 'kernel': ['rbf'], 'gamma': [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]},
    {'C': [1, 10, 100, 1000], 'kernel': ['sigmoid'], 'gamma': [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]},
    {'C': [1, 10, 100, 1000], 'kernel': ['poly'], 'degree': [2, 3, 4, 5], 'gamma': [0.01, 0.02, 0.03, 0.04, 0.05]}
]


grid_search = GridSearchCV(estimator = svc, param_grid = parameters, scoring = 'accuracy')
grid_search.fit(Xtrain, Ytrain)
```



```
# best score achieved during the GridSearchCV
print('GridSearch CV best score : {:.4f}\n\n'.format(grid_search.best_score_))


# print parameters that give the best results
print('Parameters that give the best results :','\n\n', (grid_search.best_params_))


# print estimator that was chosen by the GridSearch
print('\n\nEstimator that was chosen by the search :','\n\n', (grid_search.best_estimator_))
```

```
GridSearch CV best score : 0.9900


Parameters that give the best results :

 {'C': 1, 'gamma': 0.4, 'kernel': 'rbf'}


Estimator that was chosen by the search :

 SVC(C=1, gamma=0.4)
```

```
svc_new = SVC(C=1, gamma=0.4)
svc_new.fit(Xtrain, Ytrain)
y_pred_new = svc_new.predict(Xtest)
```

```
print(f'The classification report for the following prediction is\n\n\n {classification_report(Ytest, y_pred_new)}')
```

```
The classification report for the following prediction is


               precision    recall  f1-score   support

           0       0.98      0.99      0.98        98
           1       0.99      0.98      0.99       102

    accuracy                           0.98       200
   macro avg       0.98      0.99      0.98       200
weighted avg       0.99      0.98      0.99       200
```