Block2assign1

Shwetha

11/23/2020

Assignment 1. Ensemble methods

Fitting random forest to training data and test data

```
rforest_1 = randomForest(trainlabels~., data = traindata, ntree = 1, nodesize = 25, keep.forest = TRUE)
rforest_10 = randomForest(trainlabels~., data = traindata, ntree = 10, nodesize = 25, keep.forest = TRU
rforest_100 = randomForest(trainlabels~., data = traindata, ntree = 100, nodesize = 25, keep.forest = T

y_1 = predict(rforest_1,testdata)
y_10 = predict(rforest_10,testdata)
y_100 = predict(rforest_100,testdata)

missclass=function(X,X1){
  n=length(X)
  return(1-sum(diag(table(X,X1)))/n)
}

y1_missclass = missclass(y_1,testlabels)
y10_missclass = missclass(y_10,testlabels)
y100_missclass = missclass(y_100,testlabels)

cat(" missclassification for test data of n = 1000 with 1 tree = ", missclass(y_1,testlabels))
```

```
##  missclassification for test data of n = 1000 with 1 tree =  0.188
```

```
cat("\n missclassification for test data of n = 1000 with 10 trees = ", missclass(y_10,testlabels))
```

```
##
##  missclassification for test data of n = 1000 with 10 trees =  0.142
```

```
cat("\n missclassification for test data of n = 1000 with 100 trees = ", missclass(y_100,testlabels))
```

```
##
##  missclassification for test data of n = 1000 with 100 trees =  0.112
```

Here we can observe that missclassification error reduces as the number of trees increases.

1

Repeating the procedure for 1000 training datasets of size 100.

```
missclass_large_sample = function(condition,ns = 25){
mce_1 = mce_10 = mce_100 = c()
set.seed(12345)
for(i in 1:1000){
```

```
  x1<-runif(100)
  x2<-runif(100)
  train<-cbind(x1,x2)
  y<-as.numeric(eval(parse(text = condition)))
  trainlabels<-as.factor(y)

  rf_1 = randomForest(trainlabels~., data = train, ntree = 1, nodesize = ns, keep.forest = TRUE)
  rf_10 = randomForest(trainlabels~., data = train, ntree = 10, nodesize = ns, keep.forest = TRUE)
  rf_100 = randomForest(trainlabels~., data = train, ntree = 100, nodesize = ns, keep.forest = TRUE)

  x1<-runif(1000)
  x2<-runif(1000)
  test<-cbind(x1,x2)
  y<-as.numeric(eval(parse(text = condition)))
  testlabels<-as.factor(y)

  y_1 = predict(rf_1,newdata = test)
  y_10 = predict(rf_10,test)
  y_100 = predict(rf_100,test)

  mce_1[i] = missclass(y_1,testlabels)
  mce_10[i] = missclass(y_10,testlabels)
  mce_100[i] = missclass(y_100,testlabels)

}
result = list("Mean_MCE_1tree"=mean(mce_1), "Variance_MCE_1tree"=var(mce_1),
              "Mean_MCE_10trees"=mean(mce_10), "Variance_MCE_10trees"=var(mce_10),
              "Mean_MCE_100trees"=mean(mce_100), "Variance_MCE_100trees"=var(mce_100))
return(result)
}
```

```
a = missclass_large_sample(condition = "x1 < x2")
a
```

```
## $Mean_MCE_1tree
## [1] 0.20492
##
## $Variance_MCE_1tree
## [1] 0.003056872
##
## $Mean_MCE_10trees
## [1] 0.136552
##
## $Variance_MCE_10trees
## [1] 0.0009772706
##
## $Mean_MCE_100trees
## [1] 0.111754
##
## $Variance_MCE_100trees
## [1] 0.0008971566
```

2

Repeating the same procedure as above with condition : $x1 < x2$

```
b = missclass_large_sample(condition = "x1 < 0.5")
b
```

```
## $Mean_MCE_1tree
## [1] 0.102442
##
## $Variance_MCE_1tree
## [1] 0.01960247
##
## $Mean_MCE_10trees
## [1] 0.015674
##
## $Variance_MCE_10trees
## [1] 0.000617251
##
## $Mean_MCE_100trees
## [1] 0.005717
##
## $Variance_MCE_100trees
## [1] 4.166257e-05
```

3

Repeating the same procedure as above with condition : (x1 $<$ 0.5 AND x2 $<$ 0.5) OR (x1 $>$ 0.5 AND x2 $>$ 0.5)

```
c = missclass_large_sample(condition = "(x1 < 0.5 & x2 < 0.5) | (x1 > 0.5 & x2 > 0.5)", ns = 12)
c
```

```
## $Mean_MCE_1tree
## [1] 0.24911
##
## $Variance_MCE_1tree
## [1] 0.01344997
##
## $Mean_MCE_10trees
## [1] 0.11682
##
## $Variance_MCE_10trees
## [1] 0.00273677
##
## $Mean_MCE_100trees
## [1] 0.071714
##
## $Variance_MCE_100trees
## [1] 0.00137132
```

4

a

The mean and variance of the error decreases as the number of trees grow. In general missclassification reduces and one gets better results with increasing the number of trees. However rate improvement in the result

decreases with increase in trees, in above examples we can see that difference in mean of missclassification error between 1 tree and 10 trees is more that between 10trees and 100 trees.

b

For the third condition, we can see that missclassification error was slightly higher than the other two conditions, but the results improve with the use of more trees. Random forest uses bagging , ie it picks a sample of observations rather than all of them. For a large observation, when the number of trees are too small, then some observations will be predicted only once or even not at all. This will result in a chance of high missclassification error. But using large number of trees solves this, resulting in better classification.

c

Higher the variance , higher will be sensitivity of the model to the training data , and this will reduce the accuracy of the prediction when we fit it to the test data which which may be different from training data.