Homework 2 Solution

Harsh Yadav (hy1217), Nitish Dabas (nd1292), Kush Shah(ks4437)

11/17/2017

Introduction:

This dataset has information about the contributions made by the ALumni of an organization during different fiscal years fron 2000 to 2004 along with data such as Degree, Next Degree, Gender, Marital Status, etc. We have performed the EDA using various graphical representations which have been explained in detail.

Further Regression tree algorithm has been performed to find the value of contribution made in the year FY04Giving. So, for making this prediction we have considered all the columns in the dataset like Gender, Marital status, FY00Giving, FY01Giving, etc apart from the Major and Next Degree columns as they have more than 32 different values/factors on which regression tree analysis cannot be performed and also we cannot classify them into broad categories as it will not make any sense. So, for this we are considering the column FY04Giving as aur target variable and rest if the columns apart from Major and Next Degree as our predictor variables. Further, we have performed 10 fold cross validation for obtaining the best results and we observe that the best result occurs at the 8th fold of cross validation which has the least deviance. So, we consider this as the best validation of our data and prune our data on it.

EDA:

```
df.contributions<- read.csv("C:/Users/Harsh Yadav/Desktop/contribution.csv")
str(df.contributions)
```

```
## 'data.frame':    1230 obs. of  11 variables:
##  $ Gender         : Factor w/ 2 levels "F","M": 2 2 1 2 2 1 1 1 2 2 ...
##  $ Class.Year     : int  1957 1957 1957 1957 1957 1957 1957 1957 1957 1957 ...
##  $ Marital.Status : Factor w/ 4 levels "D","M","S","W": 2 2 2 2 2 2 3 2 2 2 ...
##  $ Major          : Factor w/ 46 levels "American Studies",..: 26 34 30 26 4 28 26 30 42 16 ...
##  $ Next.Degree    : Factor w/ 50 levels "AA","BA","BAE",..: 17 37 42 42 26 42 19 42 42 42 ...
##  $ FY04Giving     : num  2500 5000 5000 0 1000 0 0 100 100 0 ...
##  $ FY03Giving     : num  2500 5000 5000 5100 1000 0 0 100 100 0 ...
##  $ FY02Giving     : num  1400 5000 5000 200 1000 0 0 100 100 0 ...
##  $ FY01Giving     : num  12060 5000 5000 200 1005 ...
##  $ FY00Giving     : num  12000 10000 10000 0 1000 0 0 100 100 0 ...
##  $ AttendenceEvent: int  1 1 1 1 1 0 0 0 0 1 ...
```

```
#summary(df.contributions)
```

The dataframe contains 1230 observations(rows) and 11 variables(columns) where FY0*Giving denotes amount donated in that particular fiscal year.

```
sumdata<-c(sum(df.contributions$FY00Giving),sum(df.contributions$FY01Giving),
           sum(df.contributions$FY02Giving),sum(df.contributions$FY03Giving),
           sum(df.contributions$FY04Giving))
print(sprintf("Amount-%f
              Year-FY0%dGiving",max(sumdata),match(max(sumdata),sumdata)))
```

```
## [1] "Amount-340130.590000\n              Year-FY02Giving"
```

We can see that the maximum amount was donated in the year FY02Giving, this can be found by first making a vector and then using max and match functions to get the highest value with index.

```
PeopleGen<- subset(df.contributions, df.contributions$AttendenceEvent==0 &
                     df.contributions$FY00Giving>0 &
                     df.contributions$FY01Giving>0 &
                     df.contributions$FY02Giving>0 &
                     df.contributions$FY03Giving>0 &
                     df.contributions$FY04Giving>0)
dim(PeopleGen)
```

```
## [1] 61 11
```

There are 61 people who never attended the event but still made contibutions

```
PeopleNG<- subset(df.contributions, df.contributions$AttendenceEvent==1 &
                    df.contributions$FY00Giving==0 &
                    df.contributions$FY01Giving==0 &
                    df.contributions$FY02Giving==0 &
                    df.contributions$FY03Giving==0 &
                    df.contributions$FY04Giving==0)
dim(PeopleNG)
```

