

**ML ASSIGNMENT 4**  
**HARICHANDANA EPURI**  
**HXE170000**

**(A) Bagging:**

Confusion matrices for the depth  $d$  and bag size  $k$ :

$d=3$   $k=5$   
[[1130 57]  
[ 29 815]]

$d=5$   $k=5$   
[[1183 4]  
[ 0 844]]

$d=3$   $k=10$   
[[1130 57]  
[ 29 815]]

$d=5$   $k=10$   
[[1183 4]  
[ 0 844]]

**(B) Boosting:**

Confusion matrices for the depth  $d$  and ensemble size  $k$ :

$d=1$   $k=5$   
[[1011 176]  
[ 0 844]]

$d=1$   $k=10$   
[[1011 176]  
[ 0 844]]

$d=2$   $k=5$   
[[981 206]  
[ 0 844]]

$d=2$   $k=10$   
[[981 206]  
[ 0 844]]

**(C) Using Scikit:**

Confusion matrices for the depth  $d$  and bag size  $k$  (Bagging):

$d=3$   $k=5$   
[[1102 85]  
[ 4 840]]

```
d=5 k=5
[[1187  0]
 [ 24 820]]
```

```
d=3 k=10
[[1102 85]
 [  4 840]]
```

```
d=5 k=10
[[1187  0]
 [ 24 820]]
```

Confusion matrices for the depth d and ensemble size k (Boosting):

```
d=1 k=5
[[1061 126]
 [ 12 832]]
```

```
d=1 k=10
[[1157 30]
 [  5 839]]
```

```
d=2 k=5
[[1187  0]
 [  0 844]]
```

```
d=2 k=10
[[1187  0]
 [  0 844]]
```

**Observation:** AdaBoostClassifier in scikit learn supports multi class classification. Whereas the AdaBoostClassifier which I have implemented supports only binary classification and user can specify the base estimator. But in this implementation, the base estimator is a decision tree. BaggingClassifier in scikit learn supports multi-class classification and a user can specify if the sampling should be done with or without replacement. Whereas in this implementation, only binary classification is considered. User can specify the base estimator. But in this implementation, the base estimator is a decision tree and the samples are drawn with replacement. Both the classifiers implemented in scikit Learn are more generalized.