Report for Project1

MENG Qian 20466688

It is a classification problem in this project. The data set is given, which is consist of 3 csv files. We need to train a model on the training set and use the test data to predict the label of them, and then through the accuracy of the result to measure the model's quality.

Obviously, we do not know which classifier is the best one which has the highest accuracy. To carry out this project, we need to choose some of classifier at first that suitable for solving such problems. After experiments, we choose the model which the accuracy is highest as our model.

Data description

The data set consists of the training data, the labeled training data and the test data. Through command lines such as 'head', 'wc', we know that train data is 3220*57; labeled train data is 3220*57; the test data is 1380*57.

The first 10 data points of training data (incomplete),

```
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.86,0.0,0.0,0.0,0.0,0.0,0.0,1.73,0.0,0.86,0.0,0.0,0.0,6.08,3.47,0.86,0.86,0.86,0.86,0.86
0.09,0.0,0.27,0.0,0.36,0.09,0.0,0.18,0.09,0.0,0.0,0.72,0.0,0.36,0.0,0.0,0.0,0.0,2.0,0.0,3.27,0.0,0.36,0.09,0.0,0.0,0.0,0.0,0.0,0.0,0.
0.0,0.0,0.13,0.0,0.26,0.0,0.0,0.65,0.13,0.0,0.0,0.78,0.26,0.0,0.0,0.0,0.13,0.0,0.0,0.0,0.0,0.0,0.13,0.0,1.69,0.0,0.0,0.0,0.0,0.0,0.0,
0.0,1.28,0.0,0.0,0.64,0.0,0.0,0.0,0.0,1.28,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.28,0.0,0.0,0.0,0.64,0.64,1.92,0.0,0.0,0.0,0.0,0.
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,2.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,4.0,2.0,2.0,2.0,2.0,4.0,2.0,2.0,0.0,2
0.0,1.42,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.42,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,4.28,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.54,0.0,0.54,0.0,0.0,0.0,0.0
0.0,0.0,1.11,0.0,1.11,0.0,0.74,0.0,0.0,0.0,0.74,0.37,0.0,0.0,0.0,0.0,0.37,0.0,3.35,2.98,2.61,0.0,0.0,0.37,0.0,0.0,0.0,0.0,0.0,0.0,0.0
0.0,0.0,0.54,0.0,1.08,0.0,0.0,0.0,0.0,0.0,0.0,0.18,0.9,0.0,0.18,0.0,0.36,0.36,3.06,0.0,0.9,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.88,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
```

The first 10 data points of labeled training data,

```
0.0
1.0
0.0
0.0
0.0
0.0
0.0
1.0
1.0
0.0
```

The first 10 data points of test data (incomplete),

```
0.0,0.47,0.47,0.0,1.89,0.0,1.18,0.23,0.0,0.47,0.23,0.7,0.23,0.0,0.47,0.23,1.41,0.0,2.83,0.0,1.65,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
1.16,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.16,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.16,1.16,0.0,0.0,0.0,0.0,1.16,0.0,
0.0,0.0,0.32,0.0,0.65,0.32,0.32,0.32,0.0,0.0,0.65,1.3,0.0,0.0,0.0,0.98,0.0,0.65,2.61,2.61,3.26,0.0,0.0,0.65,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
0.0,0.0,1.31,0.0,0.65,0.0,0.0,0.0,0.0,0.0,0.0,0.65,0.0,0.0,0.0,0.0,0.0,0.0,5.26,1.97,4.6,0.0,0.0,1.31,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
0.0,0.26,0.0,0.0,0.26,0.0,0.0,0.0,0.0,1.88,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.26,0.0,0.26,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,16.66,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
0.0,0.0,0.81,0.0,0.27,0.0,0.0,0.0,0.0,0.27,0.0,0.27,0.27,0.0,0.0,0.0,0.0,0.0,2.16,0.0,0.54,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,20.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
0.0,0.0,0.0,0.0,0.0,0.64,0.0,0.0,0.0,0.0,0.0,0.0,0.64,0.64,0.0,0.0,0.0,0.0,0.0,1.29,0.0,1.29,5.19,1.29,0.0,0.0,0.0,0.64,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
0.15,0.0,0.3,0.0,1.23,0.61,0.0,0.61,0.0,0.15,0.0,0.3,0.0,0.0,0.0,0.15,0.3,1.54,2.32,1.85,0.0,1.08,0.0,0.0,0.3,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
```

Data import and processing

```
# import data
data_train = pd.read_csv(r'/Users/mengqian/Desktop/DLproject1/traindata.csv')
label_train = pd.read_csv(r'/Users/mengqian/Desktop/DLproject1/trainlabel.csv')
label_test = pd.read_csv(r'/Users/mengqian/Desktop/DLproject1/testdata.csv')

# data processing
train_scaled = preprocessing.scale(data_train)
test_scaled = preprocessing.scale(label_test)
train_scaled = pd.DataFrame(data=train_scaled)
label_train = np.ravel(label_train)
clf.fit(train_scaled, label_train)
```

Select models

For solving this problem, I chose three classifiers to label the test data.

```
models = [
    # Naive Bayes classifier for multivariate Bernoulli models
    (BernoulliNB(), "Naive_Bayes"),

    # Classifier implementing the k-nearest neighbors vote
    (KNeighborsClassifier(algorithm='kd_tree'), "k_nearest"),

    # C-Support Vector Classification
    (SVC(probability=True, kernel='rbf'), "svc")
]
```

Evaluate models

To avoid over-fitting, using k-fold cross-validation method to evaluate the accuracy of models.

```
# use Cross Validation to estimate the accuracy of the model
scores = cross_val_score(clf, train_scaled, label_train, cv=5)
print(scores)
```

```
# print the result and save as a csv file
result = clf.predict(test_scaled)
res = map(lambda x: int(x), result)
file_name = "/Users/mengqian/Desktop/DLproject1/test_label_%s.csv" % model_name
np.savetxt(file_name , res, delimiter=",")
print(res)
```

```
[ 0.90232558  0.90993789  0.89440994  0.89269051  0.90357698]
[0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1,
[ 0.88062016  0.90062112  0.90372671  0.9066874   0.88491446]
[0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1,
[ 0.9255814   0.93322981  0.93012422  0.9377916   0.91912908]
[0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1,
```

Conclusion

From the output, it can be included that the svm of C-Support Vector Classification method has the highest accuracy, so I choose it as my model in this project.