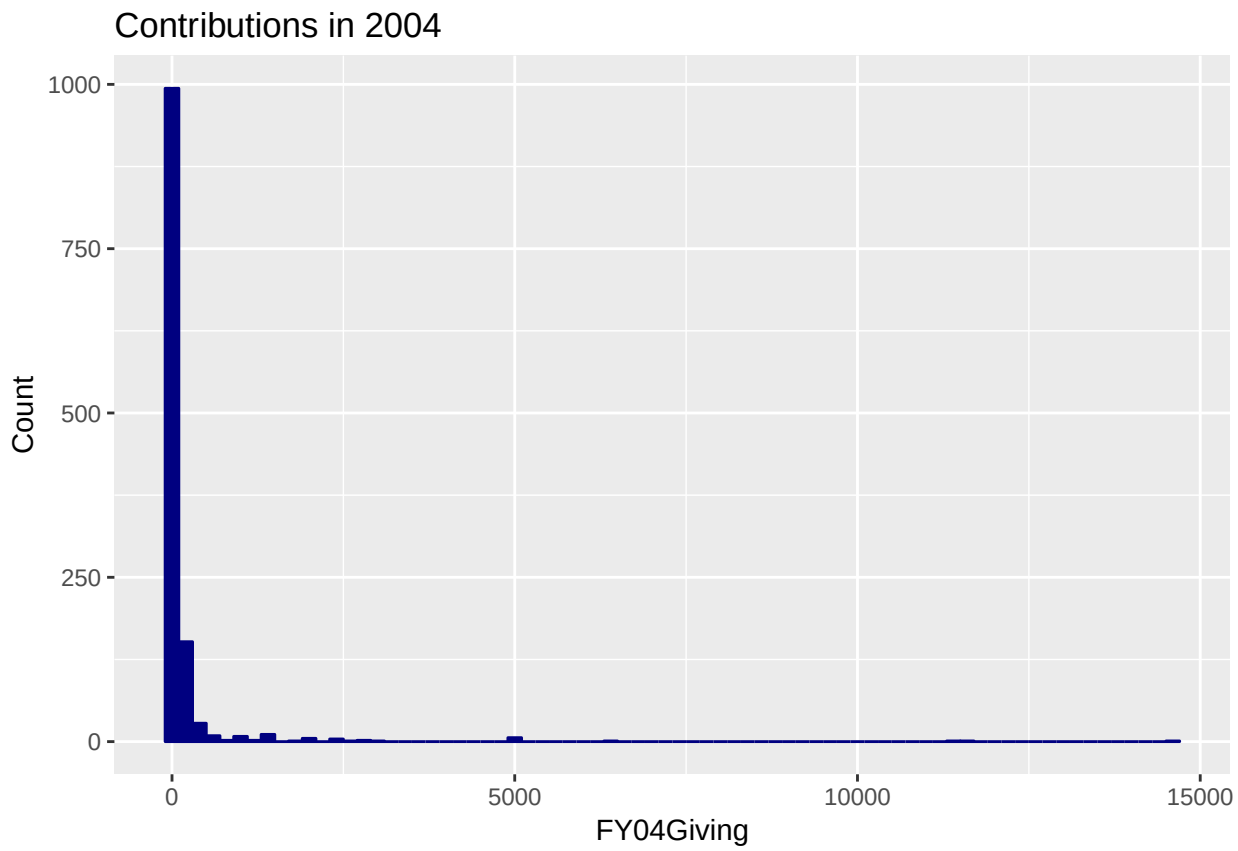
```
## [1] 184  11
```

There are 184 people who attended the event but never contributed

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.2
```

```
ggplot(df.contributions, aes(x=FY04Giving)) +
    geom_histogram(binwidth=200, colour="navy", fill="navy")  +
    labs(x="FY04Giving",y="Count") +
    ggtitle("Contributions in 2004")
```



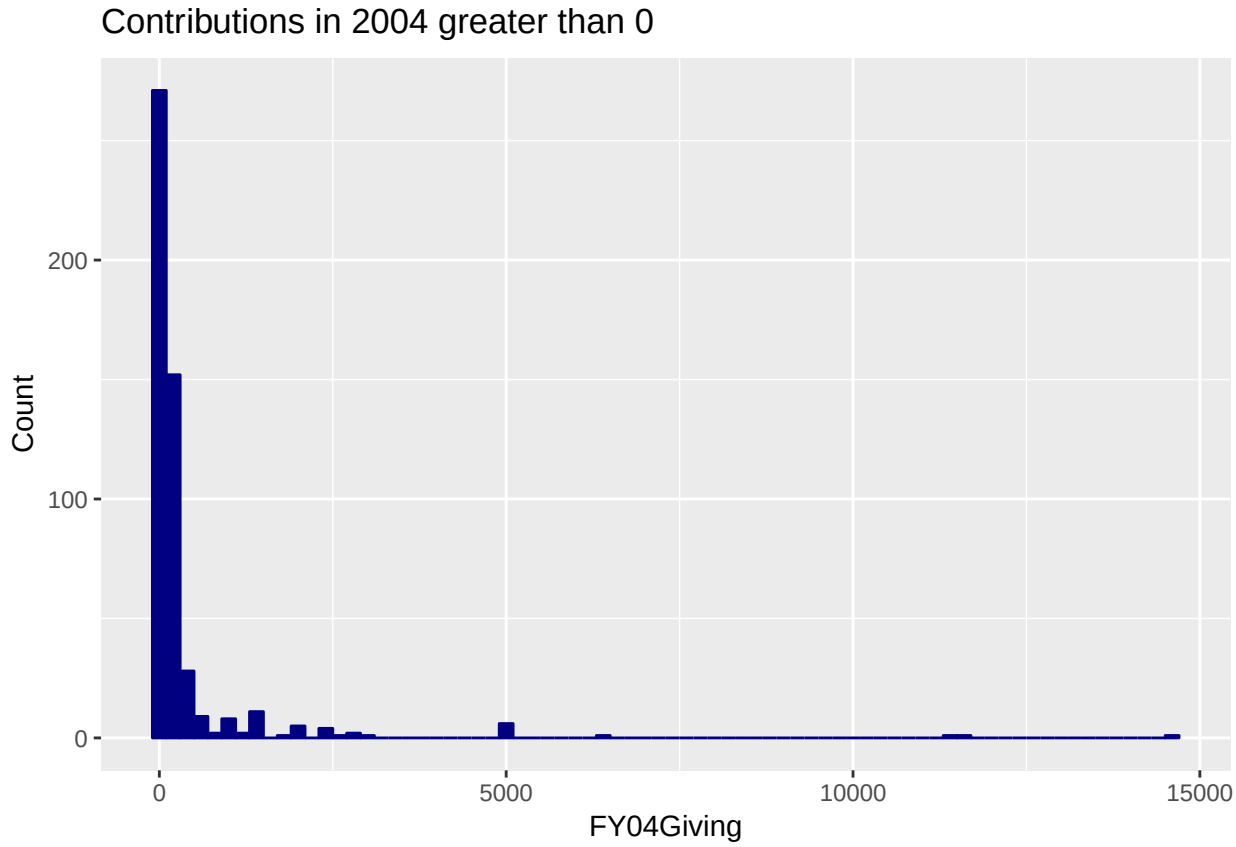
The above graph shows the histogram distribution of all the contributions in FY04Giving.

```
zero.data<-subset(df.contributions, df.contributions$FY04Giving==0)
dim(zero.data)
```

```
## [1] 723  11
```

We can see that there are 723 people whose contribution was zero during the year FY04

```
data<-subset(df.contributions, df.contributions$FY04Giving>0)
ggplot(data, aes(x=FY04Giving)) +
    geom_histogram(binwidth=200, colour="navy", fill="navy")  +
    ggtitle("Contributions in 2004 greater than 0") +
    labs(x="FY04Giving",y="Count")
```

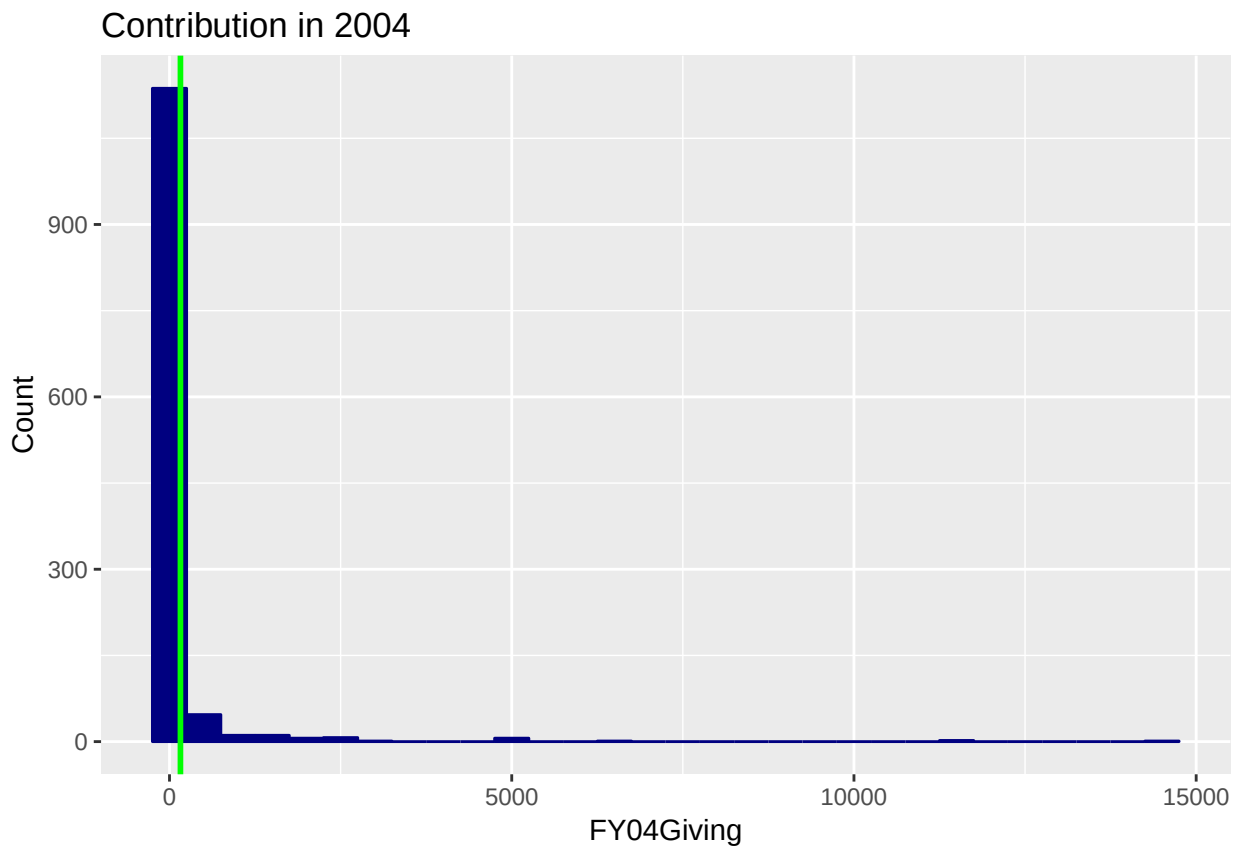


The graph above shows the histogram representation of people who actually contributed in FY04 (i.e all the 0 amount contributions have been removed) therefore we can see that count on y axis has reduced significantly.

```
FY04Giving.mean1<-mean(df.contributions$FY04Giving)
FY04Giving.mean1
```

```
## [1] 159.3999
```

```
ggplot(df.contributions, aes(x=FY04Giving)) +
    geom_histogram(binwidth=500, colour="navy", fill="navy") +
    ggtitle("Contribution in 2004") +
    labs(x="FY04Giving",y="Count") +
    geom_vline(aes(xintercept=mean(FY04Giving, na.rm=T)),
               color="green", linetype="solid", size=1)
```

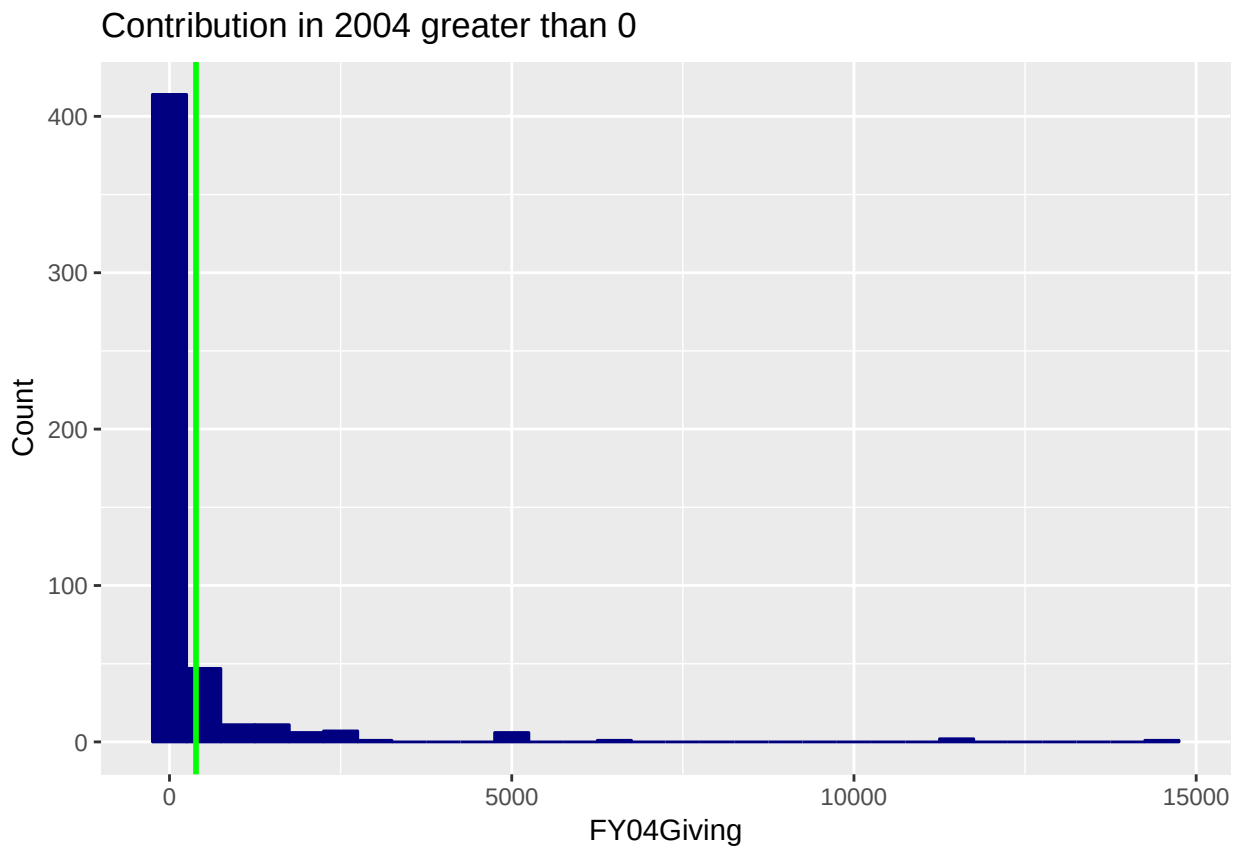


We can see the mean of contributions in FY04 is 159.399 which has been marked by green line in the graph. This graph

