Jonggoo Kang

CSC529 Assignment #2

Problem 1: (15 points) One of the shortcomings of the results in Problem 2 from Assignment#1 is due to the fact of considering a single sample for training set as well as for test set (a single trial of 66% for training and 34% for testing was used to build the model).

a. (10 points) Repeat Problem 2.a&b from Assignment#1 on the Wine Recognition Dataset at least 30 times and report the means, variances, and Confidence Intervals (CI) for the accuracy results on the training and testing sets.

```
================================================================
* Mean for the acuuracy on training sets
         0.917094017094
* Mean for the acuuracy on testing sets
         0.846994535519
================================================================
* Variance for the acuuracy on training sets
         0.000323189271023
* Variance for the acuuracy on training sets
         0.00191519397517
================================================================
* The 95% CI for DT training sets
        Low                   High
(0.91038112109875502, 0.92380691308927931)
* The 95% CI for DT Testing sets
        Low                   High
(0.83065318918051023, 0.86333588185774124)
================================================================
```

b. (5 points) Using a pair t-test, compare the mean accuracy of the Naïve Bayes and the mean accuracy of the Decision tree and discuss the results.

```
================================================================
* Mean for the acuuracy on training sets
         0.982051282051
* Mean for the acuuracy on testing sets
         0.975956284153
================================================================
* Variance for the acuuracy on training sets
         8.23717004088e-05
* Variance for the acuuracy on training sets
         0.000254535457345
================================================================
* The 95% CI for DT training sets
        Low                   High
(0.97866229065147414, 0.98544027345109009)
* The 95% CI for DT training sets
        Low                   High
(0.96999890013993895, 0.98191366816607206)
================================================================
```

Problem 2: (15 points) The objective of this exercise is to analyze the performance of the previously trained Naïve Bayes and decision trees classifiers (using the approach in Problem 1) as a function of the sample size.

Let us vary the training data (and the corresponding values of the class attribute) such that the size of the training data is 25%, 35%, 45%, 55%, 65%, 75%, and 85% of the original data. Create a decision tree and a Naive Bayes classifier for each one of the seven different data splits, record the resubstitution error and the generalization error, and present the results in a plot with 4 curves (each corresponding to one type of error and one type of classifier) as a function of the size of the data. Perform an analysis of the results obtained.

	DT Generalization Error	DT Resubstituition Error	NB Generalization Error	NB Resubstituition Error	Training Size
0	0.222222	0.082707	0.022222	0.007519	133
1	0.206349	0.060870	0.047619	0.008696	115
2	0.148148	0.041237	0.061728	0.010309	97
3	0.193878	0.050000	0.040816	0.025000	80
4	0.431034	0.290323	0.043103	0.000000	62
5	0.597015	0.613636	0.097015	0.000000	44
6	0.625000	0.461538	0.276316	0.000000	26

Extra credit (3 points): Repeat Problem 2 on the following additional datasets and interpret the results:

winered.data: https://archive.ics.uci.edu/ml/datasets/Wine+Quality

<EXTRA with WineQuality>

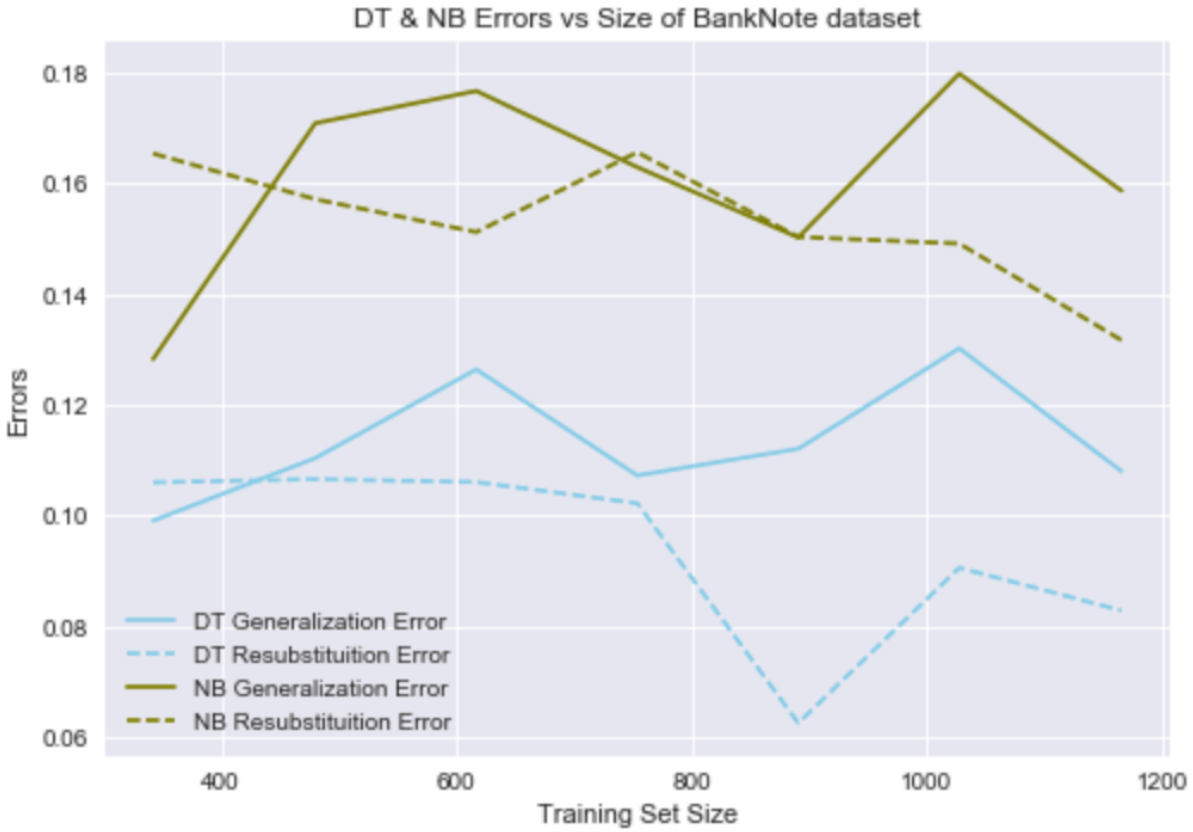
	DT Generalization Error	DT Resubstituition Error	NB Generalization Error	NB Resubstituition Error	Training Size
0	0.440000	0.448707	0.455000	0.429525	400
1	0.482143	0.433109	0.419643	0.430221	560
2	0.487500	0.425484	0.484722	0.497156	719
3	0.444318	0.439499	0.475000	0.464534	879
4	0.457692	0.402504	0.450962	0.415027	1040
5	0.468333	0.423559	0.444167	0.428571	1199
6	0.459559	0.422594	0.511765	0.447699	1360

The result shows that the graphs for the errors are fluctuating. This is because there are so many classes in 'quality' variable. I learned how important having a binary classification is.

banknote.data: https://archive.ics.uci.edu/ml/datasets/banknote+authentication

<EXTRA with BankNote>

	DT Generalization Error	DT Resubstituition Error	NB Generalization Error	NB Resubstituition Error	Training Size
0	0.099125	0.106031	0.128280	0.165370	342
1	0.110417	0.106622	0.170833	0.157127	480
2	0.126418	0.106101	0.176661	0.151194	617
3	0.107285	0.102273	0.162914	0.165584	754
4	0.112108	0.062630	0.150224	0.150313	891
5	0.130224	0.090643	0.179786	0.149123	1028
6	0.108062	0.082927	0.158662	0.131707	1166



The errors are pretty low, but there is no correlation between number of training set and the errors here too.

Problem 3: (15 points) The objective of this exercise is to get familiar with the evaluation of probabilistic classifiers using ROC and Lift curves.

a. Repeat Problem 2.b from Assignment#1 on the Wine Recognition Dataset but this time considering only two classes (let us say, class 1 (positive class) versus class 2 and class 3 (negative class) since the ROC and lift curves can only be drawn for binary classification problems).

