CSC529 HW2 JONGGOO KANG¶

I did my HW1 with R last time, but I noticed that using loop() in R is kind of time consuming and I do not know how to use iterations in R properly. So I decided to complete HW2 with python.¶

Decision Tree¶

load the modules

In [1]:

```
import os
os.chdir('/Users/jaygkay/Desktop/CSC529')
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from sklearn import tree
from sklearn.cross_validation import train_test_split
from sklearn import preprocessing, naive_bayes, neighbors
from sklearn.tree import DecisionTreeClassifier, export_graphviz
from sklearn.metrics import classification_report, confusion_matrix
from IPython.display import Image, Markdown, display
import statistics
import graphviz
import statsmodels.stats.api as sms
from mlxtend.plotting import plot_learning_curves
from mlxtend.plotting import plot_decision_regions
```

```
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/cross_validation.py:44: DeprecationWarning: This module was deprecated in version 0.18 in favor o
  "This module will be removed in 0.20.", DeprecationWarning)
```

loading the winedata and adding names of the features.

In [2]:

```
col_names = ['class', 'alcohol','malic_acid','ash','ash_alcalinity',
            'magnesium','total_phenols','flavanoids','nonflavanoid_phenols','proanthocyanins',
            'colour','hue','od280_od315','proline']
wine = pd.read_csv('wine.txt', header = None, names = col_names, sep=',')
wine.head(5)
```

Out[2]:

	class	alcohol	malic_acid	ash	ash_alcalinity	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	colour	hue	od280_od315	proline
0	1	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
2	1	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
3	1	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480
4	1	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735

Checking the size of the dataset (dimensions)

In [3]:

```
wine.shape
```

Out[3]:

```
(178, 14)
```

Dividing the dataset into two groups of a dependent variable and independent variables

In [4]:

```
wine_y = wine['class']
wine_x = wine[wine.columns[1:]]
# sizes for y and x
wine_y.shape, wine_x.shape
```

Out[4]:

```
((178,), (178, 13))
```

In [5]:

```
#holdout partitioning with 64% training and 34% testing
x_train, x_test, y_train, y_test = train_test_split(wine_x, wine_y, test_size = 0.34)
x_train.shape, x_test.shape, y_train.shape, y_test.shape
```

Out[5]:

```
((117, 13), (61, 13), (117,), (61,))
```

Decision Tree¶

In [6]:

```
# Initialize Decision Tree model
dtc = DecisionTreeClassifier(criterion = 'entropy', max_depth = 2, min_samples_split = 50)
# Fit the model
dt = dtc.fit(x_train, y_train)
dt
```

Out[6]:

```
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')
```

Visualization

In [7]:

```
export_graphviz(dt, out_file = 'dt.dot', class_names = ["1","2","3"],feature_names = x_train.columns, impurity = False, filled = True)
with open('dt.dot') as f:
    dot_graph = f.read()
graphviz.Source(dot_graph)
```

Out[7]:
proline <= 925.0 samples = 117 value = [39, 47, 31] class = 2 flavanoids <= 1.29 samples = 81 value = [4, 46, 31] class = 2 True samples = 36 value = [35, 1, 0] class = 1 False samples = 34 value = [0, 4, 30] class = 3 samples = 47 value = [4, 42, 1] class = 2

In [8]:

```
# Predict x_test
dt_pred = dt.predict(x_test)
dt_pred
```

Out[8]:

```
array([1, 3, 2, 1, 2, 3, 3, 2, 2, 1, 2, 2, 3, 1, 2, 1, 3, 2, 3, 2, 1, 2, 2,
       2, 1, 2, 3, 2, 2, 2, 3, 1, 3, 2, 1, 1, 2, 3, 2, 2, 1, 3, 2, 2, 1, 2,
       2, 2, 2, 2, 3, 3, 3, 3, 2, 2, 3, 2, 3, 2, 2])
```

In [9]:

```
# classification report
print(classification_report(y_test, dt_pred))
# Accuracy on training
print("Accuracy on training", dt.score(x_train, y_train))
# Accuracy on testing
print("Accuracy on testing", dt.score(x_test, y_test))
# Confusion matrix
print("<<Confusion matrix>>")
print(pd.DataFrame(confusion_matrix(y_test, dt_pred)))
```

```
             precision    recall  f1-score   support

          1       0.92      0.55      0.69        20
          2       0.62      0.83      0.71        24
          3       0.82      0.82      0.82        17

avg / total       0.78      0.74      0.74        61

Accuracy on training 0.91452991453
Accuracy on testing 0.737704918033
<<Confusion matrix>>
    0   1   2
0  11   9   0
1   1  20   3
2   0   3  14
```

Naïve Bayes¶

In [10]:

```
# Initialize Decision Tree model
nbc = naive_bayes.GaussianNB()
# Fit the model
nb = nbc.fit(x_train, y_train)
print(dt)
nb_pred = nb.predict(x_test)
nb_pred
```

```
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')
```

Out[10]:

```
array([1, 3, 2, 1, 2, 3, 3, 1, 2, 1, 1, 2, 3, 1, 3, 1, 3, 2, 2, 2, 1, 2, 2,
       2, 1, 3, 3, 2, 1, 1, 3, 1, 2, 1, 1, 1, 2, 3, 2, 1, 1, 3, 2, 1, 2, 2,
       2, 3, 2, 2, 3, 3, 3, 3, 2, 2, 3, 2, 2, 1, 1])
```

In [11]:

```
# classification report
print(classification_report(y_test, nb_pred))
# Accuracy on training
print("Accuracy on training", nb.score(x_train, y_train))
# Accuracy on testing
print("Accuracy on testing", nb.score(x_test, y_test))
# report
nb_cm = confusion_matrix(y_test, nb_pred)
print("<<Confusion matrix>>")
print(pd.DataFrame(nb_cm))
```

```
             precision    recall  f1-score   support

          1       1.00      1.00      1.00        20
          2       1.00      1.00      1.00        24
          3       1.00      1.00      1.00        17

avg / total       1.00      1.00      1.00        61

Accuracy on training 0.974358974359
Accuracy on testing 1.0
<<Confusion matrix>>
    0   1   2
0  20   0   0
1   0  24   0
2   0   0  17
```

Problem1 - a¶

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.¶

The code I made below assigns 30 different x_train, x_test, y_train, y_test with holdout partitioning 64% training and 34% testing. Furthermore, it also stores 30 differents accuracies on DT training, DT testing, NB trainig, NB testing.

