Machine Learning - CS 6375.004 - PA2

Abhinav Yalamaddi - YXA210040
Satwik Arvapalli - SXA220012

---

a. (Bagging, 20 points) Construct four models for each combination of maximum depth d = 3, 5 and bag size (k = 10, 20). Report the confusion matrix for these four settings.

```
----------------------BAGGING----------------------
Depth:  3 Bag Size:  10
Test Error = 4.23%
                Classifier Prediction
                        Positive        Negative
Actual | Positive         815              29
Value  | Negative         57               1130
Depth:  3 Bag Size:  20
Test Error = 4.23%
                Classifier Prediction
                        Positive        Negative
Actual | Positive         815              29
Value  | Negative         57               1130
Depth:  5 Bag Size:  10
Test Error = 0.20%
                Classifier Prediction
                        Positive        Negative
Actual | Positive         844              0
Value  | Negative         4                1183
Depth:  5 Bag Size:  20
Test Error = 0.20%
                Classifier Prediction
                        Positive        Negative
Actual | Positive         844              0
Value  | Negative         4                1183
```

Satwik Arvapalli - SXA220012
Abhinav Yalamaddi - YXA210040

b. (Boosting, 20 points) Construct four models for each combination of maximum depth d = 1, 2 and bag size (k = 20, 40). Report the confusion matrix for these four settings.

```
----------------------Boosting----------------------
Depth:  1 bag_size:  20
Test Error = 11.18%
                Classifier Prediction
                        Positive        Negative
Actual | Positive        793              51
Value  | Negative        176              1011
Depth:  1 bag_size:  40
Test Error = 11.18%
                Classifier Prediction
                        Positive        Negative
Actual | Positive        793              51
Value  | Negative        176              1011
Depth:  2 bag_size:  20
Test Error = 6.40%
                Classifier Prediction
                        Positive        Negative
Actual | Positive        823              21
Value  | Negative        109              1078
Depth:  2 bag_size:  40
Test Error = 6.40%
                Classifier Prediction
                        Positive        Negative
Actual | Positive        823              21
Value  | Negative        109              1078
```

Satwik Arvapalli - SXA220012
Abhinav Yalamaddi - YXA210040

c. (sci-kit-learn, 40 points) Use sci-kit-learn's bagging and AdaBoost learners and repeat the experiments as described in parts (a) and (b) above. Report the confusion matrices for these sets of settings. What can you say about the quality of your implementation's performance versus scikit's performance?

```
Bagging : max_depth = 3 bag_size =  10
Test Error =  4.382077794190053
Confusion matrix:
 [[1102   85]
 [   4  840]]
Bagging : max_depth = 3 bag_size =  20
Test Error =  4.382077794190053
Confusion matrix:
 [[1102   85]
 [   4  840]]
Bagging : max_depth = 5 bag_size =  10
Test Error =  1.1816838995568735
Confusion matrix:
 [[1187    0]
 [  24  820]]
Bagging : max_depth = 5 bag_size =  20
Test Error =  1.1816838995568735
Confusion matrix:
 [[1187    0]
 [  24  820]]
```

```
AdaBoost : max_depth = 1 bag_size =  20
Test Error =  0.1969473165928104
Confusion matrix:
 [[1185    2]
 [   2  842]]
AdaBoost : max_depth = 1 bag_size =  40
Test Error =  0.0
Confusion matrix:
 [[1187    0]
 [   0  844]]
AdaBoost : max_depth = 2 bag_size =  20
Test Error =  0.0
Confusion matrix:
 [[1187    0]
 [   0  844]]
AdaBoost : max_depth = 2 bag_size =  40
Test Error =  0.0
Confusion matrix:
 [[1187    0]
 [   0  844]]
```

Conclusion:

- Scikit's implementation of Bagging is nearly identical to ours.
- Scikit probably uses better decision tree criteria for boosting, giving it an advantage over our implementation.