< class1 = p, class 2&3 = n >

```
DecisionTreeClassifier(class_weight=None, criterion='entropy', max_depth=2,
            max_features=None, max_leaf_nodes=None,
            min_impurity_split=1e-07, min_samples_leaf=1,
            min_samples_split=50, min_weight_fraction_leaf=0.0,
            presort=False, random_state=None, splitter='best')
```

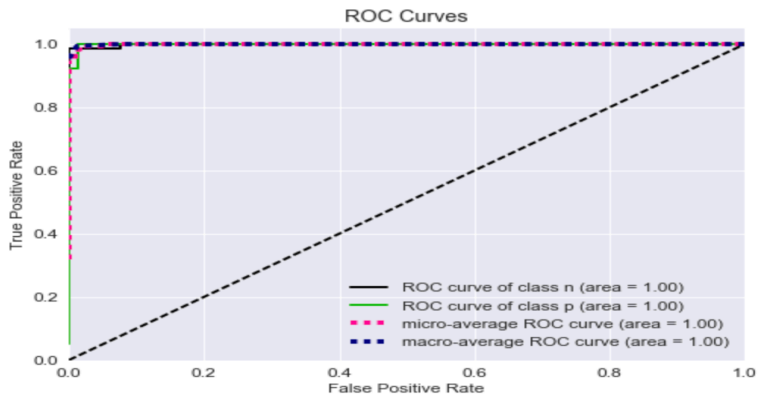
```
array(['n', 'n', 'p', 'p', 'n', 'n', 'n', 'p', 'n', 'p', 'n', 'n', 'n',
       'n', 'p', 'n', 'p', 'n', 'p', 'n', 'n', 'p', 'p', 'n', 'n', 'n',
       'p', 'n', 'n', 'p', 'n', 'n', 'p', 'p', 'n', 'n', 'n', 'p', 'n',
       'n', 'p', 'n', 'n', 'n', 'n', 'n', 'p', 'p', 'p', 'n', 'p', 'n',
       'p', 'p', 'n', 'n', 'n', 'p', 'p'],
      dtype='<U1')
```

< classification report, accuracies on training and testing, confusion matrix>

```
                precision    recall  f1-score   support

             n       1.00      0.92      0.96        39
             p       0.87      1.00      0.93        20

avg / total          0.96      0.95      0.95        59

Accuracy on training 0.983193277311
Accuracy on testing 0.949152542373
     0    1
0   36    3
1    0   20
```

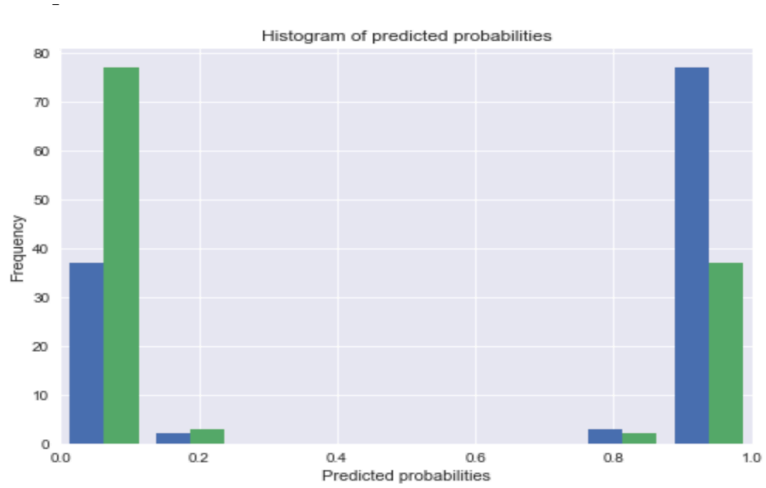
b. Draw the ROC curves for the Naïve Bayes performance on both the training and testing data. Interpret the graphs. If you would have to choose a certain probability threshold to maximize both sensitivity and specificity on the testing data, which threshold value would you select?