In [12]:

```
dtTrain_lst = []
dtTest_lst = []
nbTrain_lst = []
nbTest_lst = []

for i in range(1,31):
    xtrain, xtest, ytrain, ytest = train_test_split(wine_x, wine_y, test_size = 0.34)
    a = globals()['x_train%d'%i] = xtrain
    b = globals()['x_test%d'%i] = xtest
    c = globals()['y_train%d'%i] = ytrain
    d = globals()['y_test%d'%i] = ytest

    # Decision Tree
    e = globals()['dt%d' %i] = dtc.fit(a, c)
    f = globals()['dttr_scores%d' %i] = e.score(a,c) #accuracy on training
    g = globals()['dtts_scores%d' %i] = e.score(b,d) #accuracy on testing

    # Naïve Bayes
    h = globals()['nb%d' %i] = nbc.fit(a, c)
    i = globals()['nbtr_scores%d' %i] = h.score(a,c) #accuracy on training
    j = globals()['nbts_scores%d' %i] = h.score(b,d) #accuracy on testing

    # Decision Tree accuracies
    k = dtTrain_lst.append(f)
    l = dtTest_lst.append(g)

    # Naïve Bayese accuracies
    m = nbTrain_lst.append(i)
    n = nbTest_lst.append(j)
```

30 Accuracies on DT train and test & NB train and test¶

In [13]:

```
pb1 = pd.DataFrame({"x":range(1,31),"DT Train Acc":dtTrain_lst,"DT Test Acc":dtTest_lst,
                            "NB Train Acc": nbTrain_lst,"NB Test Acc": nbTest_lst })
pb1
```

Out[13]:

	DT Test Acc	DT Train Acc	NB Test Acc	NB Train Acc	x
0	0.836066	0.940171	0.983607	0.991453	1

	DT Test Acc	DT Train Acc	NB Test Acc	NB Train Acc	x
1	0.770492	0.897436	1.000000	0.982906	2
2	0.868852	0.940171	0.967213	0.974359	3
3	0.868852	0.931624	1.000000	0.974359	4
4	0.868852	0.897436	0.967213	1.000000	5
5	0.836066	0.888889	0.967213	0.991453	6
6	0.868852	0.940171	0.967213	1.000000	7
7	0.934426	0.897436	0.983607	0.982906	8
8	0.852459	0.897436	0.934426	0.991453	9
9	0.868852	0.940171	0.983607	0.991453	10
10	0.868852	0.888889	1.000000	0.982906	11
11	0.819672	0.888889	0.983607	0.991453	12
12	0.868852	0.931624	0.934426	1.000000	13
13	0.819672	0.914530	0.983607	0.982906	14
14	0.950820	0.957265	0.983607	0.982906	15
15	0.885246	0.931624	1.000000	0.982906	16
16	0.770492	0.923077	0.934426	0.982906	17
17	0.836066	0.948718	0.950820	0.974359	18
18	0.836066	0.914530	0.967213	0.991453	19
19	0.901639	0.897436	1.000000	0.982906	20
20	0.967213	0.948718	0.950820	0.982906	21
21	0.786885	0.940171	1.000000	0.974359	22
22	0.819672	0.905983	0.967213	0.991453	23
23	0.901639	0.905983	0.967213	0.974359	24
24	0.868852	0.905983	1.000000	0.974359	25
25	0.819672	0.897436	0.950820	0.982906	26
26	0.868852	0.931624	0.983607	0.991453	27
27	0.901639	0.974359	0.983607	0.991453	28
28	0.934426	0.888889	0.983607	0.991453	29
29	0.885246	0.931624	0.983607	0.991453	30

Decision Tree¶

In [14]:

```
print("==============================================================")
print("* Mean for the acuuracy on training sets\n","\t",statistics.mean(dtTrain_lst))
print("* Mean for the acuuracy on testing sets\n","\t",statistics.mean(dtTest_lst))
print("==============================================================")
print("* Variance for the acuuracy on training sets\n","\t", statistics.variance(dtTrain_lst))
print("* Variance for the acuuracy on training sets\n","\t", statistics.variance(dtTest_lst))
print("==============================================================")
dtTrain_CI = sms.DescrStatsW(dtTrain_lst).tconfint_mean()
print("* The 95% CI for DT training sets")
print("\tLow", "\t\t\tHigh")
print(dtTrain_CI)
dtTest_CI = sms.DescrStatsW(dtTest_lst).tconfint_mean()
print("* The 95% CI for DT Testing sets")
print("\tLow", "\t\t\tHigh")
print(dtTest_CI)
print("==============================================================")
```

```
==============================================================
* Mean for the acuuracy on training sets
   0.919943019943
* Mean for the acuuracy on testing sets
   0.862841530055
==============================================================
* Variance for the acuuracy on training sets
   0.000561655763542
* Variance for the acuuracy on training sets
   0.00234425920606
==============================================================
* The 95% CI for DT training sets
 Low    High
(0.9110935642127882, 0.92879247567325174)
* The 95% CI for DT Testing sets
 Low    High
(0.84476211953397928, 0.88092094057531056)
==============================================================
```

Naïve Bayes¶

In [15]:

```
print("==============================================================")
print("* Mean for the acuuracy on training sets\n","\t",statistics.mean(nbTrain_lst))
print("* Mean for the acuuracy on testing sets\n","\t",statistics.mean(nbTest_lst))
print("==============================================================")
print("* Variance for the acuuracy on training sets\n","\t", statistics.variance(nbTrain_lst))
print("* Variance for the acuuracy on training sets\n","\t", statistics.variance(nbTest_lst))
print("==============================================================")
nbTrain_CI = sms.DescrStatsW(nbTrain_lst).tconfint_mean()
print("* The 95% CI for DT training sets")
print("\tLow", "\t\t\tHigh")
print(nbTrain_CI)
nbTest_CI = sms.DescrStatsW(nbTest_lst).tconfint_mean()
print("* The 95% CI for DT training sets")
print("\tLow", "\t\t\tHigh")
print(nbTest_CI)
print("==============================================================")
```

```
==============================================================
* Mean for the acuuracy on training sets
   0.98603988604
* Mean for the acuuracy on testing sets
   0.975409836066
==============================================================
* Variance for the acuuracy on training sets
   6.28913390481e-05
* Variance for the acuuracy on training sets
   0.000421651576792
==============================================================
* The 95% CI for DT training sets
 Low    High
(0.98307862377901722, 0.98900114830075503)
* The 95% CI for DT training sets
 Low    High
(0.96774225607328546, 0.98307741605786203)
==============================================================
```

Problem1-b¶

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. ¶

paired t-test for DT train and NB train¶

In [16]:

```
from scipy.stats import ttest_rel
ttest_rel(dtTrain_lst, nbTrain_lst)
```

Out[16]:

```
Ttest_relResult(statistic=-14.14982499751272, pvalue=1.4980975846195755e-14)
```

According to the result, the p-value is very close to zero 2.06e-12. Thus, the decision is to reject the null hypothesis of the difference between the means is statistically significant. Therefore, I can conclude with that the differnece between the accuracy means from Decision Tree Training and Naïve Bayes Training are very significant as much as 11.59 t-test scores.

paired t-test for DT test and NB test¶

In [17]:

```
ttest_rel(nbTest_lst, dtTest_lst)
```

Out[17]:

```
Ttest_relResult(statistic=12.159065095955716, pvalue=6.5755841676138397e-13)
```

Like previous result, the p-value is very close to zero 8.08e-15. So the decision is to reject the Ho. Thus, I can concclude with that the difference between the accuracy means from Decision Tree Testing and Naïve Bayes Testing are very significant as much as 14.496 t-test socres.