```
FY04Giving.mean2<-mean(subset(df.contributions$FY04Giving,df.contributions$FY04Giving>0))
FY04Giving.mean2
```

```
## [1] 386.7097
```

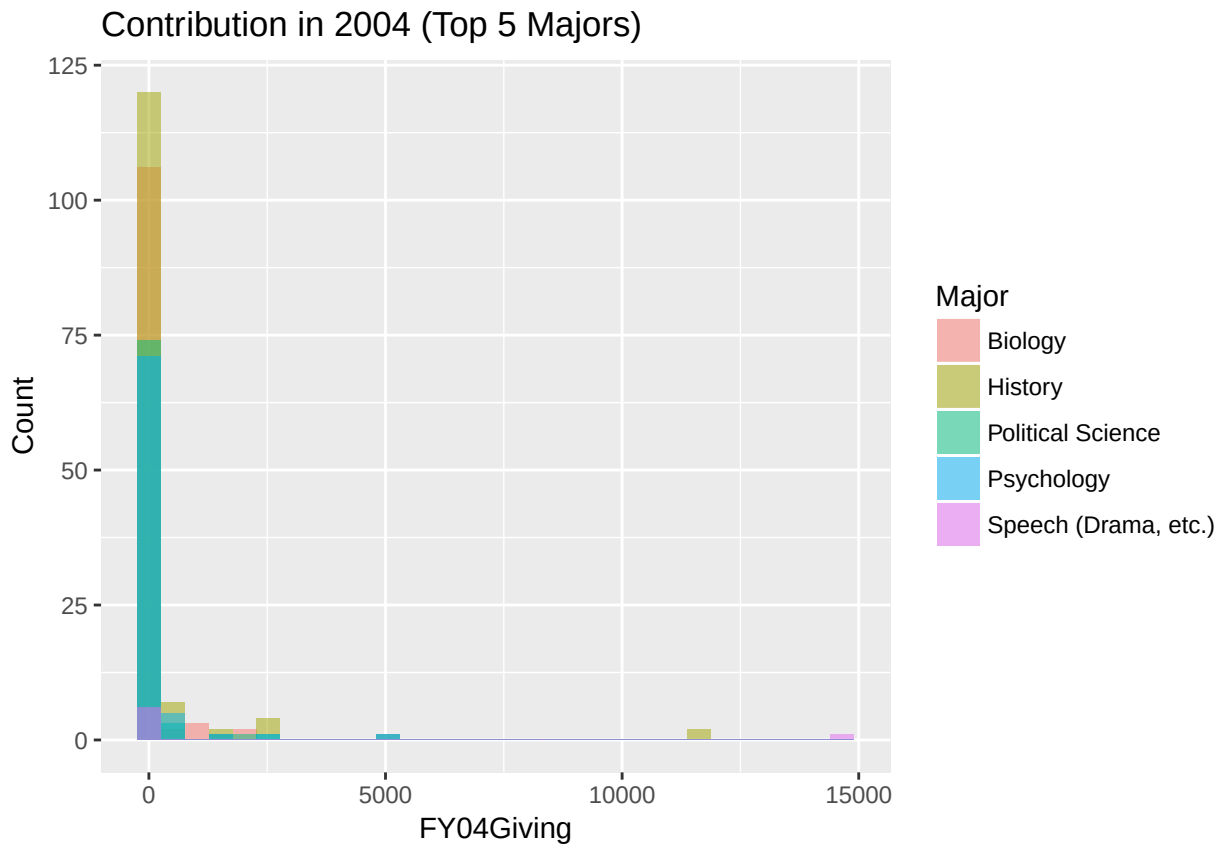
```
data<-subset(df.contributions, df.contributions$FY04Giving>0)
ggplot(data, aes(x=FY04Giving)) +
    geom_histogram(binwidth=500, colour="navy", fill="navy") +
    ggtitle("Contribution in 2004 greater than 0") +
    labs(x="FY04Giving",y="Count") +
    geom_vline(aes(xintercept=mean(subset(df.contributions$FY04Giving,
                                          df.contributions$FY04Giving>0))),
               color="green", linetype="solid", size=1)
```



Now if we exclude people whose contribution was 0 during FY04 then the mean value will obviouldy increase to 386.7097 i.e the green line has moved to the right on x-axis of our graph.

```
majorlist<-c(unique(df.contributions$Major))
i=0;
sum_total2 <- c()
for (majorname in majorlist)
{
  i=i+1;
  sum_total2[i] <- (sum(subset(df.contributions$FY04Giving,   df.contributions$Major==majorname)))
}
majordata <- data.frame(majorlist, sum_total2)

newdata <- majordata[order(-sum_total2),]
df.new<-subset(df.contributions,df.contributions$Major=="Biology"
               |df.contributions$Major=="History"
               |df.contributions$Major=="Political Science"
               |df.contributions$Major=="Speech (Drama, etc.)"
               |df.contributions$Major=="Psychology")
ggplot(df.new, aes(x=FY04Giving, fill=Major)) +
    geom_histogram(alpha=0.5, position="identity") +
    labs(x="FY04Giving",y="Count") +
    ggtitle("Contribution in 2004 (Top 5 Majors)")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The above graph gives histogram representation for top five contributing majors. To get top five contributing major I have used the code otherwise there were 46 different majors in our dataset.

```
library(magrittr)
library(dplyr)
```