<ROC curve for Naïve Bayes performance on Training data>



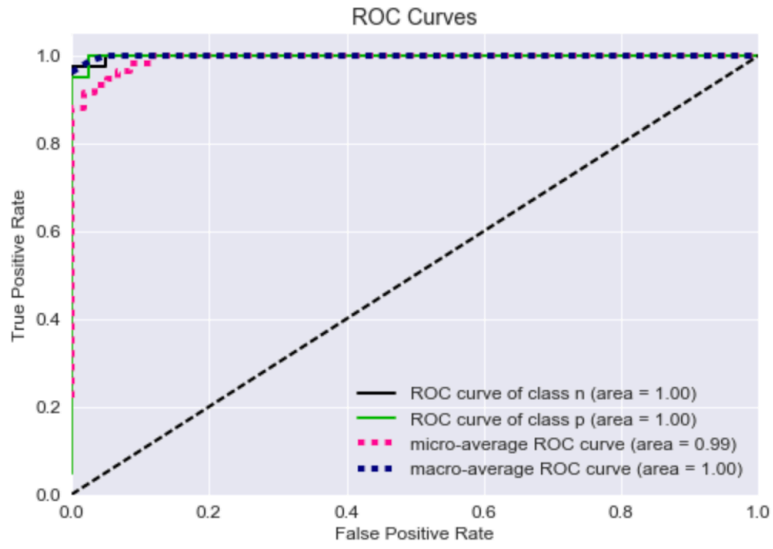
Based on the result, the separation between class1(p) and classes2&3(n) are very significant. Futhermore, the graph shows that it has a perfect convex curve.

The graph below shows the binary values for the target variables in training set.



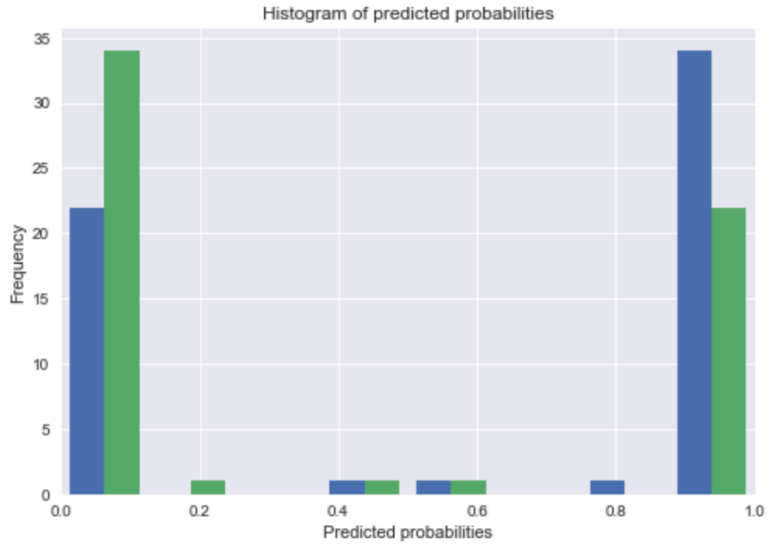
This graph shows that the threshold of 0.5. In the python function, a default threshold is 0.5. Thus, there will not be any change of threshold.

<ROC curve for Naïve Bayes performance on Testing data>



Based on the result, the separation between class1(p) and classes2&3(n) are very significant. Futhermore, the graph shows that it has a perfect convex curve.

The graph below shows the binary values for the target variables in testing set.

