Support Vector Machine

Prajjwal Vijaywargiya - 2017B3A70954H
Siddhi Mahesh Burse - 2017B3A70972H
Parth Krishna Sharma - 2017B3A70907H

In [1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn import svm
from sklearn.metrics import confusion_matrix,classification_report,accuracy_score
```

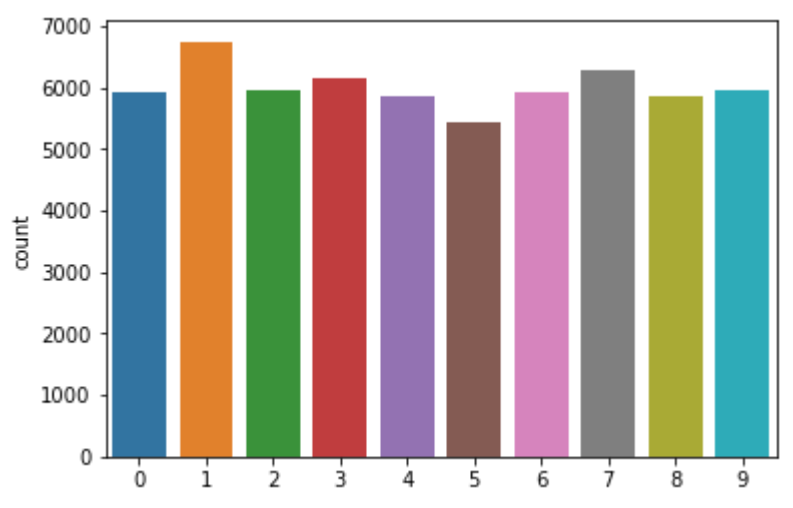
In [2]:
```
train = pd.read_csv('D:/BITS/DL/Assignments/Assignment1/mnist_train.csv')
test = pd.read_csv('D:/BITS/DL/Assignments/Assignment1/mnist_test.csv')
```

In [3]:
```
data_train = train.iloc[:,1:].values
label_train = train.iloc[:,0].values
```

In [4]:
```
data_test = test.iloc[:,1:].values
label_test = test.iloc[:,0].values
```

In [5]:
```
sns.countplot(label_train)
```

Out[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x17fa626cd88>
```



In [6]:
```
data_train[data_train>0]=1
data_test[data_test>0]=1
```

In [7]:
```
print(data_train)
print(label_train)
```

```
[[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
[5 0 4 ... 5 6 8]
```

In [8]:
```
classifier = svm.SVC(verbose=True)
```

In [9]:
```
classifier.fit(data_train,label_train)
```

```
[LibSVM]
```

Out[9]:
```
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,
    tol=0.001, verbose=True)
```

In [10]:
```
predicted = classifier.predict(data_test)
```

In [12]:
```
confusion_matrix(label_test,predicted)
```

Out[12]:
```
array([[ 972,    1,    0,    0,    0,    0,    2,    1,    3,    1],
       [   0, 1122,    2,    1,    1,    1,    2,    0,    6,    0],
       [   4,    0, 1010,    3,    1,    0,    0,    7,    7,    0],
       [   0,    0,    3,  986,    0,    2,    0,    5,   11,    3],
       [   2,    0,    2,    0,  956,    0,    3,    1,    2,   16],
       [   2,    0,    0,   11,    0,  863,    4,    1,    9,    2],
       [   7,    2,    0,    0,    2,    4,  940,    0,    3,    0],
       [   2,    5,    9,    2,    2,    0,    0,  998,    1,    9],
       [   5,    1,    4,    8,    5,    3,    1,    4,  941,    2],
       [   1,    6,    1,   10,    8,    3,    1,    5,    6,  968]],
      dtype=int64)
```

In [17]:
```
acc = accuracy_score(label_test,predicted)
acc
```

Out[17]:
```
0.9756
```

In [18]:
```
sns.heatmap(confusion_matrix(label_test,predicted),cmap='Blues_r',fmt='',annot=True,cbar=False)
plt.ylabel("Correct Label")
plt.xlabel("Predicted Label")
plt.title("Confusion Matrix for SVM\nModel Accuracy: {}%".format(acc*100))
```

Out[18]:
```
Text(0.5, 1.0, 'Confusion Matrix for SVM\nModel Accuracy: 97.56%')
```