Harsh Seksaria

2048011

2 - MDS

Machine Learning Lab - 7 and 8

    Support Vector Machine Regression & Classification

    25 March 2021

CHRIST (Deemed to be University)

In [1]:

```
#Import statements
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_error
```

Reading dataset

In [2]:

```
# Read dataset in dataframe
data = pd.read_csv("../diabetes.csv")

#Dividing dataset into x and y
x = data.drop('Outcome', axis=1)
y = data['Outcome']

#Dividing dataset into training and testing set
X_train, X_test, Y_train, Y_test = train_test_split(x, y, test_size=0.25, random_state=11)
```

SVM Regression

In [9]:

```
#Gaussian Kernel

from sklearn.svm import SVR
svrclassifier = SVR(kernel="rbf")
svrclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svr = svrclassifier.predict(X_test)

from sklearn.metrics import r2_score
print("R2 score: ", r2_score(Y_test, Y_pred_svr))

from sklearn.metrics import mean_squared_error
print("Mean Squared Error: ", mean_squared_error(Y_test, Y_pred_svr))
```

```
R2 score:  0.21718928679514748
Mean Squared Error:  0.1765910886233603
```

In [8]:

```
#Linear Kernel

from sklearn.svm import SVR
svrclassifier = SVR(kernel="linear")
svrclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svr = svrclassifier.predict(X_test)

from sklearn.metrics import r2_score
print("R2 score: ", r2_score(Y_test, Y_pred_svr))

from sklearn.metrics import mean_squared_error
print("Mean Squared Error: ", mean_squared_error(Y_test, Y_pred_svr))
```

```
R2 score:  0.24060830364949182
R2 score:  0.17130808775094472
```

In [10]:

```
#Polynomial Kernel

from sklearn.svm import SVR
svrclassifier = SVR(kernel="poly", degree=8)
svrclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svr = svrclassifier.predict(X_test)

from sklearn.metrics import r2_score
print("R2 score: ", r2_score(Y_test, Y_pred_svr))

from sklearn.metrics import mean_squared_error
print("Mean Squared Error: ", mean_squared_error(Y_test, Y_pred_svr))
```

```
R2 score:  -6214.2287841745765
Mean Squared Error:  1402.068212055007
```

In [11]:

```
#Sigmoid Kernel

from sklearn.svm import SVR
svrclassifier = SVR(kernel="sigmoid", degree=8)
svrclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svr = svrclassifier.predict(X_test)

from sklearn.metrics import r2_score
print("R2 score: ", r2_score(Y_test, Y_pred_svr))

from sklearn.metrics import mean_squared_error
print("Mean Squared Error: ", mean_squared_error(Y_test, Y_pred_svr))
```

```
R2 score:  -153.5318515223614
Mean Squared Error:  34.8602125992827
```

SVM Classification

In [13]:

```
#Gaussian Kernel

from sklearn.svm import SVC
svcclassifier = SVC(kernel="rbf")
svcclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svm = svcclassifier.predict(X_test)

#Printing the predictions
Y_pred_svm

#Evaluation
from sklearn.metrics import classification_report, confusion_matrix
print("CLASSIFICATION REPORT:\n\n", classification_report(Y_test, Y_pred_svm))
print("\nCONFUSION MATRIX:\n\n", confusion_matrix(Y_test, Y_pred_svm))
```

```
CLASSIFICATION REPORT:

               precision    recall  f1-score   support

           0       0.74      0.92      0.82       126
           1       0.71      0.38      0.50        66

    accuracy                           0.73       192
   macro avg       0.73      0.65      0.66       192
weighted avg       0.73      0.73      0.71       192


CONFUSION MATRIX:

 [[116  10]
 [ 41  25]]
```

In [14]:

```
#Polynomial Kernel

from sklearn.svm import SVC
svcclassifier = SVC(kernel="poly", degree=8)
svcclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svm = svcclassifier.predict(X_test)

#Printing the predictions
Y_pred_svm

#Evaluation
from sklearn.metrics import classification_report, confusion_matrix
print("CLASSIFICATION REPORT:\n\n", classification_report(Y_test, Y_pred_svm))
print("\nCONFUSION MATRIX:\n\n", confusion_matrix(Y_test, Y_pred_svm))
```

```
CLASSIFICATION REPORT:

               precision    recall  f1-score   support

           0       0.72      0.97      0.83       126
           1       0.83      0.29      0.43        66

    accuracy                           0.73       192
   macro avg       0.77      0.63      0.63       192
weighted avg       0.76      0.73      0.69       192


CONFUSION MATRIX:

 [[122   4]
 [ 47  19]]
```

In [15]:

```
#Sigmoid Kernel

from sklearn.svm import SVC
svcclassifier = SVC(kernel="sigmoid")
svcclassifier.fit(X_train, Y_train)

#Prediction
Y_pred_svm = svcclassifier.predict(X_test)

#Printing the predictions
Y_pred_svm

#Evaluation
from sklearn.metrics import classification_report, confusion_matrix
print("CLASSIFICATION REPORT:\n\n", classification_report(Y_test, Y_pred_svm))
print("\nCONFUSION MATRIX:\n\n", confusion_matrix(Y_test, Y_pred_svm))
```

```
CLASSIFICATION REPORT:

               precision    recall  f1-score   support

           0       0.60      0.72      0.65       126
           1       0.12      0.08      0.09        66

    accuracy                           0.50       192
   macro avg       0.36      0.40      0.37       192
weighted avg       0.44      0.50      0.46       192


CONFUSION MATRIX:

 [[91 35]
 [61  5]]
```