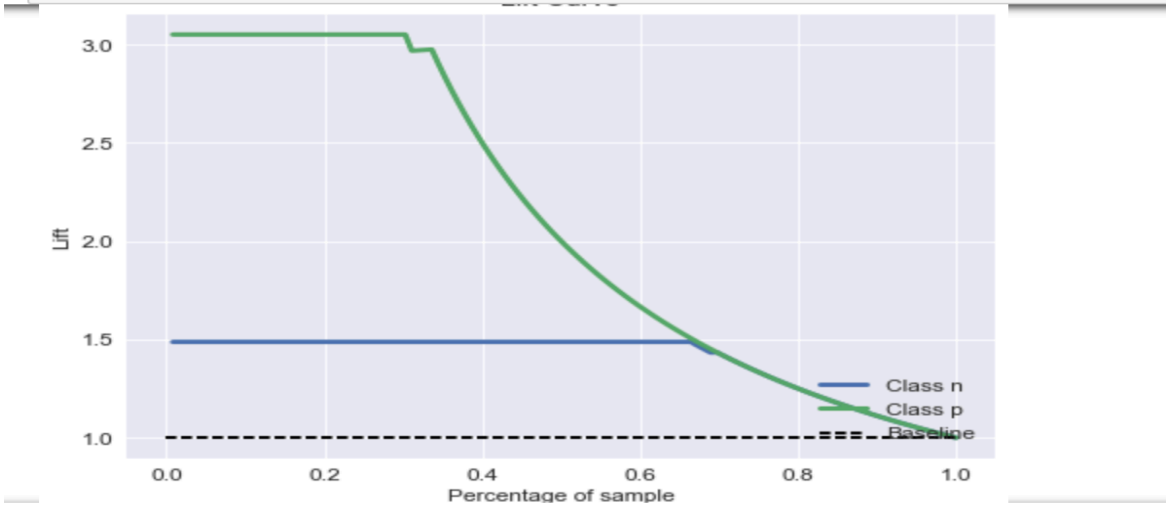


This graph shows that the threshold of 0.5. In the python function, a default threshold is 0.5. Thus, there will not be any change of threshold.

c. Draw the lift curves for the Naïve Bayes performance on both the training and testing data. Interpret the results. If the requirement is to get at least 80% accuracy on the data with a minimum cost of data acquisition, what size for the data would you recommend to reach that accuracy performance?

Lift curve for Naïve Bayes performance on Training data

```
skplt.metrics.plot_lift_curve(y_train_pb3, pred_prob_pb3)
```

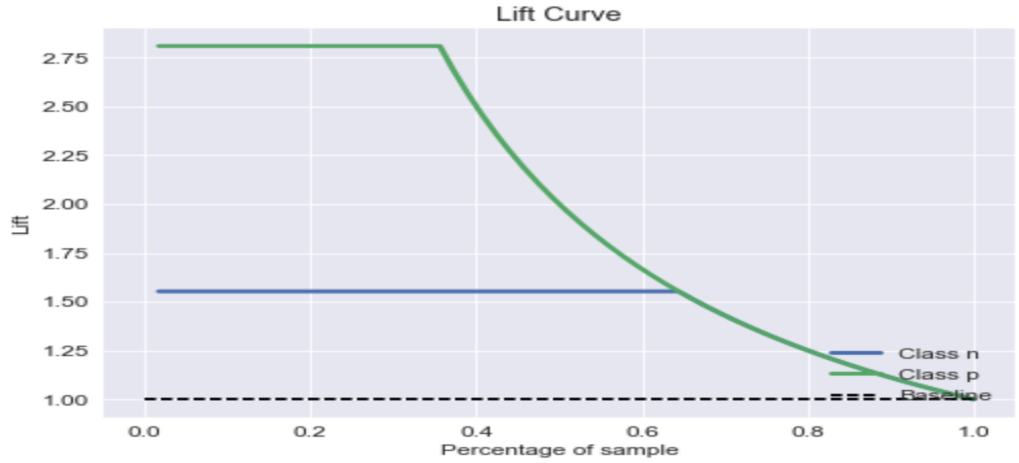


Lift curve shows that the effectiveness of a binary classifier. Here, the predictive model will choose three times more of class1 at the percentage of sample of 0% for training set.. Also, two times more of class at the percentage of sample of 0% for testing set. However, it would be very precise for 70%

in training, 60% in testing. If the requirement is to get at least 80%, we need to collect more data to be enough to holdout partition.

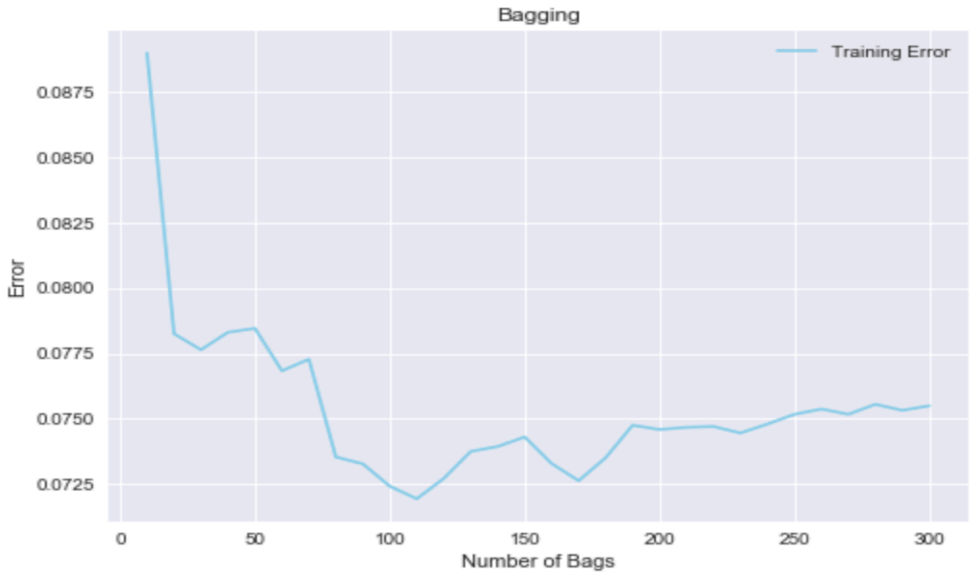
Lift curve for Naïve Bayes performance on Testing data

```
skplt.metrics.plot_lift_curve(y_test_pb3, pred_probas_pb3)
<matplotlib.axes._subplots.AxesSubplot at 0x113f7c588>
```



Problem 4: (15 points) The objective of this exercise is to get familiar with ensemble of classifiers and understand how the number of classifiers/learners affect the accuracy of the classifier.

a. (10 points) Model the diagnosis using ensemble learning based on bagging (with decision trees as learners) and plot the error rate as a function of the number of trees in the ensemble. Perform an analysis of your results.



The graph shows that with more number of bags provide low error.

b. (2.5 points) Explain if bagging is an appropriate choice for the proposed ensemble for this particular data.

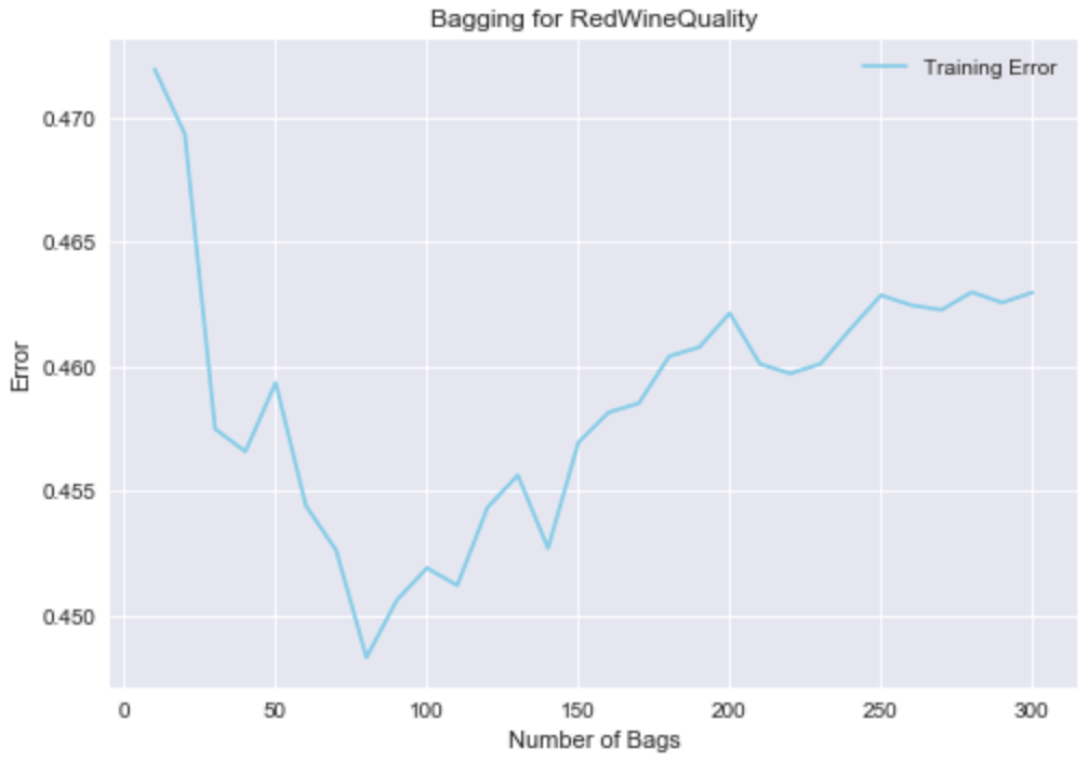
According to the graph, the more the number of bags in the ensemble model, the lower the error you would have in the model. In other words, if you have more data, the accuracy increases. Therefore, I can conclude that bagging is an appropriate choice.

c. (2.5 points) Briefly describe the differences between bagging and boosting.
Bagging samples are drawn with replacement.

Boosting incremetally build an ensemble by training each new model instance to emphasize the training instances that previous model misclassified

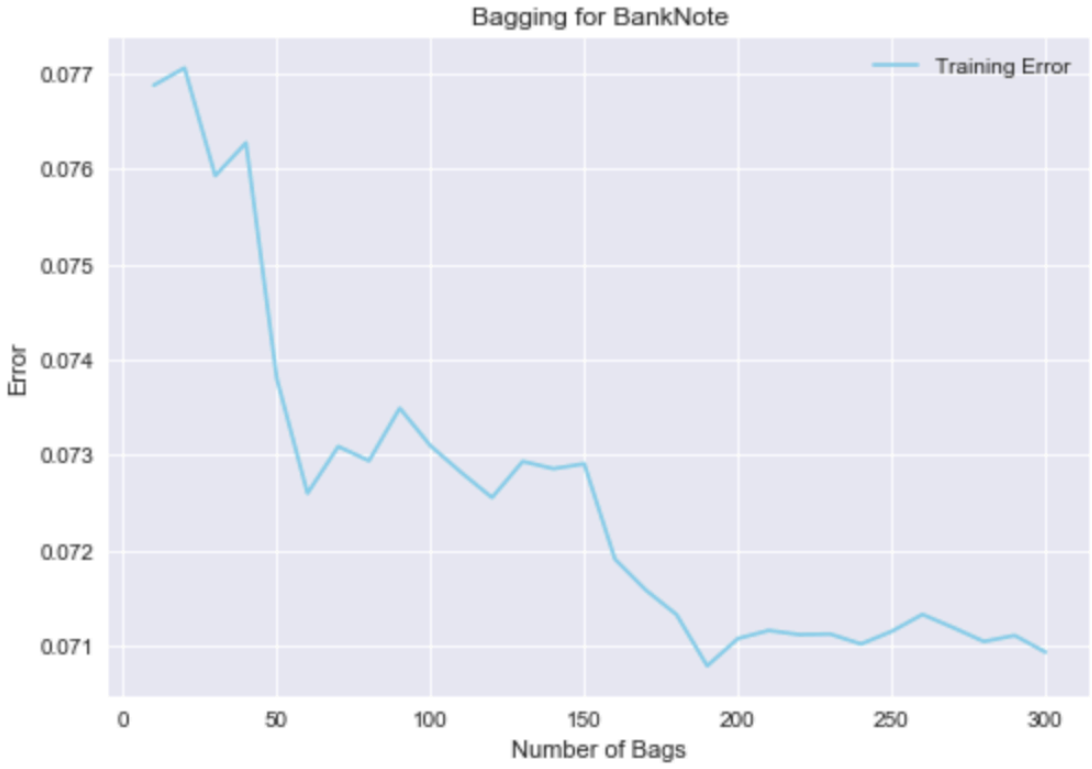
Extra credit (3 points): Repeat Problem 4 on the following additional datasets and interpret the results:

d. winered.data: https://archive.ics.uci.edu/ml/datasets/Wine+Quality



The graph shows that the as number of bags increase the error decreases from 0~80, however, it goes up as the number of bags increases. Thus, the bagging here is not appropriate here.

e. banknote.data: https://archive.ics.uci.edu/ml/datasets/banknote+authentication



This graph shows that as number of bags increase the error decreases. Thus, the bagging here is appropriate.