Problem2¶

In [18]:

```
dtTrain_lst2 = []
dtTest_lst2 = []

nbTrain_lst2 = []
nbTest_lst2 = []


prop = 0.25
for i in range(1,8):
    xtrain, xtest, ytrain, ytest = train_test_split(wine_x, wine_y, test_size = prop)
    a = globals()['x2_train%d'%i] = xtrain
    b = globals()['x2_test%d'%i] = xtest
    c = globals()['y2_train%d'%i] = ytrain
    d = globals()['y2_test%d'%i] = ytest

    # Decision Tree
    e = globals()['pb2dt%d' %i] = dtc.fit(a, c)
    f = globals()['pb2dttr_scores%d' %i] = e.score(a,c) #accuracy on training
    g = globals()['pb2dtts_scores%d' %i] = e.score(b,d) #accuracy on testing

    # Naïve Bayes
    h = globals()['pb2nb%d' %i] = nbc.fit(a, c)
    i = globals()['pb2nbtr_scores%d' %i] = h.score(a,c) #accuracy on training
    j = globals()['pb2nbts_scores%d' %i] = h.score(b,d) #accuracy on testing

    # Decision Tree accuracies
    k = dtTrain_lst2.append(f)
    l = dtTest_lst2.append(g)

    # Naïve Bayese accuracies
    m = nbTrain_lst2.append(i)
    n = nbTest_lst2.append(j)
    prop += 0.10
```

In [19]:

```
# Decision tree Accuracies on Training set
size = [133, 115, 97, 80, 62, 44, 26]
pb2ACC = pd.DataFrame({"Training Size":size,"DT Acc Train": dtTrain_lst2, "DT Acc Test": dtTest_lst2,
                    "NB Acc Train": nbTrain_lst2, "NB Acc Test": dtTest_lst2})
pb2ACC
```

Out[19]:

	DT Acc Test	DT Acc Train	NB Acc Test	NB Acc Train	Training Size
0	0.933333	0.902256	0.933333	1.000000	133
1	0.809524	0.886957	0.809524	0.982609	115
2	0.864198	0.948454	0.864198	0.989691	97
3	0.877551	0.950000	0.877551	0.987500	80
4	0.551724	0.709677	0.551724	0.983871	62
5	0.231343	0.386364	0.231343	1.000000	44
6	0.388158	0.461538	0.388158	1.000000	26

Resubstitution errors and Generalization errors from DT and NB¶

In [20]:

```
# Resubstitution error on Decision Tree on Training set
dtTrain_reErr = []
dtTest_reErr = []
nbTrain_geErr = []
nbTest_geErr = []

for i in dtTrain_lst2:
    dtTrain_reErr.append(1-i)
for i in dtTest_lst2:
    dtTest_reErr.append(1-i)
for i in nbTrain_lst2:
    nbTrain_geErr.append(1-i)
for i in nbTest_lst2:
    nbTest_geErr.append(1-i)
```

In [21]:

```
# Decision tree Accuracies on Training set
pb2Err = pd.DataFrame({"Training Size":size, "DT Resubstituition Error": dtTrain_reErr,
                       "DT Generalization Error": dtTest_reErr,
                       "NB Resubstituition Error": nbTrain_geErr,
                       "NB Generalization Error": nbTest_geErr})
pb2Err
```

Out[21]:

	DT Generalization Error	DT Resubstituition Error	NB Generalization Error	NB Resubstituition Error	Training Size
0	0.066667	0.097744	0.066667	0.000000	133
1	0.190476	0.113043	0.000000	0.017391	115
2	0.135802	0.051546	0.024691	0.010309	97
3	0.122449	0.050000	0.040816	0.012500	80
4	0.448276	0.290323	0.034483	0.016129	62
5	0.768657	0.613636	0.022388	0.000000	44
6	0.611842	0.538462	0.065789	0.000000	26

In [22]:

```
import matplotlib.pyplot as plt
plt.plot('Training Size','DT Generalization Error', data = pb2Err, marker ='', color = 'skyblue' )
plt.plot('Training Size','DT Resubstituition Error', data = pb2Err, marker = '', color = 'skyblue', linestyle = 'dashed')
plt.plot('Training Size','NB Generalization Error', data = pb2Err, marker = '', color = 'olive')
plt.plot('Training Size','NB Resubstituition Error', data = pb2Err, marker = '', color = 'olive', linestyle = 'dashed')
plt.xlabel("Training Set Size")
plt.ylabel("Errors")
plt.title("DT & NB Errors vs Size of Training dataset")
plt.legend()
```

Out[22]:

```
<matplotlib.legend.Legend at 0x113fa2da0>
```

According to the observed performance, I can conclude that with an large training set, Errors for Decision Tree and Naïve Bayes decrease. In other words, with an large training set, all algorithms' Accuracy increase.

Extra credit (winered.data)¶

In [23]:

```
redwine = pd.read_csv('redwine.csv')
redwine.head()
```

Out[23]:

	Unnamed: 0	fixed.acidity	volatile.acidity	citric.acid	residual.sugar	chlorides	free.sulfur.dioxide	total.sulfur.dioxide	density	pH	sulphates	alcohol	quality
0	1	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	2	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	3	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	4	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	5	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

In [24]:

```
redwine = redwine.drop(['Unnamed: 0'],axis=1)
```

In [25]:

```
redwine.shape
```

Out[25]:

```
(1599, 12)
```

In [26]:

```
y_ex1 = pd.DataFrame(redwine['quality'])
x_ex1 = pd.DataFrame(redwine[redwine.columns[:11]])
```

In [27]:

```
exTrain_lst1 = []
exTest_lst1 = []

exTrain_lst2 = []
exTest_lst2 = []


prop = 0.25
for i in range(1,8):
    xtrain, xtest, ytrain, ytest = train_test_split(x_ex1, y_ex1, test_size = prop)
    a = globals()['eX_train%d'%i] = xtrain
    b = globals()['eX_test%d'%i] = xtest
    c = globals()['eY_train%d'%i] = ytrain
    d = globals()['eY_test%d'%i] = ytest

    # Decision Tree
    e = globals()['pb2dt%d' %i] = dtc.fit(a, c)
    f = globals()['pb2dttr_scores%d' %i] = e.score(a,c) #accuracy on training
    g = globals()['pb2dtts_scores%d' %i] = e.score(b,d) #accuracy on testing

    # Naïve Bayes
    h = globals()['pb2nb%d' %i] = nbc.fit(a, c)
    i = globals()['pb2nbtr_scores%d' %i] = h.score(a,c) #accuracy on training
    j = globals()['pb2nbts_scores%d' %i] = h.score(b,d) #accuracy on testing

    # Decision Tree accuracies
    k = exTrain_lst1.append(f)
    l = exTest_lst1.append(g)

    # Naïve Bayese accuracies
    m = exTrain_lst2.append(i)
    n = exTest_lst2.append(j)
    prop += 0.10
```

```
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

In [28]:

```
# Decision tree Accuracies on Training set
size = [400, 560, 719, 879, 1040, 1199, 1360]
exACC = pd.DataFrame({"Training Size":size,"DT Acc Train": exTrain_lst1, "DT Acc Test": exTest_lst1,
                    "NB Acc Train": exTrain_lst2, "NB Acc Test": exTest_lst2})
exACC
```

Out[28]:

	DT Acc Test	DT Acc Train	NB Acc Test	NB Acc Train	Training Size
0	0.552500	0.553795	0.530000	0.559633	400
1	0.573214	0.542830	0.517857	0.529355	560
2	0.554167	0.551763	0.536111	0.568828	719
3	0.568182	0.535466	0.531818	0.581363	879
4	0.522115	0.572451	0.508654	0.568873	1040
5	0.545833	0.561404	0.505000	0.541353	1199
6	0.554412	0.548117	0.515441	0.556485	1360

In [29]:

```
# Resubstitution error on Decision Tree on Training set
dtTrain_reErr = []
dtTest_reErr = []
nbTrain_geErr = []
nbTest_geErr = []

for i in exTrain_lst1:
    dtTrain_reErr.append(1-i)
for i in exTest_lst1:
    dtTest_reErr.append(1-i)
for i in exTrain_lst2:
    nbTrain_geErr.append(1-i)
for i in exTest_lst2:
    nbTest_geErr.append(1-i)
```

In [30]:

```
# Decision tree Accuracies on Training set
exErr = pd.DataFrame({"Training Size":size, "DT Resubstituition Error": dtTrain_reErr,
                       "DT Generalization Error": dtTest_reErr,
                       "NB Resubstituition Error": nbTrain_geErr,
```

```
                       "NB Generalization Error": nbTest_geErr})
exErr
```

Out[30]:

	DT Generalization Error	DT Resubstituition Error	NB Generalization Error	NB Resubstituition Error	Training Size
0	0.447500	0.446205	0.470000	0.440367	400
1	0.426786	0.457170	0.482143	0.470645	560
2	0.445833	0.448237	0.463889	0.431172	719
3	0.431818	0.464534	0.468182	0.418637	879
4	0.477885	0.427549	0.491346	0.431127	1040
5	0.454167	0.438596	0.495000	0.458647	1199
6	0.445588	0.451883	0.484559	0.443515	1360

In [31]:

```
import matplotlib.pyplot as plt
plt.plot('Training Size','DT Generalization Error', data = exErr, marker ='', color = 'skyblue' )
plt.plot('Training Size','DT Resubstituition Error', data = exErr, marker = '', color = 'skyblue', linestyle = 'dashed')
plt.plot('Training Size','NB Generalization Error', data = exErr, marker = '', color = 'olive')
plt.plot('Training Size','NB Resubstituition Error', data = exErr, marker = '', color = 'olive', linestyle = 'dashed')
plt.xlabel("Training Set Size")
plt.ylabel("Errors")
plt.title("DT & NB Errors vs Size of WineQuality-Red dataset")
plt.legend()
```

Out[31]:

```
<matplotlib.legend.Legend at 0x1142efeb8>
```

In [ ]:

In [ ]:

Extra 2. Banknote.data¶

In [32]:

```
bank = pd.read_csv('bank.csv')
bank.head()
```

Out[32]:

	3.6216	8.6661	-2.8073	-0.44699	0
0	4.54590	8.1674	-2.4586	-1.46210	0
1	3.86600	-2.6383	1.9242	0.10645	0
2	3.45660	9.5228	-4.0112	-3.59440	0
3	0.32924	-4.4552	4.5718	-0.98880	0
4	4.36840	9.6718	-3.9606	-3.16250	0

In [33]:

```
bank.shape
```

Out[33]:

```
(1371, 5)
```

In [34]:

```
y_ex2 = pd.DataFrame(bank['0'])
x_ex2 = pd.DataFrame(bank[bank.columns[:3]])
```

In [35]:

```
ex1Train_lst1 = []
ex1Test_lst1 = []

ex1Train_lst2 = []
ex1Test_lst2 = []


prop = 0.25
for i in range(1,8):
    xtrain, xtest, ytrain, ytest = train_test_split(x_ex2, y_ex2, test_size = prop)
    a = globals()['eX1_train%d'%i] = xtrain
    b = globals()['eX1_test%d'%i] = xtest
    c = globals()['eY1_train%d'%i] = ytrain
    d = globals()['eY1_test%d'%i] = ytest

    # Decision Tree
    e = globals()['pb2dt%d' %i] = dtc.fit(a, c)
    f = globals()['pb2dttr_scores%d' %i] = e.score(a,c) #accuracy on training
    g = globals()['pb2dtts_scores%d' %i] = e.score(b,d) #accuracy on testing

    # Naïve Bayes
    h = globals()['pb2nb%d' %i] = nbc.fit(a, c)
    i = globals()['pb2nbtr_scores%d' %i] = h.score(a,c) #accuracy on training
    j = globals()['pb2nbts_scores%d' %i] = h.score(b,d) #accuracy on testing

    # Decision Tree accuracies
    k = ex1Train_lst1.append(f)
    l = ex1Test_lst1.append(g)

    # Naïve Bayese accuracies
    m = ex1Train_lst2.append(i)
    n = ex1Test_lst2.append(j)
    prop += 0.10
```

```
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

In [36]:

```
# Decision tree Accuracies on Training set
size = [342, 480, 617, 754, 891, 1028, 1166]
ex1ACC = pd.DataFrame({"Training Size":size, "DT Acc Train": ex1Train_lst1, "DT Acc Test": ex1Test_lst1,
                    "NB Acc Train": ex1Train_lst2, "NB Acc Test": ex1Test_lst2})
ex1ACC
```

Out[36]:

	DT Acc Test	DT Acc Train	NB Acc Test	NB Acc Train	Training Size
0	0.892128	0.895914	0.819242	0.843385	342
1	0.862500	0.893378	0.845833	0.833895	480
2	0.896272	0.912467	0.854133	0.832891	617
3	0.895364	0.904221	0.859603	0.857143	754
4	0.874439	0.910230	0.820628	0.864301	891
5	0.889213	0.926901	0.840622	0.871345	1028
6	0.876501	0.892683	0.835334	0.843902	1166

In [37]:

```
# Resubstitution error on Decision Tree on Training set
dt1Train_reErr = []
dt1Test_reErr = []
nb1Train_geErr = []
nb1Test_geErr = []

for i in ex1Train_lst1:
    dt1Train_reErr.append(1-i)
for i in ex1Test_lst1:
    dt1Test_reErr.append(1-i)
for i in ex1Train_lst2:
    nb1Train_geErr.append(1-i)
for i in ex1Test_lst2:
    nb1Test_geErr.append(1-i)
```

In [38]:

```
# Decision tree Accuracies on Training set
ex1Err = pd.DataFrame({"Training Size":size, "DT Resubstituition Error": dt1Train_reErr,
                       "DT Generalization Error": dt1Test_reErr,
                       "NB Resubstituition Error": nb1Train_geErr,
                       "NB Generalization Error": nb1Test_geErr})
ex1Err
```

Out[38]:

	DT Generalization Error	DT Resubstituition Error	NB Generalization Error	NB Resubstituition Error	Training Size
0	0.107872	0.104086	0.180758	0.156615	342
1	0.137500	0.106622	0.154167	0.166105	480
2	0.103728	0.087533	0.145867	0.167109	617
3	0.104636	0.095779	0.140397	0.142857	754
4	0.125561	0.089770	0.179372	0.135699	891
5	0.110787	0.073099	0.159378	0.128655	1028
6	0.123499	0.107317	0.164666	0.156098	1166

In [39]:

```
import matplotlib.pyplot as plt
plt.plot('Training Size','DT Generalization Error', data = ex1Err, marker ='', color = 'skyblue' )
plt.plot('Training Size','DT Resubstituition Error', data = ex1Err, marker = '', color = 'skyblue', linestyle = 'dashed')
plt.plot('Training Size','NB Generalization Error', data = ex1Err, marker = '', color = 'olive')
plt.plot('Training Size','NB Resubstituition Error', data = ex1Err, marker = '', color = 'olive', linestyle = 'dashed')
plt.xlabel("Training Set Size")
plt.ylabel("Errors")
plt.title("DT & NB Errors vs Size of BankNote dataset")
plt.legend()
```

Out[39]:

```
<matplotlib.legend.Legend at 0x114261cc0>
```

In [ ]:

Problem3¶

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).¶

Naïve Bayes with two calsses¶

In [40]:

```
wine.tail()
```

Out[40]:

	class	alcohol	malic_acid	ash	ash_alcalinity	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	colour	hue	od280_od315	proline
173	3	13.71	5.65	2.45	20.5	95	1.68	0.61	0.52	1.06	7.7	0.64	1.74	740
174	3	13.40	3.91	2.48	23.0	102	1.80	0.75	0.43	1.41	7.3	0.70	1.56	750
175	3	13.27	4.28	2.26	20.0	120	1.59	0.69	0.43	1.35	10.2	0.59	1.56	835
176	3	13.17	2.59	2.37	20.0	120	1.65	0.68	0.53	1.46	9.3	0.60	1.62	840
177	3	14.13	4.10	2.74	24.5	96	2.05	0.76	0.56	1.35	9.2	0.61	1.60	560

In [41]:

```
wine['class'] = wine['class'].map({1:'p',2:'n',3:'n'})
wine.tail()
```

Out[41]:

	class	alcohol	malic_acid	ash	ash_alcalinity	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	colour	hue	od280_od315	proline
173	n	13.71	5.65	2.45	20.5	95	1.68	0.61	0.52	1.06	7.7	0.64	1.74	740
174	n	13.40	3.91	2.48	23.0	102	1.80	0.75	0.43	1.41	7.3	0.70	1.56	750
175	n	13.27	4.28	2.26	20.0	120	1.59	0.69	0.43	1.35	10.2	0.59	1.56	835
176	n	13.17	2.59	2.37	20.0	120	1.65	0.68	0.53	1.46	9.3	0.60	1.62	840
177	n	14.13	4.10	2.74	24.5	96	2.05	0.76	0.56	1.35	9.2	0.61	1.60	560

In [42]:

```
wine_y_pb3 = wine['class']
wine_x_pb3= wine[wine.columns[1:]]
# sizes for y and x
wine_y_pb3.shape, wine_x_pb3.shape
```

Out[42]:

```
((178,), (178, 13))
```

In [43]:

```
#holdout partitioning with 64% training and 34% testing
x_train_pb3, x_test_pb3, y_train_pb3, y_test_pb3 = train_test_split(wine_x_pb3, wine_y_pb3, test_size = 0.33)
x_train_pb3.shape, x_test_pb3.shape, y_train_pb3.shape, y_test_pb3.shape
```

Out[43]:

```
((119, 13), (59, 13), (119,), (59,))
```

In [44]:

```
# Initialize Decision Tree model
nbc_pb3 = naive_bayes.GaussianNB()
# Fit the model
nb_pb3 = nbc_pb3.fit(x_train_pb3, y_train_pb3)
print(dt)
nb_pred_pb3 = nb_pb3.predict(x_test_pb3)
nb_pred_pb3
```

```
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')
```

Out[44]:

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

In [45]:

```
# classification report
print(classification_report(y_test_pb3, nb_pred_pb3))
# Accuracy on training
print("Accuracy on training", nb_pb3.score(x_train_pb3, y_train_pb3))
# Accuracy on testing
print("Accuracy on testing", nb_pb3.score(x_test_pb3, y_test_pb3))
# report
print(pd.DataFrame(confusion_matrix(y_test_pb3, nb_pred_pb3)))
```

```
             precision    recall  f1-score   support

          n       0.95      1.00      0.97        38
          p       1.00      0.90      0.95        21

avg / total       0.97      0.97      0.97        59

Accuracy on training 0.991596638655
Accuracy on testing 0.966101694915
    0   1
0  38   0
1   2  19
```

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¶

In [46]:

```
import scikitplot as skplt
pred_prob_pb3 = nb_pb3.predict_proba(x_train_pb3)

plt.hist(pred_prob_pb3, bins = 8)
plt.xlim(0,1)
plt.title('Histogram of predicted probabilities')
plt.xlabel('Predicted probabilities')
plt.ylabel('Frequency')
```

Out[46]:

```
<matplotlib.text.Text at 0x11494f5f8>
```

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.

In [47]:

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

Out[47]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x114943908>
```

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.

ROC curve for Naïve Bayes performance on Testing data¶

In [48]:

```
pred_probas_pb3 = nb_pb3.predict_proba(x_test_pb3)

plt.hist(pred_probas_pb3, bins = 8)
plt.xlim(0,1)
plt.title('Histogram of predicted probabilities')
plt.xlabel('Predicted probabilities')
plt.ylabel('Frequency')
```

Out[48]:

```
<matplotlib.text.Text at 0x114e3ab38>
```

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.

In [49]:

```
skplt.metrics.plot_roc_curve(y_test_pb3, pred_probas_pb3)
```

Out[49]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x114e8afd0>
```

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.

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¶

In [50]:

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

Out[50]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x114f15a20>
```

Lift curve for Naïve Bayes performance on Testing data¶

In [51]:

```
skplt.metrics.plot_lift_curve(y_test_pb3, pred_probas_pb3)
```

Out[51]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11507d550>
```

Lift curve shows that the effectiveness of a binary classifer. Here, by 0% of class 2&3 (negative) will be chosen based on the predictive model, we will get almost 3 times more positive class (class1)

Problem 4¶

In [52]:

```
df = pd.read_csv('data.csv')
df.head()
```

Out[52]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texture_worst	perimeter_worst	area_worst	smoothness
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	...	17.33	184.60	2019.0	0.1622
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	...	23.41	158.80	1956.0	0.1238
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	...	25.53	152.50	1709.0	0.1444
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	...	26.50	98.87	567.7	0.2098
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	...	16.67	152.20	1575.0	0.1374

5 rows × 33 columns

In [53]:

```
df.shape
```

Out[53]:

```
(569, 33)
```

In [54]:

```
df.columns
```

Out[54]:

```
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst', 'Unnamed: 32'],
      dtype='object')
```

