CCPS 844 Data Mining (Lab 7) Submit your solution as a pdf file

- Q-1 Select a dataset/datasets of your choice. Apply SVM Classification Evaluate the results
- Q-2 Select a dataset/datasets of your choice. Apply SVM Regression Evaluate the results

In [34]:

```
% matplotlib inline

from IPython.core.display import display, HTML
display(HTML("<style>.container { width:90% !important; }</style>"))
```

In [35]:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
pd.set_option('display.max_columns', 100)
```

In [36]:

```
from sklearn import svm

df = pd.read_csv('winequality-red.csv')
df['overfive'] = (df.quality > 5).astype(int)
```

In [37]:

```
df.sample(5)
```

Out[37]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
191	6.4	0.370	0.25	1.9	0.074	21.0	49.0	0.99740	3.57	0.62	9.800000	6
859	6.8	0.560	0.22	1.8	0.074	15.0	24.0	0.99438	3.40	0.82	11.200000	6
1359	11.6	0.475	0.40	1.4	0.091	6.0	28.0	0.99704	3.07	0.65	10.033333	6
423	10.5	0.240	0.47	2.1	0.066	6.0	24.0	0.99780	3.15	0.90	11.000000	7
523	9.3	0.400	0.49	2.5	0.085	38.0	142.0	0.99780	3.22	0.55	9.400000	5

In [38]:

```
X=df.iloc[:,0:3].values
y=df.quality.values
```

In [39]:

```
clf = svm.SVR()
clf.fit(X, y)
```

Out[39]:

```
SVR(C=1.0, cache_size=200, coef0=0.0, degree=3, epsilon=0.1, gamma='auto',
  kernel='rbf', max_iter=-1, shrinking=True, tol=0.001, verbose=False)
```

In [40]:

```
clf.predict([[7,0.65,0.02]])
```

Out[40]:

```
array([ 5.30354839])
```

In [41]:

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
```

In [45]:

```
y=np.reshape(y,(-1,1))
```

In [49]:

```
from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
sc_y = StandardScaler()
X1 = sc_X.fit_transform(X)
y1 = sc_y.fit_transform(y)
```

```
/Users/dee/anaconda3/lib/python3.6/site-packages/sklearn/utils/validation.py:475: DataCo
nversionWarning: Data with input dtype int64 was converted to float64 by StandardScaler.
  warnings.warn(msg, DataConversionWarning)
```

In [56]:

```
X1
```

Out[56]:

```
array([[-0.52835961,  0.96187667, -1.39147228],
       [-0.29854743,  1.96744245, -1.39147228],
       [-0.29854743,  1.29706527, -1.18607043],
       ...,
       [-1.1603431 , -0.09955388, -0.72391627],
       [-1.39015528,  0.65462046, -0.77526673],
       [-1.33270223, -1.21684919,  1.02199944]])
```

```
In [48]: regressor = svm.SVR(kernel='rbf')
         regressor.fit(X1, y1)
```

```
/Users/dee/anaconda3/lib/python3.6/site-packages/sklearn/utils/validation.py:578: DataCo
nversionWarning: A column-vector y was passed when a 1d array was expected. Please chang
e the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Out[48]: SVR(C=1.0, cache_size=200, coef0=0.0, degree=3, epsilon=0.1, gamma='auto',
           kernel='rbf', max_iter=-1, shrinking=True, tol=0.001, verbose=False)
```

```
In [58]: y_pred = regressor.predict(sc_X.transform(np.array([[0,0,0]])))
         y_pred = sc_y.inverse_transform(y_pred)
         y_pred
```

```
/Users/dee/anaconda3/lib/python3.6/site-packages/sklearn/utils/validation.py:475: DataCo
nversionWarning: Data with input dtype int64 was converted to float64 by StandardScaler.
  warnings.warn(msg, DataConversionWarning)
```

```
Out[58]: array([ 5.5876168])
```