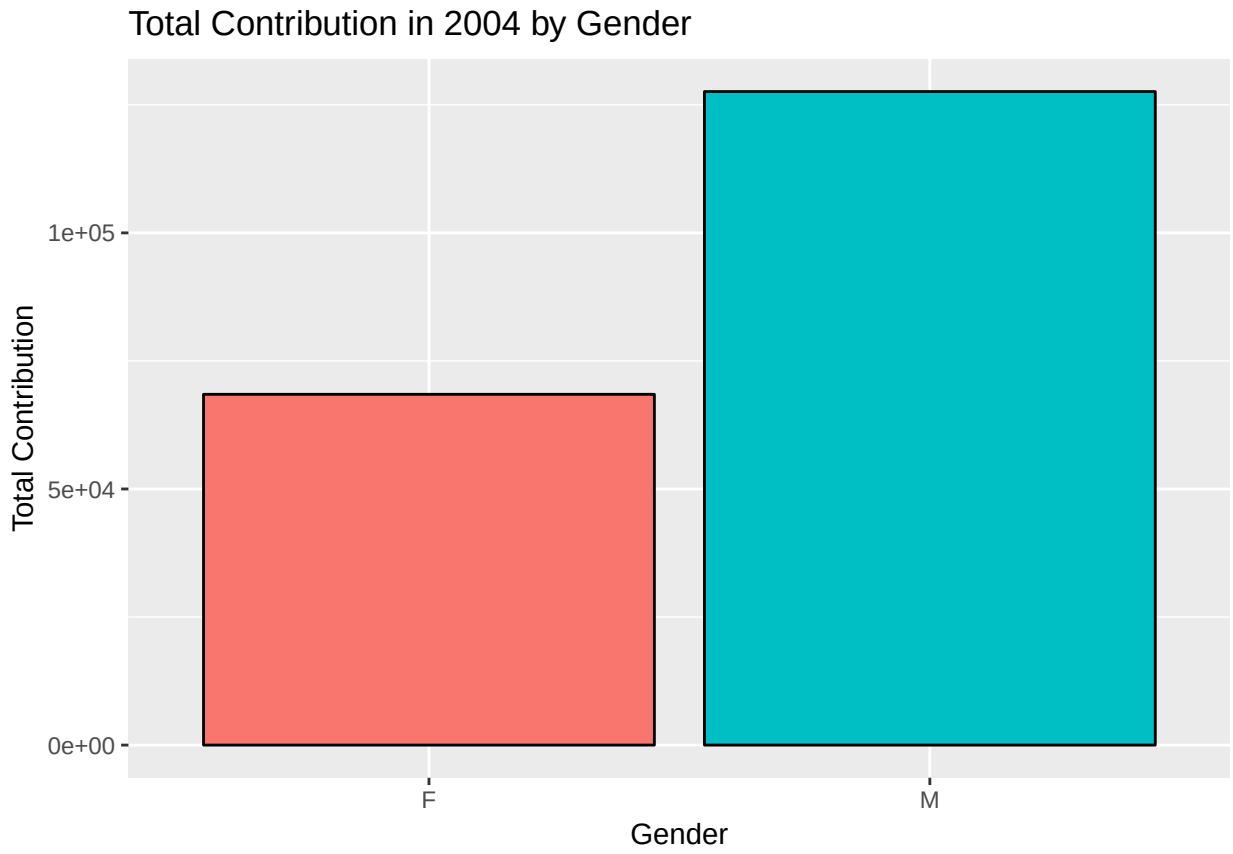
```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

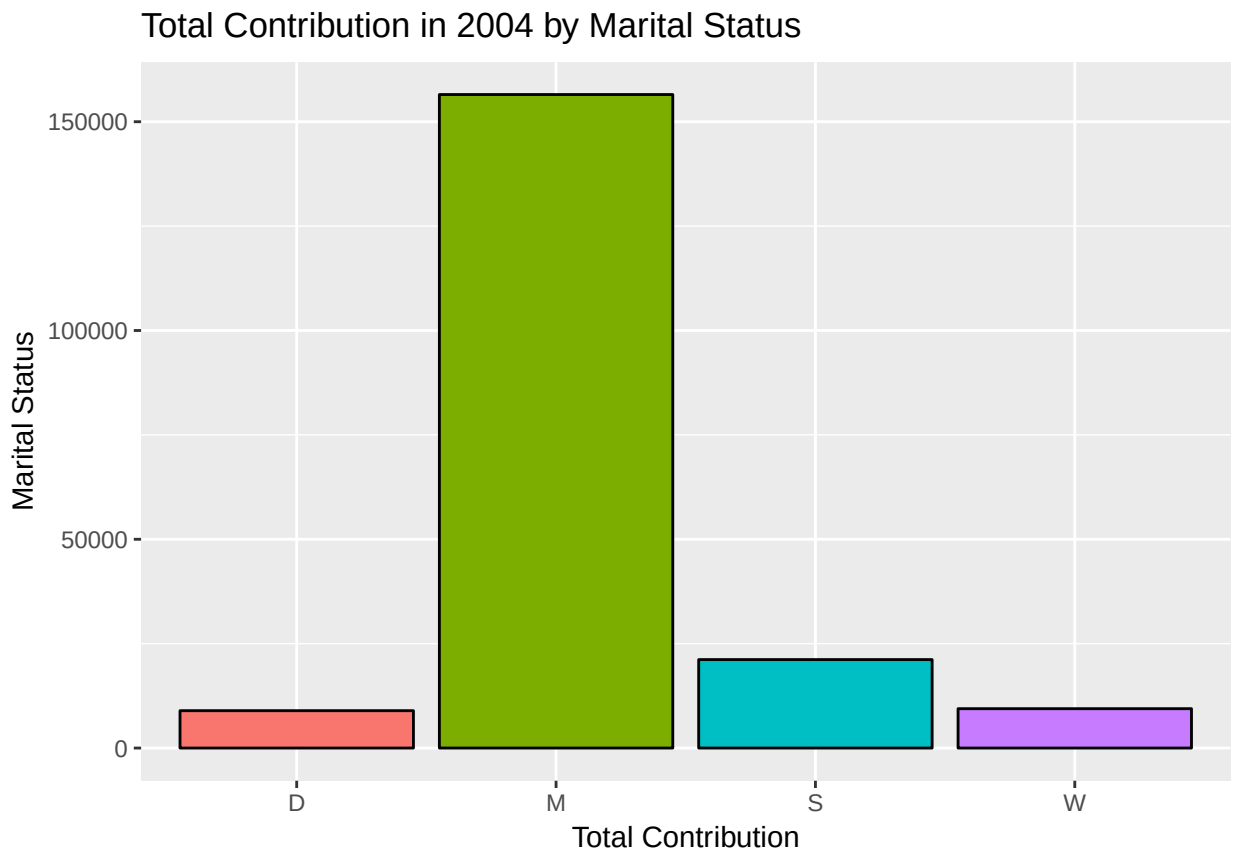
```
contribution.by.gender <- df.contributions %>%
  group_by(Gender) %>%
  summarise(
    n.obs = n(),
    tot.contributionFY04 = sum(FY04Giving)
  )
```

```
ggplot(data=contribution.by.gender, aes(x=Gender, y=tot.contributionFY04, fill=Gender)) +
  geom_bar(colour="black", stat="identity") +
  guides(fill=FALSE) +
  labs(x="Gender",y="Total Contribution") +
  ggtitle("Total Contribution in 2004 by Gender")
```



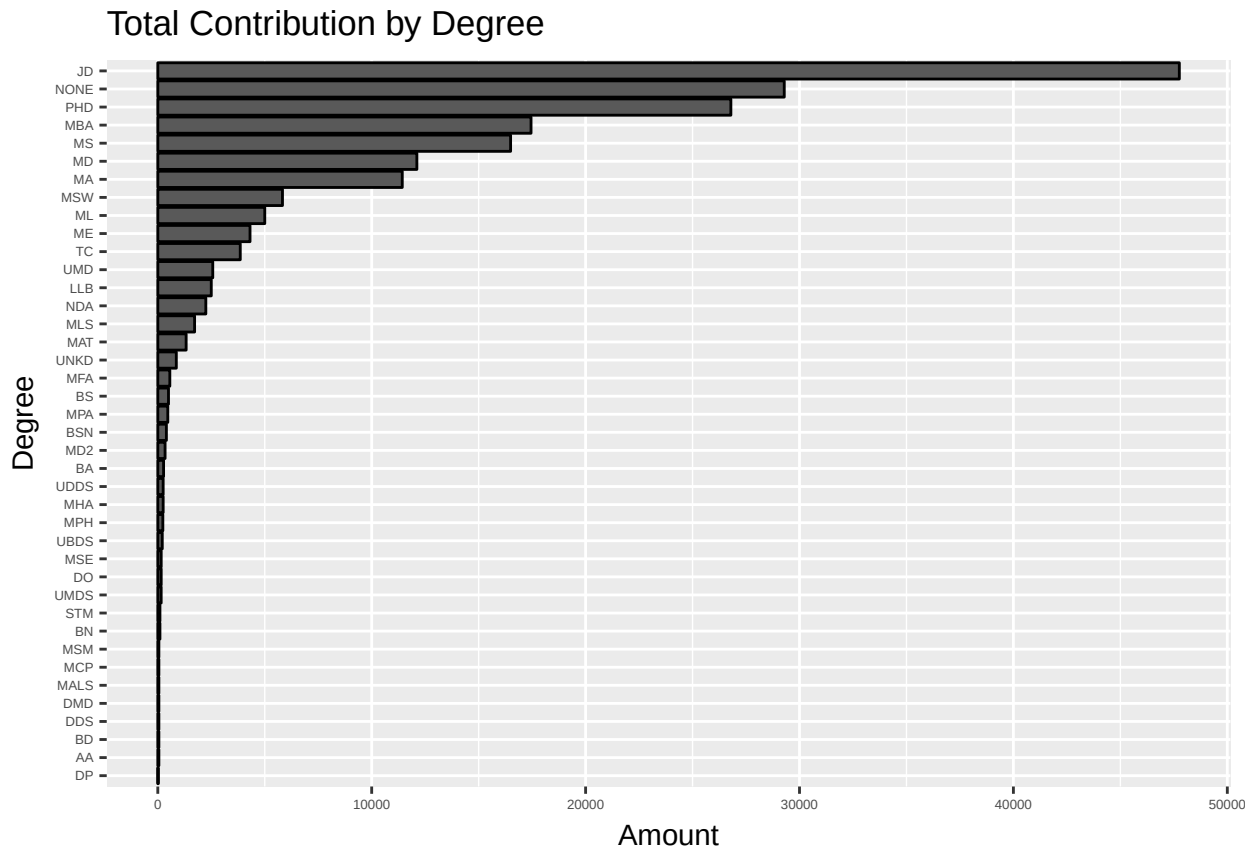
The above graph shows the contributions based on gender. We can see that the males have made more contribution.