In [55]:

```
df = df.drop(['Unnamed: 32','id'],axis=1)
df
```

Out[55]:

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	...	radius_worst	texture_worst	perimeter_worst
0	M	17.990	10.38	122.80	1001.0	0.11840	0.27760	0.300100	0.147100	0.2419	...	25.380	17.33	184.60
1	M	20.570	17.77	132.90	1326.0	0.08474	0.07864	0.086900	0.070170	0.1812	...	24.990	23.41	158.80
2	M	19.690	21.25	130.00	1203.0	0.10960	0.15990	0.197400	0.127900	0.2069	...	23.570	25.53	152.50
3	M	11.420	20.38	77.58	386.1	0.14250	0.28390	0.241400	0.105200	0.2597	...	14.910	26.50	98.87
4	M	20.290	14.34	135.10	1297.0	0.10030	0.13280	0.198000	0.104300	0.1809	...	22.540	16.67	152.20
5	M	12.450	15.70	82.57	477.1	0.12780	0.17000	0.157800	0.080890	0.2087	...	15.470	23.75	103.40
6	M	18.250	19.98	119.60	1040.0	0.09463	0.10900	0.112700	0.074000	0.1794	...	22.880	27.66	153.20
7	M	13.710	20.83	90.20	577.9	0.11890	0.16450	0.093660	0.059850	0.2196	...	17.060	28.14	110.60
8	M	13.000	21.82	87.50	519.8	0.12730	0.19320	0.185900	0.093530	0.2350	...	15.490	30.73	106.20
9	M	12.460	24.04	83.97	475.9	0.11860	0.23960	0.227300	0.085430	0.2030	...	15.090	40.68	97.65
10	M	16.020	23.24	102.70	797.8	0.08206	0.06669	0.032990	0.033230	0.1528	...	19.190	33.88	123.80
11	M	15.780	17.89	103.60	781.0	0.09710	0.12920	0.099540	0.066060	0.1842	...	20.420	27.28	136.50
12	M	19.170	24.80	132.40	1123.0	0.09740	0.24580	0.206500	0.111800	0.2397	...	20.960	29.94	151.70
13	M	15.850	23.95	103.70	782.7	0.08401	0.10020	0.099380	0.053640	0.1847	...	16.840	27.66	112.00
14	M	13.730	22.61	93.60	578.3	0.11310	0.22930	0.212800	0.080250	0.2069	...	15.030	32.01	108.80
15	M	14.540	27.54	96.73	658.8	0.11390	0.15950	0.163900	0.073640	0.2303	...	17.460	37.13	124.10
16	M	14.680	20.13	94.74	684.5	0.09867	0.07200	0.073950	0.052590	0.1586	...	19.070	30.88	123.40
17	M	16.130	20.68	108.10	798.8	0.11700	0.20220	0.172200	0.102800	0.2164	...	20.960	31.48	136.80
18	M	19.810	22.15	130.00	1260.0	0.09831	0.10270	0.147900	0.094980	0.1582	...	27.320	30.88	186.80
19	B	13.540	14.36	87.46	566.3	0.09779	0.08129	0.066640	0.047810	0.1885	...	15.110	19.26	99.70
20	B	13.080	15.71	85.63	520.0	0.10750	0.12700	0.045680	0.031100	0.1967	...	14.500	20.49	96.09
21	B	9.504	12.44	60.34	273.9	0.10240	0.06492	0.029560	0.020760	0.1815	...	10.230	15.66	65.13
22	M	15.340	14.26	102.50	704.4	0.10730	0.21350	0.207700	0.097560	0.2521	...	18.070	19.08	125.10
23	M	21.160	23.04	137.20	1404.0	0.09428	0.10220	0.109700	0.086320	0.1769	...	29.170	35.59	188.00
24	M	16.650	21.38	110.00	904.6	0.11210	0.14570	0.152500	0.091700	0.1995	...	26.460	31.56	177.00
25	M	17.140	16.40	116.00	912.7	0.11860	0.22760	0.222900	0.140100	0.3040	...	22.250	21.40	152.40
26	M	14.580	21.53	97.41	644.8	0.10540	0.18680	0.142500	0.087830	0.2252	...	17.620	33.21	122.40
27	M	18.610	20.25	122.10	1094.0	0.09440	0.10660	0.149000	0.077310	0.1697	...	21.310	27.26	139.90
28	M	15.300	25.27	102.40	732.4	0.10820	0.16970	0.168300	0.087510	0.1926	...	20.270	36.71	149.30
29	M	17.570	15.05	115.00	955.1	0.09847	0.11570	0.098750	0.079530	0.1739	...	20.010	19.52	134.90
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
539	B	7.691	25.44	48.34	170.4	0.08668	0.11990	0.092520	0.013640	0.2037	...	8.678	31.89	54.49
540	B	11.540	14.44	74.65	402.9	0.09984	0.11200	0.067370	0.025940	0.1818	...	12.260	19.68	78.78
541	B	14.470	24.99	95.81	656.4	0.08837	0.12300	0.100900	0.038900	0.1872	...	16.220	31.73	113.50
542	B	14.740	25.42	94.70	668.6	0.08275	0.07214	0.041050	0.030270	0.1840	...	16.510	32.29	107.40
543	B	13.210	28.06	84.88	538.4	0.08671	0.06877	0.029870	0.032750	0.1628	...	14.370	37.17	92.48
544	B	13.870	20.70	89.77	584.8	0.09578	0.10180	0.036880	0.023690	0.1620	...	15.050	24.75	99.17
545	B	13.620	23.23	87.19	573.2	0.09246	0.06747	0.029740	0.024430	0.1664	...	15.350	29.09	97.58
546	B	10.320	16.35	65.31	324.9	0.09434	0.04994	0.010120	0.005495	0.1885	...	11.250	21.77	71.12
547	B	10.260	16.58	65.85	320.8	0.08877	0.08066	0.043580	0.024380	0.1669	...	10.830	22.04	71.08
548	B	9.683	19.34	61.05	285.7	0.08491	0.05030	0.023370	0.009615	0.1580	...	10.930	25.59	69.10
549	B	10.820	24.21	68.89	361.6	0.08192	0.06602	0.015480	0.008160	0.1976	...	13.030	31.45	83.90
550	B	10.860	21.48	68.51	360.5	0.07431	0.04227	0.000000	0.000000	0.1661	...	11.660	24.77	74.08
551	B	11.130	22.44	71.49	378.4	0.09566	0.08194	0.048240	0.022570	0.2030	...	12.020	28.26	77.80
552	B	12.770	29.43	81.35	507.9	0.08276	0.04234	0.019970	0.014990	0.1539	...	13.870	36.00	88.10
553	B	9.333	21.94	59.01	264.0	0.09240	0.05605	0.039960	0.012820	0.1692	...	9.845	25.05	62.86
554	B	12.880	28.92	82.50	514.3	0.08123	0.05824	0.061950	0.023430	0.1566	...	13.890	35.74	88.84
555	B	10.290	27.61	65.67	321.4	0.09030	0.07658	0.059990	0.027380	0.1593	...	10.840	34.91	69.57
556	B	10.160	19.59	64.73	311.7	0.10030	0.07504	0.005025	0.011160	0.1791	...	10.650	22.88	67.88
557	B	9.423	27.88	59.26	271.3	0.08123	0.04971	0.000000	0.000000	0.1742	...	10.490	34.24	66.50
558	B	14.590	22.68	96.39	657.1	0.08473	0.13300	0.102900	0.037360	0.1454	...	15.480	27.27	105.90
559	B	11.510	23.93	74.52	403.5	0.09261	0.10210	0.111200	0.041050	0.1388	...	12.480	37.16	82.28

