Lab 1 Block 2

Mowniesh Asokan(mowas455)

11/23/2020

Assignment 1

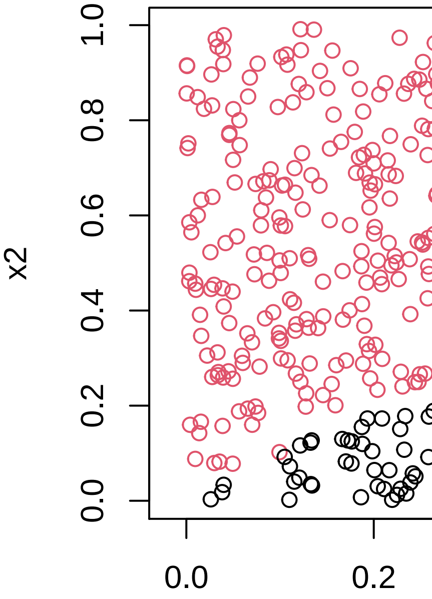
Ensemble Methods

Working of Random Forest

When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample. This oob (out-of-bag) data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of variable importance.

After each tree is built, all of the data are run down the tree, and proximities are computed for each pair of cases. If two cases occupy the same terminal node, their proximity is increased by one. At the end of the run, the proximities are normalized by dividing by the number of trees. Proximities are used in replacing missing data, locating outliers, and producing illuminating low-dimensional views of the data[1].

a.create 1000 training datasets of size 100, learn a random forest from each dataset, and compute the misclassification error in the same test dataset of size 1000. Report results for when the random forest has 1, 10 and 100 trees.

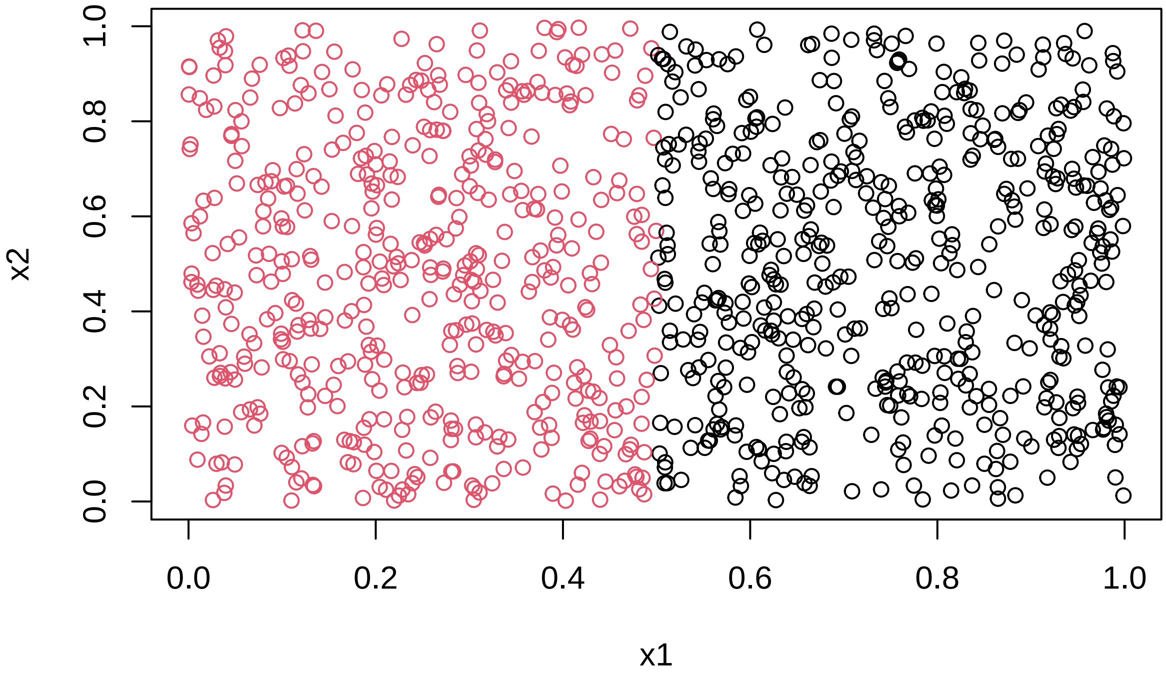


Random forest model can evaluated by using the data x1 and x2 ,where target y=x1<x2

```
## Mean of Miss classification using n_tree
##  n_tree=1- 0.186376
##  n_tree=10- 0.136079
##  n_tree=100- 0.112806
```

```
## Variance of Miss classification using n_tree
##  n_tree=1- 0.002220069
##  n_tree=10- 0.0009500808
##  n_tree=100- 0.0008645069
```

b.Repeat the exercise above but this time use the condition (x1<0.5) instead of (x1<x2) when producing the training and test datasets.

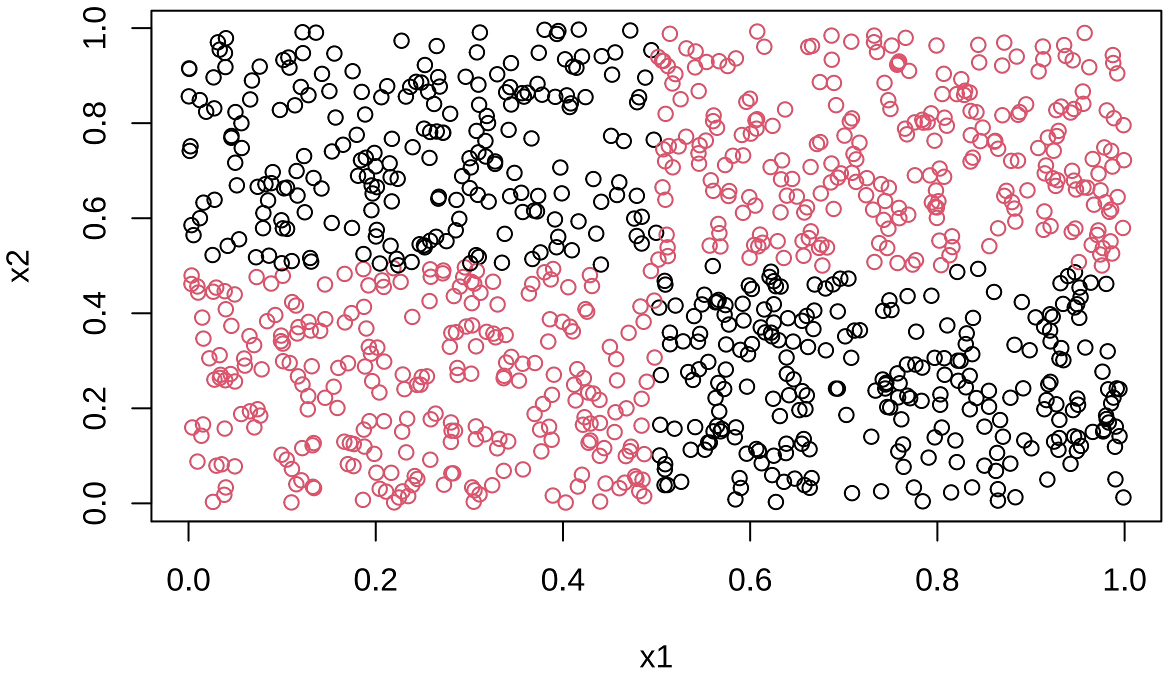


```
## telabels2
##   0   1
## 518 482

## Mean of Miss classification using n_tree
##  n_tree=1- 0.008663
##  n_tree=10- 0.017818
##  n_tree=100- 0.006733

## Variance of Miss classification using n_tree
##  n_tree=1- 8.866009e-05
##  n_tree=10- 0.0008040389
##  n_tree=100- 6.344315e-05
```

c.Repeat the exercise above but this time use the condition ((x1<0.5 & x2<0.5) | (x1>0.5 & x2>0.5)) instead of (x1<x2) when producing the training and test datasets. Unlike above, use nodesize = 12 for this exercise.



```
## telabels3
##   0   1
## 477 523

## Mean of Miss classification using n_tree
##  n_tree=1- 0.24032
##  n_tree=10- 0.120329
##  n_tree=100- 0.075465

## Variance of Miss classification using n_tree
##  n_tree=1- 0.01416965
##  n_tree=10- 0.002856301
##  n_tree=100- 0.001215871
```

d. Answer the following questions:

a. What happens with the mean and variance of the error rate when the number of trees in the random forest grows ? To increase the predictiveness of the model as much as possible at each partitioning so that the model keeps gaining information about the dataset by increasing the number of trees used in the model.From the above observations mean of miss classification error rate is decreased by the increasing the number of trees.At the same time variance of the error rate is keep decreasing.

b. The third dataset represents a slightly more complicated classification problem than the first one. Still, you should get better performance for it when using sufficient trees in the random forest. Explain why you get better performance. Random forest uses bagging (picking a sample of observations rather than all of them) and random subspace method (picking a sample of features rather than all of them, in other words - attribute bagging) to grow a tree. If the number of observations is large, but the number of trees is too small, then some observations will be predicted only once or even not at all. If the number of predictors is large but the number of trees is too small, then some features can (theoretically) be missed in all subspaces used. Both cases results in the decrease of random forest predictive power. But the last is a rather extreme case, since the selection of subspace is performed at each node[2].

But in this case , number of predictors (p=2) that is very small.To estimate an average of a real-valued random variable, from x1 and x2 we can take a sample.The expected variance will decrease as the square root of the sample size, and at a certain point the cost of collecting a larger sample will be higher than the benefit in accuracy obtained from such larger sample.

In this case we observe that in a multiple experiment on a single test set a forest of 12 trees performs better than a forest of 25 trees. This may be due to statistical variance.

Finally, ntree for the model is selected based on the number of predictors that we used in our data.

c. Why is it desirable to have low error variance ? The variances denote how well the model can generalize the data and how much our prediction accuracy is varying for training and test data. The model can Randomize the training data clearly for the all 1000 datasets of 100 rows.So the variance of miss classification error is low for test data.

References

1.https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#remarks

2.https://stats.stackexchange.com/questions/36165/does-the-optimal-number-of-trees-in-a-random-forest-depend-on-the-number-of-pred/36183

Appendix: All code for this report

```r
knitr::opts_chunk$set(echo = TRUE)
library(randomForest)
set.seed(12345)
list1 = list()
for (i in 1:1000) { # Indicate number of iterations with "i"
  x1 =  runif(100)
  x2 = runif(100)
  y<-as.factor(as.numeric(x1<x2))
  list1[[i]] = data.frame(x1,x2,y)
}


set.seed(1234)
x1<-runif(1000)
x2<-runif(1000)
tedata<-cbind(x1,x2)
y<-as.numeric(x1<x2)
```

```r
telabels<-as.factor(y)
plot(x1,x2,col=(y+1))
#table(telabels)
#m- Denotes the miss classification error rate (_number indicates the number of trees)
#n- Denotes the accuracy

m_1<-c()
m_10<-c()
m_100<-c()
n_1<-c()
n_10<-c()
n_100<-c()
for (i in list1){
  set.seed(12345)

  n<-as.data.frame(i)

  rf1=randomForest(y~.,data=n,ntree=1,nodesize=25,keep.forest = TRUE)

  rf2=randomForest(y~.,data=n,ntree=10,nodesize=25,keep.forest = TRUE)

  rf3=randomForest(y~.,data=n,ntree=100,nodesize=25,keep.forest = TRUE)

  pred1<-predict(rf1,newdata = tedata)

  pred2<-predict(rf2,newdata = tedata)

  pred3<-predict(rf3,newdata = tedata)

  accuracy1<-mean(telabels == pred1)

  accuracy2<-mean(telabels == pred2)

  accuracy3<-mean(telabels == pred3)

  cm1 <- table(actual = telabels, fitted = pred1)

  mmce1 <- 1 - (sum(diag(cm1))/sum(cm1))

  cm2 <- table(actual = telabels, fitted = pred2)

  mmce2 <- 1 - (sum(diag(cm2))/sum(cm2))

  cm3 <- table(actual = telabels, fitted = pred3)

  mmce3 <- 1 - (sum(diag(cm3))/sum(cm3))


  n_1<-c(n_1,accuracy1)

  n_10<-c(n_10,accuracy2)

  n_100<-c(n_100,accuracy3)
  m_1<-c(m_1,mmce1)

  m_10<-c(m_10,mmce2)

  m_100<-c(m_100,mmce3)

}

trd<-data.frame(n_1,n_10,n_100,m_1,m_10,m_100)

cat("Mean of Miss classification using n_tree","\n","n_tree=1-",mean(trd$m_1),"\n","n_tree=10-",mean(tr

cat("Variance of Miss classification using n_tree","\n","n_tree=1-",var(trd$m_1),"\n","n_tree=10-",var(

#Condition- 2 (x1<0.5)

set.seed(12345)
list2 = list()
for (i in 1:1000) { # Indicate number of iterations with "i"
  x1 =  runif(100)
  x2 = runif(100)
  y<-as.factor(as.numeric(x1<0.5))
  list2[[i]] = data.frame(x1,x2,y)
}


set.seed(1234)
x1<-runif(1000)
x2<-runif(1000)
tedata2<-cbind(x1,x2)
y<-as.numeric((x1<0.5))
telabels2<-as.factor(y)
plot(x1,x2,col=(y+1))
table(telabels2)

#list2

m2_1<-c()
m2_10<-c()
m2_100<-c()
n2_1<-c()
n2_10<-c()
n2_100<-c()
for (i in list2){
  set.seed(12345)

  n<-as.data.frame(i)

  rf12=randomForest(y~.,data=n,ntree=1,nodesize=25,keep.forest = TRUE)

  rf22=randomForest(y~.,data=n,ntree=10,nodesize=25,keep.forest = TRUE)

  rf32=randomForest(y~.,data=n,ntree=100,nodesize=25,keep.forest = TRUE)
  pred1<-predict(rf12,newdata = tedata2)

  pred2<-predict(rf22,newdata = tedata2)

  pred3<-predict(rf32,newdata = tedata2)

  accuracy1<-mean(telabels2 == pred1)

  accuracy2<-mean(telabels2 == pred2)

  accuracy3<-mean(telabels2 == pred3)

  cm1 <- table(actual = telabels2, fitted = pred1)

  mmce1 <- 1 - (sum(diag(cm1))/sum(cm1))

  cm2 <- table(actual = telabels2, fitted = pred2)

  mmce2 <- 1 - (sum(diag(cm2))/sum(cm2))

  cm3 <- table(actual = telabels2, fitted = pred3)

  mmce3 <- 1 - (sum(diag(cm3))/sum(cm3))


  n2_1<-c(n2_1,accuracy1)

  n2_10<-c(n2_10,accuracy2)

  n2_100<-c(n2_100,accuracy3)

  m2_1<-c(m2_1,mmce1)

  m2_10<-c(m2_10,mmce2)

  m2_100<-c(m2_100,mmce3)

}

trd2<-data.frame(n2_1,n2_10,n2_100,m2_1,m2_10,m2_100)

cat("Mean of Miss classification using n_tree","\n","n_tree=1-",mean(trd2$m2_1),"\n","n_tree=10-",mean(

cat("Variance of Miss classification using n_tree","\n","n_tree=1-",var(trd2$m2_1),"\n","n_tree=10-",va


#Condition- 3 ((x1<0.5 & x2<0.5) | (x1>0.5 & x2>0.5))

set.seed(12345)
list3 = list()
for (i in 1:1000) { # Indicate number of iterations with "i"
  x1 =  runif(100)
  x2 = runif(100)
  y<-as.factor(as.numeric((x1<0.5 & x2<0.5) | (x1>0.5 & x2>0.5)))
  list3[[i]] = data.frame(x1,x2,y)
}


set.seed(1234)
x1<-runif(1000)
x2<-runif(1000)
tedata3<-cbind(x1,x2)
y<-as.numeric((x1<0.5 & x2<0.5) | (x1>0.5 & x2>0.5))
telabels3<-as.factor(y)
plot(x1,x2,col=(y+1))
table(telabels3)

#list3

m3_1<-c()
m3_10<-c()
m3_100<-c()
n3_1<-c()
n3_10<-c()
n3_100<-c()
for (i in list3){
  set.seed(12345)

  n<-as.data.frame(i)

  rf13=randomForest(y~.,data=n,ntree=1,nodesize=12,keep.forest = TRUE)

  rf23=randomForest(y~.,data=n,ntree=10,nodesize=12,keep.forest = TRUE)

  rf33=randomForest(y~.,data=n,ntree=100,nodesize=12,keep.forest = TRUE)

  pred1<-predict(rf13,newdata = tedata3)

  pred2<-predict(rf23,newdata = tedata3)

  pred3<-predict(rf33,newdata = tedata3)

  accuracy1<-mean(telabels3 == pred1)

  accuracy2<-mean(telabels3 == pred2)

  accuracy3<-mean(telabels3 == pred3)

  cm1 <- table(actual = telabels3, fitted = pred1)

  mmce1 <- 1 - (sum(diag(cm1))/sum(cm1))

  cm2 <- table(actual = telabels3, fitted = pred2)

  mmce2 <- 1 - (sum(diag(cm2))/sum(cm2))
  cm3 <- table(actual = telabels3, fitted = pred3)

  mmce3 <- 1 - (sum(diag(cm3))/sum(cm3))


  n3_1<-c(n3_1,accuracy1)

  n3_10<-c(n3_10,accuracy2)

  n3_100<-c(n3_100,accuracy3)

  m3_1<-c(m3_1,mmce1)

  m3_10<-c(m3_10,mmce2)

  m3_100<-c(m3_100,mmce3)

}

trd3<-data.frame(n3_1,n3_10,n3_100,m3_1,m3_10,m3_100)

cat("Mean of Miss classification using n_tree","\n","n_tree=1-",mean(trd3$m3_1),"\n","n_tree=10-",mean(t

cat("Variance of Miss classification using n_tree","\n","n_tree=1-",var(trd3$m3_1),"\n","n_tree=10-",va
#mean(trd3$n3_100)
```