```
contribution.by.status <- df.contributions %>%
  group_by(Marital.Status) %>%
  summarise(
    n.obs = n(),
    tot.contributionFY04 = sum(FY04Giving)
  )
ggplot(data=contribution.by.status, aes(x=Marital.Status, y=tot.contributionFY04, fill=Marital.Status))
  geom_bar(colour="black", stat="identity") +
  labs(x="Total Contribution",y="Marital Status") +
  guides(fill=FALSE) +
  ggtitle("Total Contribution in 2004 by Marital Status")
```



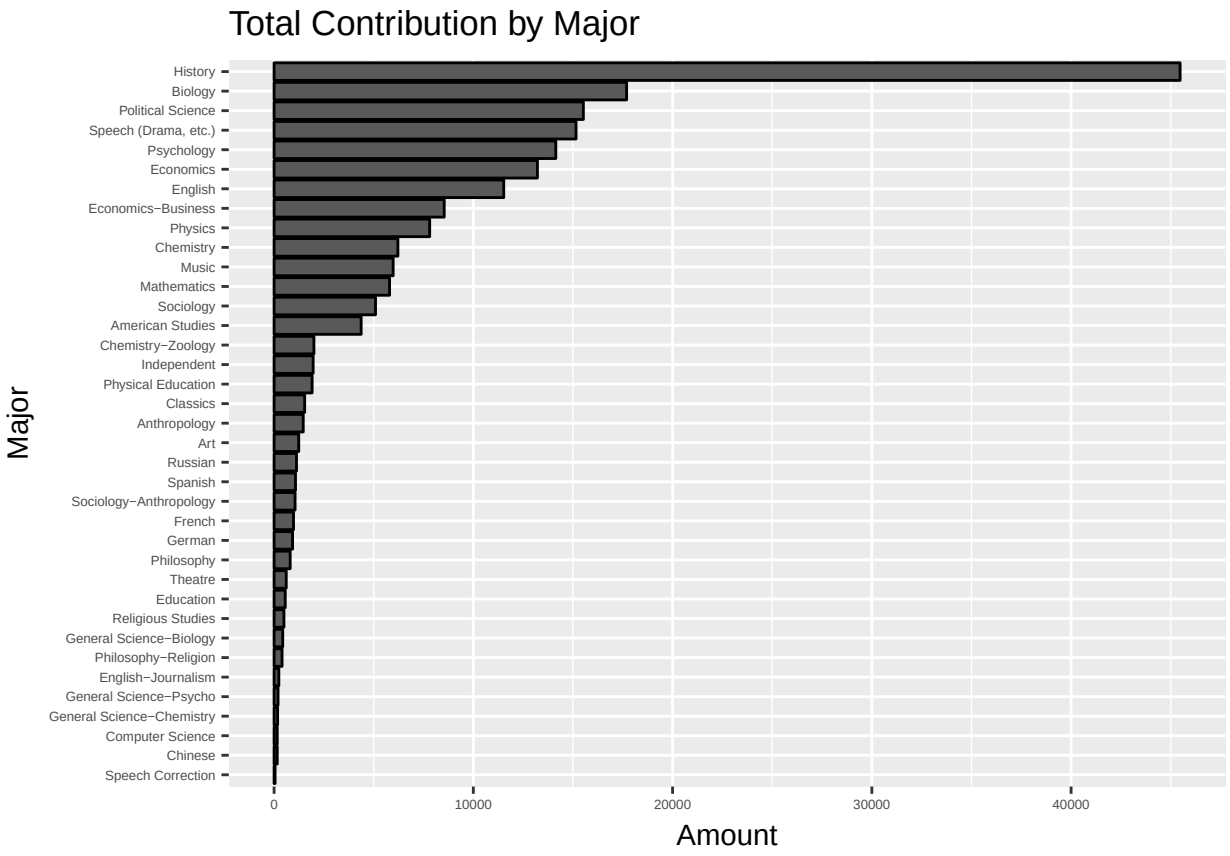
The above graph shows the total contributions based on marital status, we can see that Married people have contributed the most based on the graph representation.

```
contri.by.degree <- df.contributions %>%
  group_by(Next.Degree) %>%
  summarise(
    tot.FY04 = sum(FY04Giving)
  )
#typeof(contri.by.degree)

df1 <- subset(contri.by.degree,contri.by.degree$tot.FY04>0)
df2 <- df1[order(df1$tot.FY04),]
df2$Next.Degree <- factor(df2$Next.Degree, levels = df2$Next.Degree[order(df2$tot.FY04)])

#df2
p <- ggplot(data=df2, aes(x=Next.Degree, y=tot.FY04)) +
  ggtitle("Total Contribution by Degree") +
  geom_bar(colour="black", stat="identity") +
  labs(x="Degree",y="Amount") +
  guides(fill=FALSE)
p + coord_flip() + theme(axis.text=element_text(size=5))
```



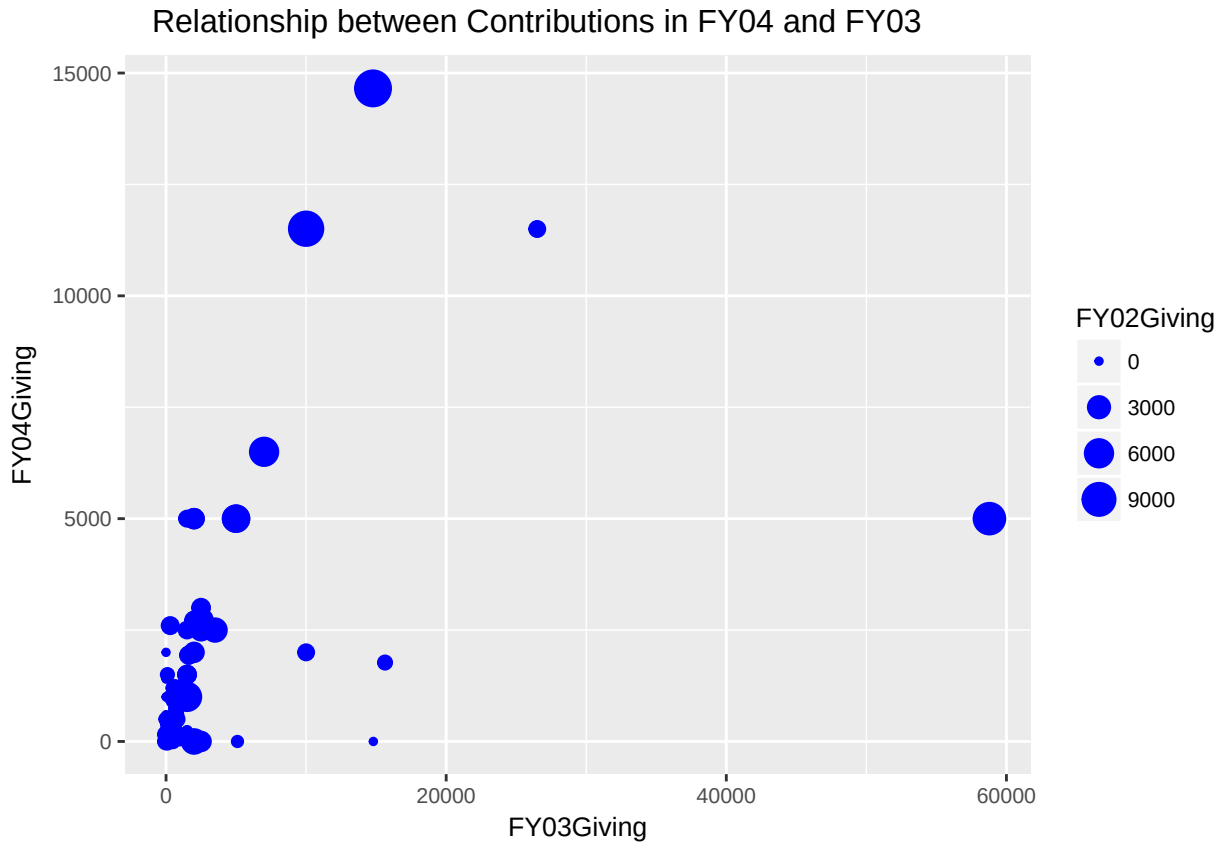
The above representation shows the total contributions made by each degree in a descending order from top to bottom. I have excluded the degrees which made 0 contribution for clear graphical representation.

```
contri.by.major <- df.contributions %>%
  group_by(Major) %>%
  summarise(
    tot.FY04 = sum(FY04Giving)
  )
#typeof(contri.by.degree)

df1 <- subset(contri.by.major,contri.by.major$tot.FY04>0)
df2 <- df1[order(df1$tot.FY04),]
df2$Major <- factor(df2$Major, levels = df2$Major[order(df2$tot.FY04)])

#df2
p <- ggplot(data=df2, aes(x=Major, y=tot.FY04)) +
  ggtitle("Total Contribution by Major") +
  geom_bar(colour="black", stat="identity") +
  labs(x="Major",y="Amount") +
  guides(fill=FALSE)
p + coord_flip() + theme(axis.text=element_text(size=5))
```



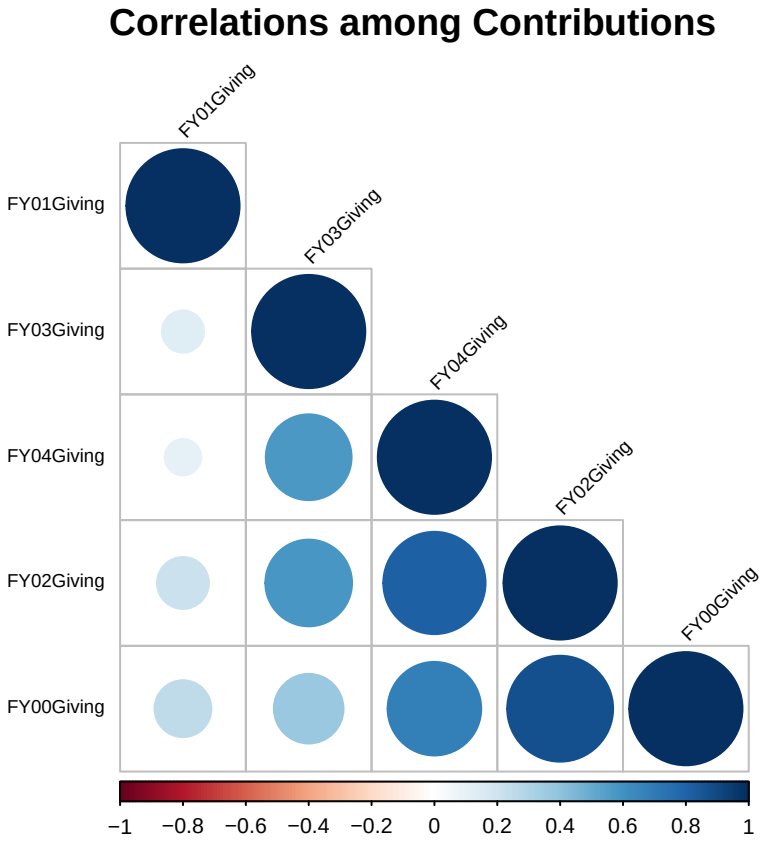
The above representation shows the total contributions made by each major in a descending order from top to bottom. I have excluded the majors which made 0 contribution for clear graphical representation.

```
ggplot(df.contributions, aes(x=FY03Giving, y=FY04Giving)) +
  geom_point(aes(size=FY02Giving),colour='Blue') +
  ggtitle("Relationship between Contributions in FY04 and FY03") +
  theme(text=element_text(size=10), plot.title = element_text(hjust = 0.2))
```



This plot is shows the contributions made in years FY02,FY03,FY04. The contributions in year 2003, 2004 can be measured by the x-y axis and 2002 contributions can be estimated based on the circle size shown in the legend.

```
library(corrplot)
corrplot(cor(df.contributions[,6:10]), tl.cex=0.6, cl.cex=0.7,
         type = "lower", order = "hclust", tl.col = "black", tl.srt = 45,
         title="Correlations among Contributions ", mar=c(0,0,2,0))
```



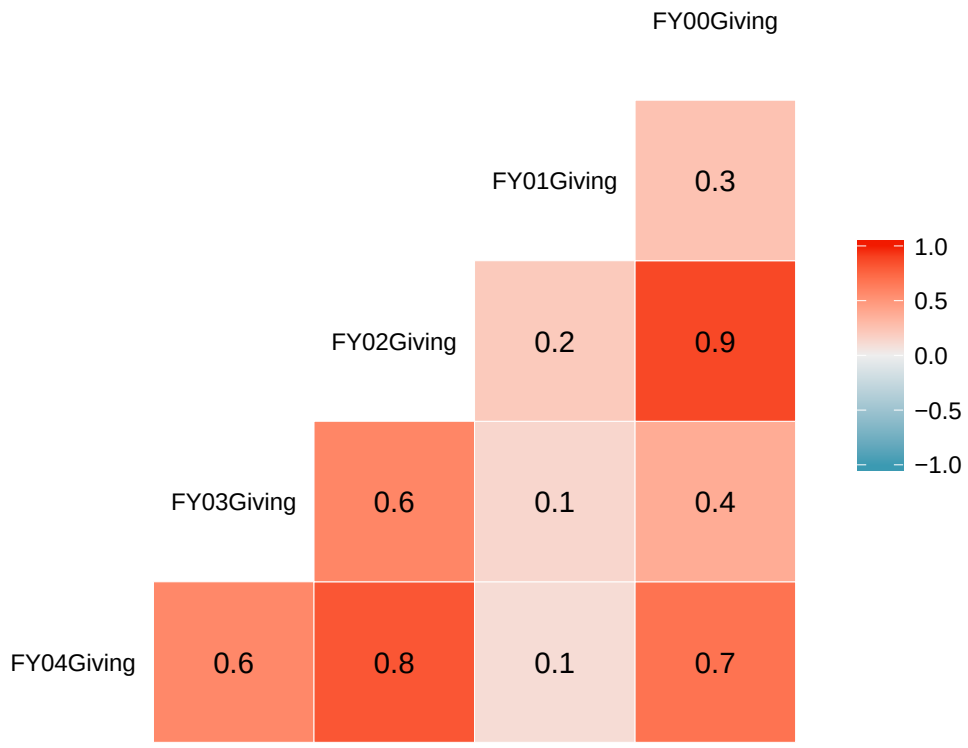
From this correlation graph we can infer that the years FY02Giving has a strong correlation with year FY04Giving and year FY00Giving has strong correlation with year FY02Giving. Also, year FY03Giving and FY04Giving have very little correlation with year FY01Giving.

```
library(GGally)
```

```
##
## Attaching package: 'GGally'
```

```
## The following object is masked from 'package:dplyr':
##
##     nasa
```

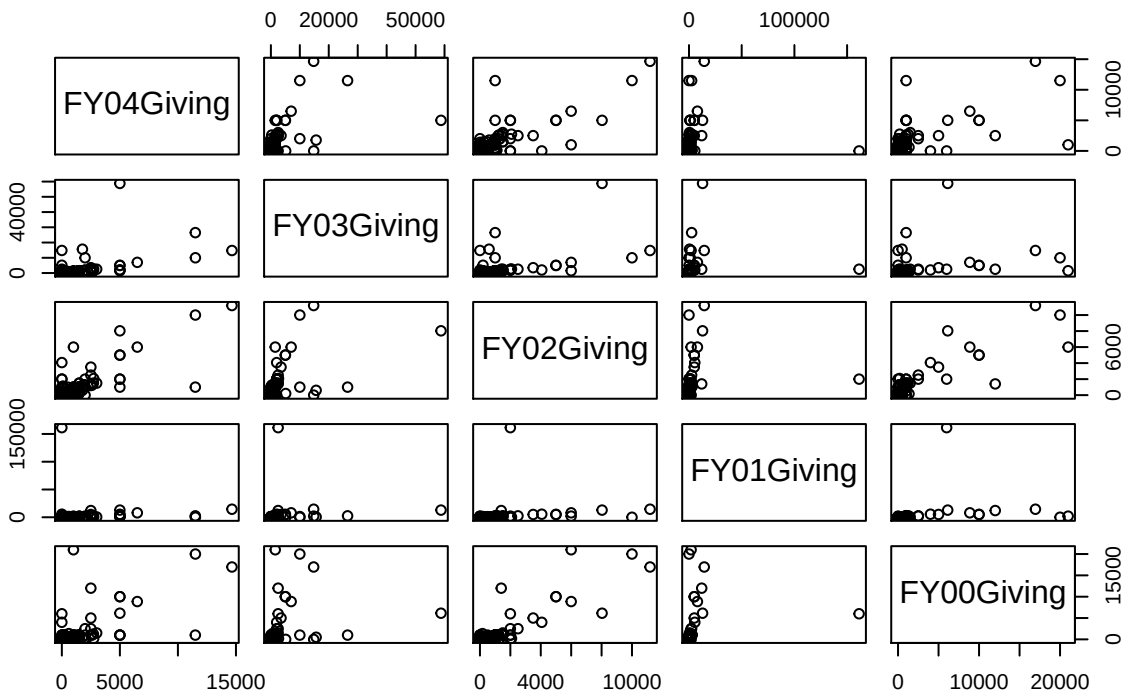
```
ggcorr(df.contributions[,6:10], label_size = 4, size = 3, palette = "BuRd", label = TRUE)
```



The above graph explains correlation between different Fiscal Years.

The above correlation values are positive meaning that the variables are directly proportional to each other. For eg: FY00Giving is strongly correlated to FY02Giving and FY04Giving.

```
pairs(df.contributions[,6:10], main="Relationships of Contributions from 2000-2004")
```



The above graph shows the relationship of all the years with each other and how they vry for each value of a particular column. For example if we see the first element of third row that the points almost overlap each other which shows that FY00 has strong correlation with FY02.

Regression Tree:

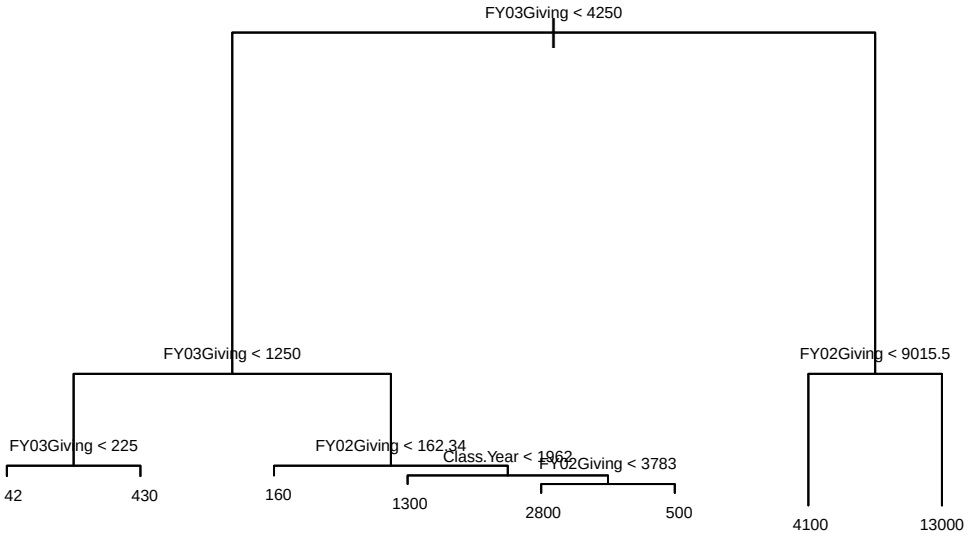
```
contribution <- read.csv("contribution.csv")
library(tree)
```

```
## Warning: package 'tree' was built under R version 3.4.2
```

```
pstree <- tree(FY04Giving ~Gender+Marital.Status+AttendenceEvent+Class.Year
               +FY03Giving+FY02Giving+FY02Giving+FY01Giving,
               data=contribution, mincut=1)
```

We have not considered variables like Major and Next.Degree for predicting FY04Giving as they have more than 32 levels (more than 32 distinct values in a column) which will produce an error while making the regression tree. We cannot categorize these distinct values into abstract categories for eg. There is no relationship between two majors like 'History' and 'Biology' cannot be grouped together and doing this will be a wrong approach.

```
library(rpart)
plot(pstree)
text(pstree, digits=2,cex=0.5)
```



For the above plotted graph we have taken the value of mincut as 1, which represents the minimum number of cuts required to crreate a new branch of the tree.

- As per the above obtained regression tree, we can see that for predicting the value of our target variable that is FY04Giving.

- We first have to see the value of variable FY03Giving.

- If the value of FY03Giving is less than 4250 then we move to the left side of the tree and see if the value of FY03Giving is gain less than 1250.

- Then we move to the left side of the tree again otherwise we move to the right side and see the value of FY02Giving and move so on.

- But, in the beginning, if the value of FY02Giving is more than 4250, the we see for the value of FY02Giving variable and if it is less than 9015.5 then the value of our Target variable i.e. FY04Giving is 4100 otherwise its value is 13000.

```
pstree
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 1230 771000000   159.40
##    2) FY03Giving < 4250 1219 183900000   109.20
##      4) FY03Giving < 1250 1190  36480000    66.23
##        8) FY03Giving < 225 1116  14770000    42.07 *
##        9) FY03Giving > 225 74  11220000   430.70 *
##      5) FY03Giving > 1250 29  55020000  1873.00
##       10) FY02Giving < 162.34 3     39730   162.60 *
##       11) FY02Giving > 162.34 26  45200000  2070.00
##         22) Class.Year < 1962 9   5056000  1278.00 *
##         23) Class.Year > 1962 17  31500000  2490.00
##           46) FY02Giving < 3783 15  22030000  2755.00 *
##           47) FY02Giving > 3783 2    500000   500.00 *
##    3) FY03Giving > 4250 11 243800000  5721.00
##      6) FY02Giving < 9015.5 9 106400000  4086.00 *
##      7) FY02Giving > 9015.5 2   4959000 13080.00 *
```

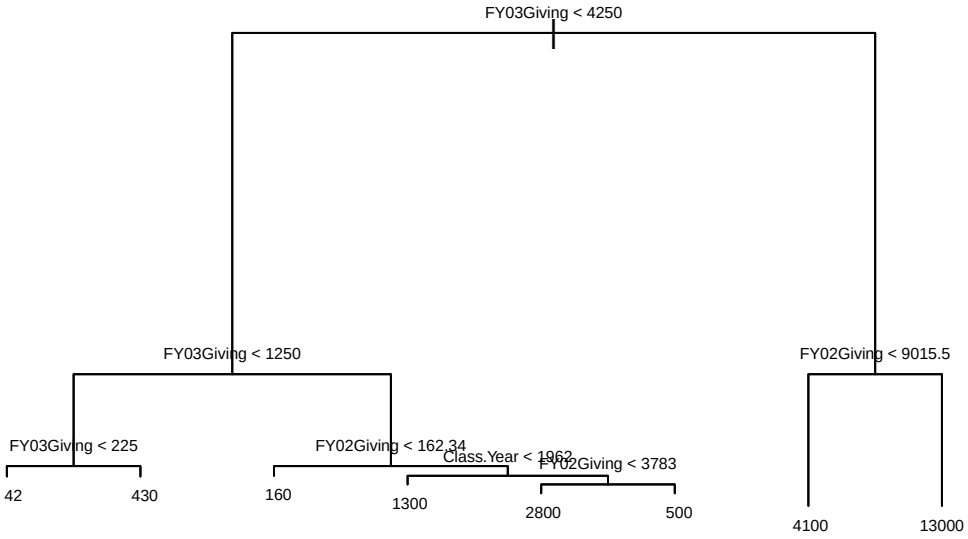
```
summary(pstree)
```

```
##
## Regression tree:
## tree(formula = FY04Giving ~ Gender + Marital.Status + AttendenceEvent +
##     Class.Year + FY03Giving + FY02Giving + FY02Giving + FY01Giving,
##     data = contribution, mincut = 1)
## Variables actually used in tree construction:
## [1] "FY03Giving" "FY02Giving" "Class.Year"
## Number of terminal nodes:  8
## Residual mean deviance:  135000 = 1.65e+08 / 1222
## Distribution of residuals:
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -4086.000   -42.070   -42.070     0.000     7.934  7414.000
```

Here we can see our regression tree that has been shown above, but in the form of levels. Each level has its own value and the value of deviances for the target variable FY04Giving at that level.

```
pstree2 <- tree(FY04Giving ~Gender+Marital.Status+AttendenceEvent+Class.Year
                +FY03Giving+FY02Giving+FY02Giving+FY01Giving,
                data=contribution, mincut=2)
plot(pstree2)
text(pstree2, digits=2,cex=0.5)
```



Here we have used mincut value 2 and it will only create a branch if we have more than 2 values in a cut.

```
pstree2
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 1230 771000000   159.40
##    2) FY03Giving < 4250 1219 183900000   109.20
```

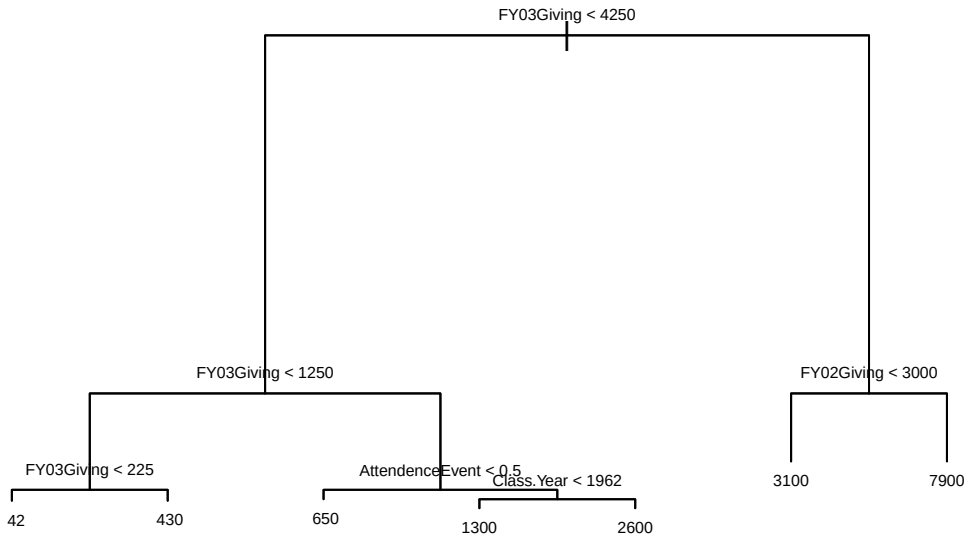
```
##      4) FY03Giving < 1250 1190  36480000    66.23
##        8) FY03Giving < 225 1116  14770000    42.07 *
##        9) FY03Giving > 225 74  11220000   430.70 *
##      5) FY03Giving > 1250 29  55020000  1873.00
##       10) FY02Giving < 162.34 3     39730   162.60 *
##       11) FY02Giving > 162.34 26  45200000  2070.00
##         22) Class.Year < 1962 9   5056000  1278.00 *
##         23) Class.Year > 1962 17  31500000  2490.00
##           46) FY02Giving < 3783 15  22030000  2755.00 *
##           47) FY02Giving > 3783 2    500000   500.00 *
##    3) FY03Giving > 4250 11 243800000  5721.00
##      6) FY02Giving < 9015.5 9 106400000  4086.00 *
##      7) FY02Giving > 9015.5 2   4959000 13080.00 *
```

```
summary(pstree2)
```

```
##
## Regression tree:
## tree(formula = FY04Giving ~ Gender + Marital.Status + AttendenceEvent +
##     Class.Year + FY03Giving + FY02Giving + FY02Giving + FY01Giving,
##     data = contribution, mincut = 2)
## Variables actually used in tree construction:
## [1] "FY03Giving" "FY02Giving" "Class.Year"
## Number of terminal nodes:  8
## Residual mean deviance:  135000 = 1.65e+08 / 1222
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -4086.000   -42.070   -42.070     0.000     7.934  7414.000
```

Here we can see that that the Residual mean Deviance is the least and is 135000, which is the least when co mpared to all the mincut values from 2 to 5. So, we will consider this for our cross validation analysis.

```
pstree4 <- tree(FY04Giving ~Gender+Marital.Status+AttendenceEvent+Class.Year
                +FY03Giving+FY02Giving+FY02Giving+FY01Giving,
                data=contribution, mincut=5)
plot(pstree4)
text(pstree4, digits=2,cex=0.5)
```



Here we can see that the nodes 46 and 47 no more exist in our tree and our tree size has reduced.

```
pstree4
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 1230 771000000  159.40
##    2) FY03Giving < 4250 1219 183900000  109.20
##      4) FY03Giving < 1250 1190  36480000   66.23
##        8) FY03Giving < 225 1116  14770000   42.07 *
##        9) FY03Giving > 225 74  11220000  430.70 *
##      5) FY03Giving > 1250 29  55020000 1873.00
##       10) AttendenceEvent < 0.5 5   2440000  648.80 *
##       11) AttendenceEvent > 0.5 24  43530000 2128.00
##         22) Class.Year < 1962 8   4455000 1280.00 *
##         23) Class.Year > 1962 16  30450000 2552.00 *
##    3) FY03Giving > 4250 11 243800000 5721.00
##      6) FY02Giving < 3000 5  92750000 3054.00 *
##      7) FY02Giving > 3000 6  85810000 7944.00 *
```

```
summary(pstree4)
```

```
##
```

```
## Regression tree:
## tree(formula = FY04Giving ~ Gender + Marital.Status + AttendenceEvent +
##     Class.Year + FY03Giving + FY02Giving + FY02Giving + FY01Giving,
##     data = contribution, mincut = 5)
## Variables actually used in tree construction:
## [1] "FY03Giving"      "AttendenceEvent" "Class.Year"      "FY02Giving"
## Number of terminal nodes:  7
## Residual mean deviance:  197800 = 241900000 / 1223
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -3054.000   -42.070   -42.070     0.000     7.934  8446.000
```

```
set.seed(2)
cvpst <- cv.tree(pstree2, K=10)
cvpst$size
```

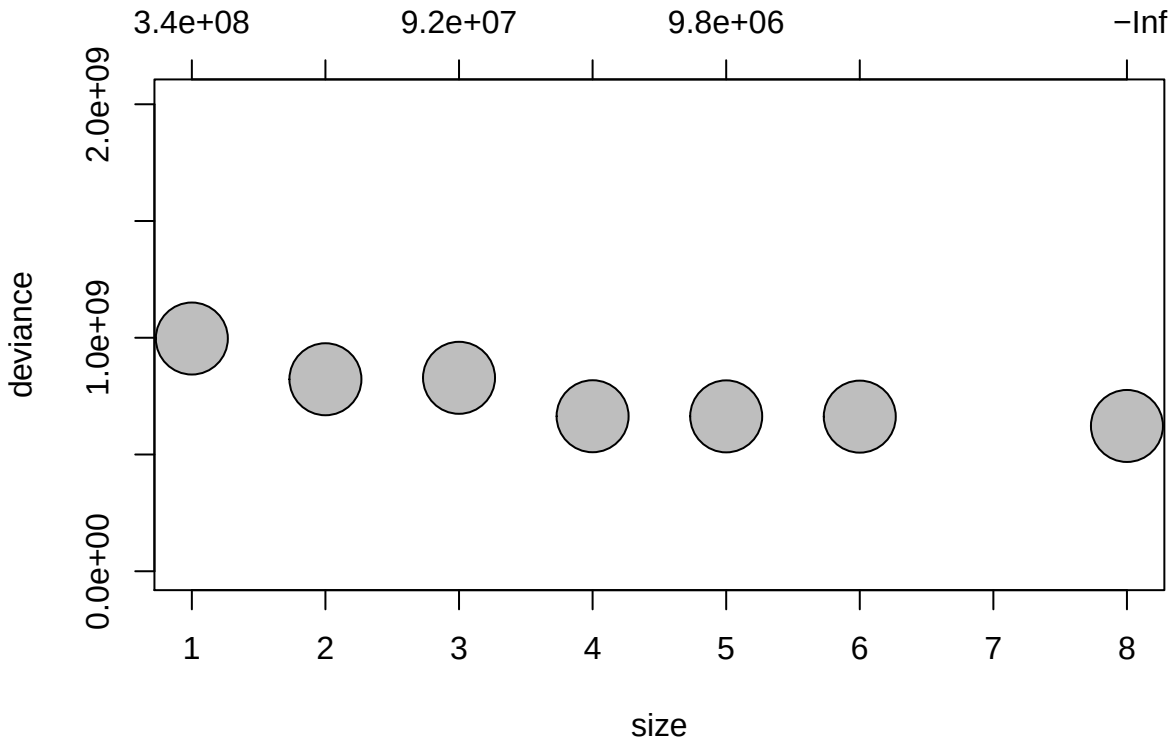
```
## [1] 8 6 5 4 3 2 1
```

```
cvpst$dev
```

```
## [1] 622226914 662185264 663242711 663806876 828430609 822243433 996715580
```

From the above obtained results, we can infer that we have huge deviances at each fold of our cross validation and the minimum deviance occurs at the 8th fold, which we take as our best value.

```
plot(cvpst, pch=21, bg=8, type="p", cex=5, ylim=c(0,2025316401))
```

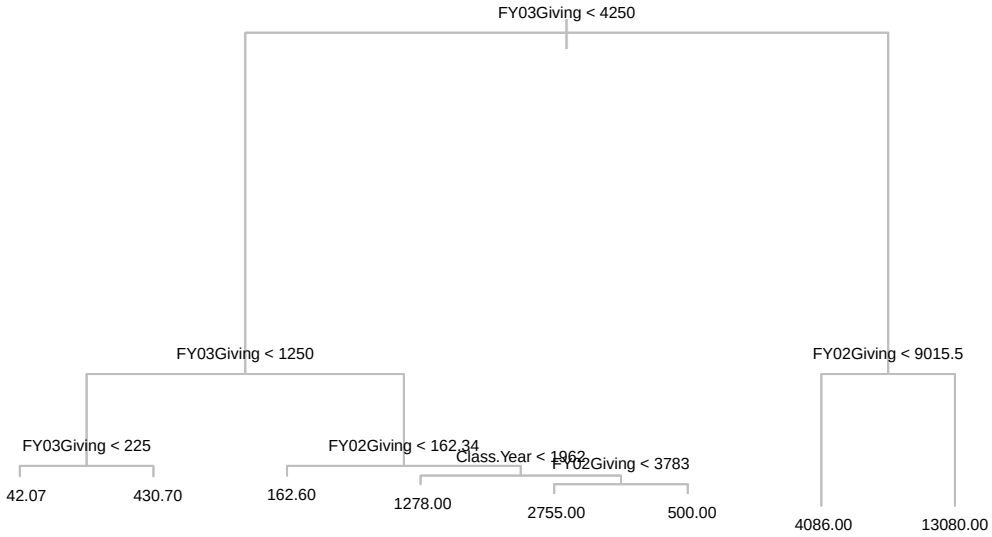


This plot shows the deviances of the tree based on the sizes and we can see that the minimum deviance occurs at the eighth cross validation fold.

```
pstcut3 <- prune.tree(pstree2, best=8)
pstcut3
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 1230 771000000   159.40
##    2) FY03Giving < 4250 1219 183900000   109.20
##      4) FY03Giving < 1250 1190  36480000    66.23
##        8) FY03Giving < 225 1116  14770000    42.07 *
##        9) FY03Giving > 225 74  11220000   430.70 *
##      5) FY03Giving > 1250 29  55020000  1873.00
##       10) FY02Giving < 162.34 3     39730   162.60 *
##       11) FY02Giving > 162.34 26  45200000  2070.00
##         22) Class.Year < 1962 9   5056000  1278.00 *
##         23) Class.Year > 1962 17  31500000  2490.00
##           46) FY02Giving < 3783 15  22030000  2755.00 *
##           47) FY02Giving > 3783 2    500000   500.00 *
##    3) FY03Giving > 4250 11 243800000  5721.00
##      6) FY02Giving < 9015.5 9 106400000  4086.00 *
##      7) FY02Giving > 9015.5 2   4959000 13080.00 *
```

```
plot(pstcut3, col=8)
text(pstcut3, cex=0.5)
```



From the obtained tree we can see our pruned tree at our best cross validation value that is for the 8th cross validation. We need not prune the tree further as the tree is good enough and does not involve much levels of complexity. So, we can easily see that our tree model is not overfitting and this is the best results that we can obtain through regression trees algorithm for predicting our traget variable value FY04Giving.