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	...	radius_worst	texture_worst	perimeter_worst
560	B	14.050	27.15	91.38	600.4	0.09929	0.11260	0.044620	0.043040	0.1537	...	15.300	33.17	100.20
561	B	11.200	29.37	70.67	386.0	0.07449	0.03558	0.000000	0.000000	0.1060	...	11.920	38.30	75.19
562	M	15.220	30.62	103.40	716.9	0.10480	0.20870	0.255000	0.094290	0.2128	...	17.520	42.79	128.70
563	M	20.920	25.09	143.00	1347.0	0.10990	0.22360	0.317400	0.147400	0.2149	...	24.290	29.41	179.10
564	M	21.560	22.39	142.00	1479.0	0.11100	0.11590	0.243900	0.138900	0.1726	...	25.450	26.40	166.10
565	M	20.130	28.25	131.20	1261.0	0.09780	0.10340	0.144000	0.097910	0.1752	...	23.690	38.25	155.00
566	M	16.600	28.08	108.30	858.1	0.08455	0.10230	0.092510	0.053020	0.1590	...	18.980	34.12	126.70
567	M	20.600	29.33	140.10	1265.0	0.11780	0.27700	0.351400	0.152000	0.2397	...	25.740	39.42	184.60
568	B	7.760	24.54	47.92	181.0	0.05263	0.04362	0.000000	0.000000	0.1587	...	9.456	30.37	59.16

569 rows × 31 columns

In [56]:

```
df['diagnosis'] = df['diagnosis'].replace('M',1)
df['diagnosis'] = df['diagnosis'].replace('B',0)
```

In [57]:

```
y_pb4 = pd.DataFrame(df['diagnosis'])
x_pb4 = pd.DataFrame(df[df.columns[1:]])
```

In [58]:

```
# plot learning curves
x_train_pb4, x_test_pb4, y_train_pb4, y_test_pb4 = train_test_split(x_pb4, y_pb4, test_size = 0.34, random_state = 1)
x_train_pb4.shape, x_test_pb4.shape, y_train_pb4.shape, y_test_pb4.shape
```

Out[58]:

```
((375, 30), (194, 30), (375, 1), (194, 1))
```

In [59]:

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import BaggingClassifier
from mlxtend.plotting import plot_learning_curves
from mlxtend.plotting import plot_decision_regions
from sklearn.model_selection import cross_val_score
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import BaggingRegressor

clf = DecisionTreeClassifier(criterion = 'entropy', max_depth = 2)

random_state = 30
n_estimators = 300
step_factor = 10
axis_step = int(n_estimators/step_factor)

estimators = np.zeros(axis_step)
bagging_mse = np.zeros(axis_step)
bagging_mse_ts= np.zeros(axis_step)

est_lst = []
acc_tr = []
acc_ts = []


for i in range(0,axis_step):
    print("Bagging Estimator: %d of %d..."%(step_factor*(i+1), n_estimators))
    bag = BaggingRegressor(clf, n_estimators =step_factor*(i+1), n_jobs = 1, random_state = random_state)
    a= bag.fit(x_train_pb4, y_train_pb4)
    b= mean_squared_error(y_test_pb4, bag.predict(x_test_pb4))
    c = estimators[i] = step_factor*(i+1)
    d = bagging_mse[i] = b

    e = bag.fit(x_test_pb4, y_test_pb4)
    f = mean_squared_error(y_test_pb4, bag.predict(x_test_pb4))
    g = bagging_mse_ts[i] = f

    est_lst.append(c)
    acc_tr.append(d)
    #acc_ts.append(g)
```

```
Bagging Estimator: 10 of 300...
Bagging Estimator: 20 of 300...
Bagging Estimator: 30 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 40 of 300...
Bagging Estimator: 50 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 60 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 70 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 80 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 90 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

```
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 100 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 110 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 120 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 130 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 140 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 150 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 160 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 170 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 180 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 190 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 200 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 210 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 220 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 230 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 240 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 250 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 260 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 270 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 280 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 290 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 300 of 300...
```

```
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

In [60]:

```
pb4 = pd.DataFrame({'Number of Bags':est_lst, 'Training Error':acc_tr})#,'Testing Error':acc_tr})
pb4
```

Out[60]:

	Number of Bags	Training Error
0	10	0.088969
1	20	0.078247
2	30	0.077635
3	40	0.078302
4	50	0.078454
5	60	0.076831
6	70	0.077273
7	80	0.073540
8	90	0.073280
9	100	0.072424
10	110	0.071946
11	120	0.072723
12	130	0.073753
13	140	0.073941
14	150	0.074306
15	160	0.073291
16	170	0.072636
17	180	0.073512
18	190	0.074754
19	200	0.074591
20	210	0.074673
21	220	0.074705
22	230	0.074457
23	240	0.074800
24	250	0.075171
25	260	0.075373
26	270	0.075171
27	280	0.075554
28	290	0.075323
29	300	0.075497

In [61]:

```
plt.plot('Number of Bags','Training Error', data = pb4, marker = '', color = 'skyblue' )
#plt.plot('Number of Bags','Testing Error', data = pb4, marker = '', color = 'olive' )
plt.xlabel("Number of Bags")
plt.ylabel("Error")
plt.title("Bagging")
plt.legend()
```

Out[61]:

```
<matplotlib.legend.Legend at 0x1157ba5f8>
```

b. 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. 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 with RedWineQuality¶

In [62]:

```
redwine.head()
```

Out[62]:

	fixed.acidity	volatile.acidity	citric.acid	residual.sugar	chlorides	free.sulfur.dioxide	total.sulfur.dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

In [63]:

```
y_ex11 = pd.DataFrame(redwine['quality'])
x_ex11 = pd.DataFrame(redwine[redwine.columns[:11]])
# plot learning curves
x_train_ex4, x_test_ex4, y_train_ex4, y_test_ex4 = train_test_split(x_ex11, y_ex11, test_size = 0.34, random_state = 1)
x_train_ex4.shape, x_test_ex4.shape, y_train_ex4.shape, y_test_ex4.shape
```

Out[63]:

```
((1055, 11), (544, 11), (1055, 1), (544, 1))
```

In [64]:

```
clf = DecisionTreeClassifier(criterion = 'entropy', max_depth = 2)

random_state = 30
n_estimators = 300
step_factor = 10
axis_step = int(n_estimators/step_factor)

estimators = np.zeros(axis_step)
bagging_mse = np.zeros(axis_step)
bagging_mse_ts= np.zeros(axis_step)

est_lst = []
acc_tr = []
acc_ts = []
```

```
for i in range(0,axis_step):
    print("Bagging Estimator: %d of %d..."%(step_factor*(i+1), n_estimators))
    bag = BaggingRegressor(clf, n_estimators =step_factor*(i+1), n_jobs = 1, random_state = random_state)
    a= bag.fit(x_train_ex4, y_train_ex4)
    b= mean_squared_error(y_test_ex4, bag.predict(x_test_ex4))
    c = estimators[i] = step_factor*(i+1)
    d = bagging_mse[i] = b

    e = bag.fit(x_test_ex4, y_test_ex4)
    f = mean_squared_error(y_test_ex4, bag.predict(x_test_ex4))
    g = bagging_mse_ts[i] = f

    est_lst.append(c)
    acc_tr.append(d)
    #acc_ts.append(g)
```

```
Bagging Estimator: 10 of 300...
Bagging Estimator: 20 of 300...
Bagging Estimator: 30 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 40 of 300...
Bagging Estimator: 50 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 60 of 300...
Bagging Estimator: 70 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 80 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 90 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 100 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 110 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 120 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 130 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 140 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 150 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 160 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 170 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 180 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 190 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

```
Bagging Estimator: 200 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 210 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 220 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 230 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 240 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 250 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 260 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 270 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 280 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 290 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 300 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

In [65]:

```
ex4 = pd.DataFrame({'Number of Bags':est_lst, 'Training Error':acc_tr})#,'Testing Error':acc_tr})
ex4
```

Out[65]:

	Number of Bags	Training Error
0	10	0.471930
1	20	0.469320
2	30	0.457498
3	40	0.456589
4	50	0.459351
5	60	0.454415
6	70	0.452634
7	80	0.448331
8	90	0.450626
9	100	0.451922
10	110	0.451222
11	120	0.454348
12	130	0.455643
13	140	0.452714
14	150	0.456953
15	160	0.458160
16	170	0.458525
17	180	0.460410
18	190	0.460778
19	200	0.462141
20	210	0.460100
21	220	0.459717
22	230	0.460108
23	240	0.461515
24	250	0.462858
25	260	0.462460
26	270	0.462269
27	280	0.462984
28	290	0.462560
29	300	0.462967

In [66]:

```
plt.plot('Number of Bags','Training Error', data = ex4, marker = '', color = 'skyblue' )
#plt.plot('Number of Bags','Testing Error', data = pb4, marker = '', color = 'olive' )
plt.xlabel("Number of Bags")
plt.ylabel("Error")
plt.title("Bagging for RedWineQuality")
```

```
plt.legend()
```

Out[66]:

```
<matplotlib.legend.Legend at 0x115575eb8>
```

Extra with Banknote¶

In [67]:

```
print(bank.head())
y_ex22 = pd.DataFrame(bank['0'])
x_ex22 = pd.DataFrame(bank[bank.columns[:3]])
```

```
    3.6216  8.6661  -2.8073  -0.44699  0
0  4.54590  8.1674  -2.4586  -1.46210  0
1  3.86600 -2.6383   1.9242   0.10645  0
2  3.45660  9.5228  -4.0112  -3.59440  0
3  0.32924 -4.4552   4.5718  -0.98880  0
4  4.36840  9.6718  -3.9606  -3.16250  0
```

In [68]:

```
# plot learning curves
x_train_ex44, x_test_ex44, y_train_ex44, y_test_ex44 = train_test_split(x_ex22, y_ex22, test_size = 0.34, random_state = 1)
x_train_ex44.shape, x_test_ex44.shape, y_train_ex44.shape, y_test_ex44.shape
```

Out[68]:

```
((904, 3), (467, 3), (904, 1), (467, 1))
```

In [69]:

```
clf = DecisionTreeClassifier(criterion = 'entropy', max_depth = 2)

random_state = 30
n_estimators = 300
step_factor = 10
axis_step = int(n_estimators/step_factor)

estimators = np.zeros(axis_step)
bagging_mse = np.zeros(axis_step)
bagging_mse_ts= np.zeros(axis_step)

est_lst = []
acc_tr = []
acc_ts = []


for i in range(0,axis_step):
    print("Bagging Estimator: %d of %d..."%(step_factor*(i+1), n_estimators))
    bag = BaggingRegressor(clf, n_estimators =step_factor*(i+1), n_jobs = 1, random_state = random_state)
    a= bag.fit(x_train_ex44, y_train_ex44)
    b= mean_squared_error(y_test_ex44, bag.predict(x_test_ex44))
    c = estimators[i] = step_factor*(i+1)
    d = bagging_mse[i] = b

    e = bag.fit(x_test_ex44, y_test_ex44)
    f = mean_squared_error(y_test_ex44, bag.predict(x_test_ex44))
    g = bagging_mse_ts[i] = f

    est_lst.append(c)
    acc_tr.append(d)
    #acc_ts.append(g)
```

```
Bagging Estimator: 10 of 300...
Bagging Estimator: 20 of 300...
Bagging Estimator: 30 of 300...
Bagging Estimator: 40 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 50 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 60 of 300...
Bagging Estimator: 70 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 80 of 300...
Bagging Estimator: 90 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 100 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 110 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 120 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

```
Bagging Estimator: 130 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 140 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 150 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 160 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 170 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 180 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 190 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 200 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 210 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 220 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 230 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 240 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 250 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 260 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 270 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 280 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 290 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)

Bagging Estimator: 300 of 300...

/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/sklearn/utils/validation.py:526: DataConversionWarning: A column-vector y was passed when a 1d array was
  y = column_or_1d(y, warn=True)
```

In [70]:

```
ex44 = pd.DataFrame({'Number of Bags':est_lst, 'Training Error':acc_tr})#,'Testing Error':acc_tr})
ex44
```

Out[70]:

	Number of Bags	Training Error
0	10	0.076874
1	20	0.077056
2	30	0.075927
3	40	0.076271
4	50	0.073804
5	60	0.072601

	Number of Bags	Training Error
6	70	0.073091
7	80	0.072941
8	90	0.073493
9	100	0.073098
10	110	0.072819
11	120	0.072556
12	130	0.072935
13	140	0.072858
14	150	0.072910
15	160	0.071911
16	170	0.071588
17	180	0.071332
18	190	0.070794
19	200	0.071080
20	210	0.071164
21	220	0.071121
22	230	0.071127
23	240	0.071023
24	250	0.071155
25	260	0.071332
26	270	0.071195
27	280	0.071049
28	290	0.071112
29	300	0.070938

In [71]:

```
plt.plot('Number of Bags','Training Error', data = ex44, marker = '', color = 'skyblue' )
#plt.plot('Number of Bags','Testing Error', data = pb4, marker = '', color = 'olive' )
plt.xlabel("Number of Bags")
plt.ylabel("Error")
plt.title("Bagging for BankNote")
plt.legend()
```

Out[71]:

```
<matplotlib.legend.Legend at 0x115ab7fd0>
```

In [ ]: