CMDA4654Project

Matt

May 7, 2019

Matthew Dorris

READ DATA

```
rm(list=ls())
dev.off()
```

```
## null device
##           1
```

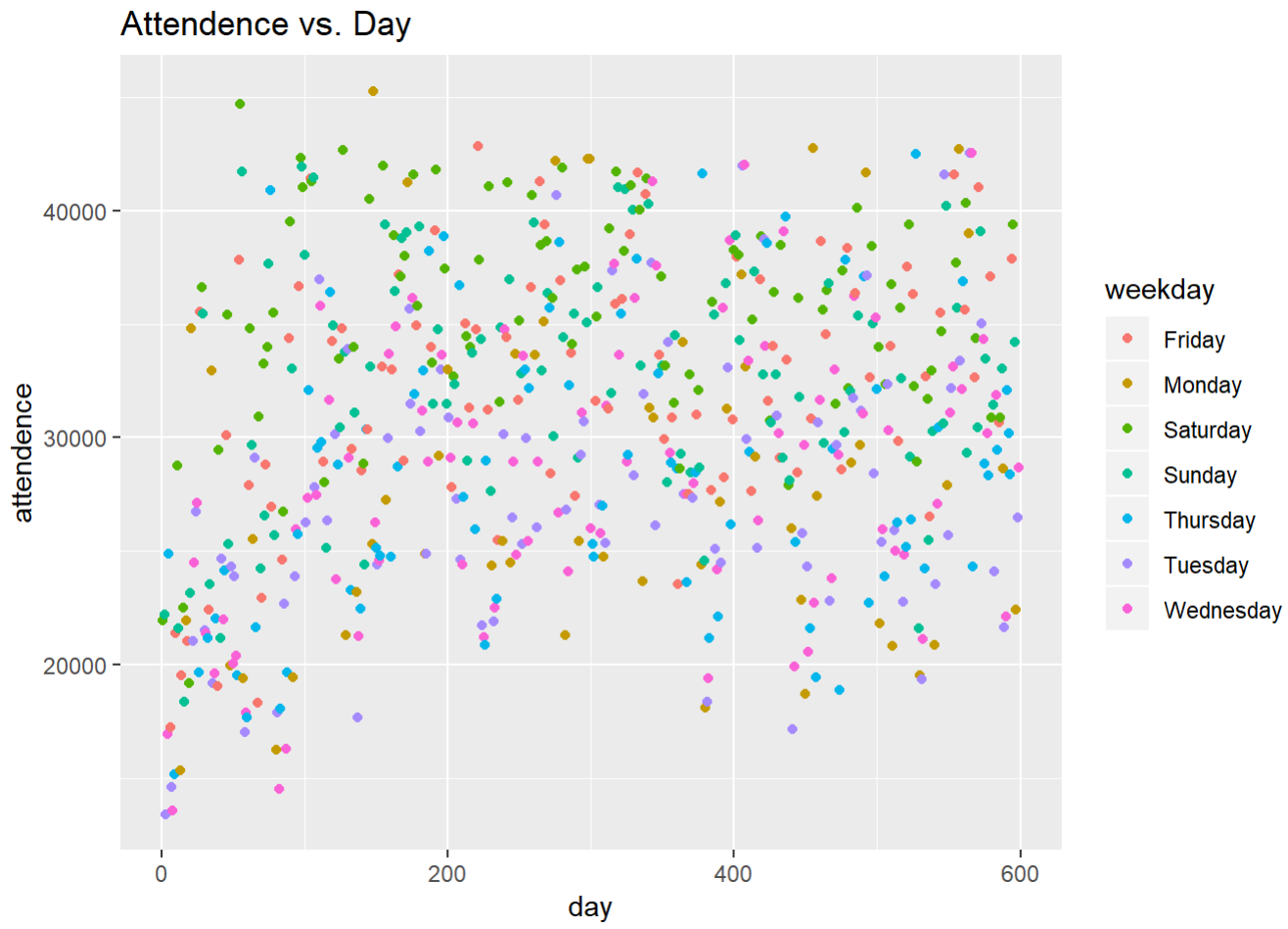
```
#reading the data
data <- read.csv(file = "C:/College/Senior/data.csv")
head(data)
```

```
##                        date year day attendence   weekday month  time temp
## 1   Saturday, April 2, 2011 2011   1      21941  Saturday April   Day   43
## 2     Sunday, April 3, 2011 2011   2      22210    Sunday April   Day   53
## 3   Tuesday, April 12, 2011 2011   3      13413   Tuesday April Night   55
## 4 Wednesday, April 13, 2011 2011   4      16914 Wednesday April Night   56
## 5  Thursday, April 14, 2011 2011   5      24875  Thursday April Night   69
## 6    Friday, April 15, 2011 2011   6      17217    Friday April Night   56
```

```
data$attendence <- as.numeric(data$attendence)

set.seed(0)
n = length(data$attendence)
trainIDX = sample(1:n, round(.7*n))
train = data[trainIDX,]
test = data[-trainIDX,]
```

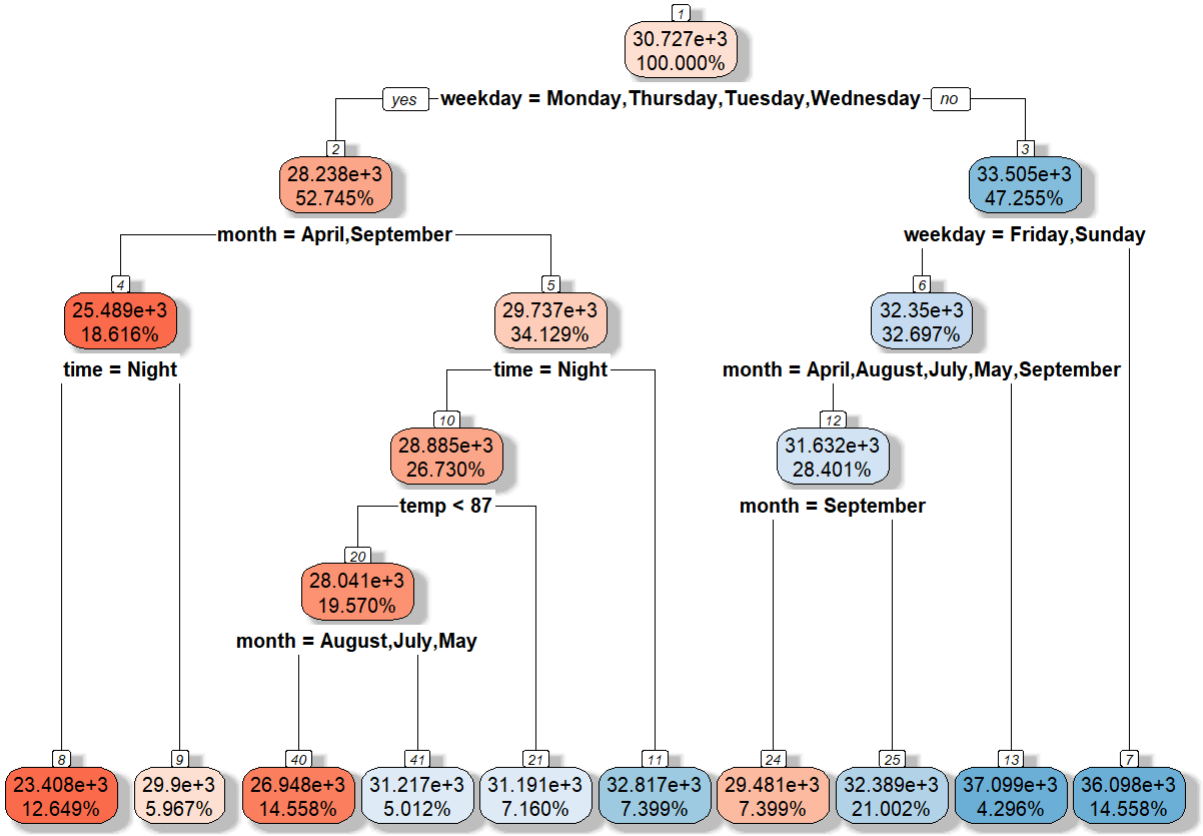
Exploratory data analysis graph

```
#exploratory data analysis graph
library(ggplot2)
ggplot(data, aes(x=day, y=attendence, color=weekday)) + geom_point() + ggtitle("Attendence vs. D
ay")
```



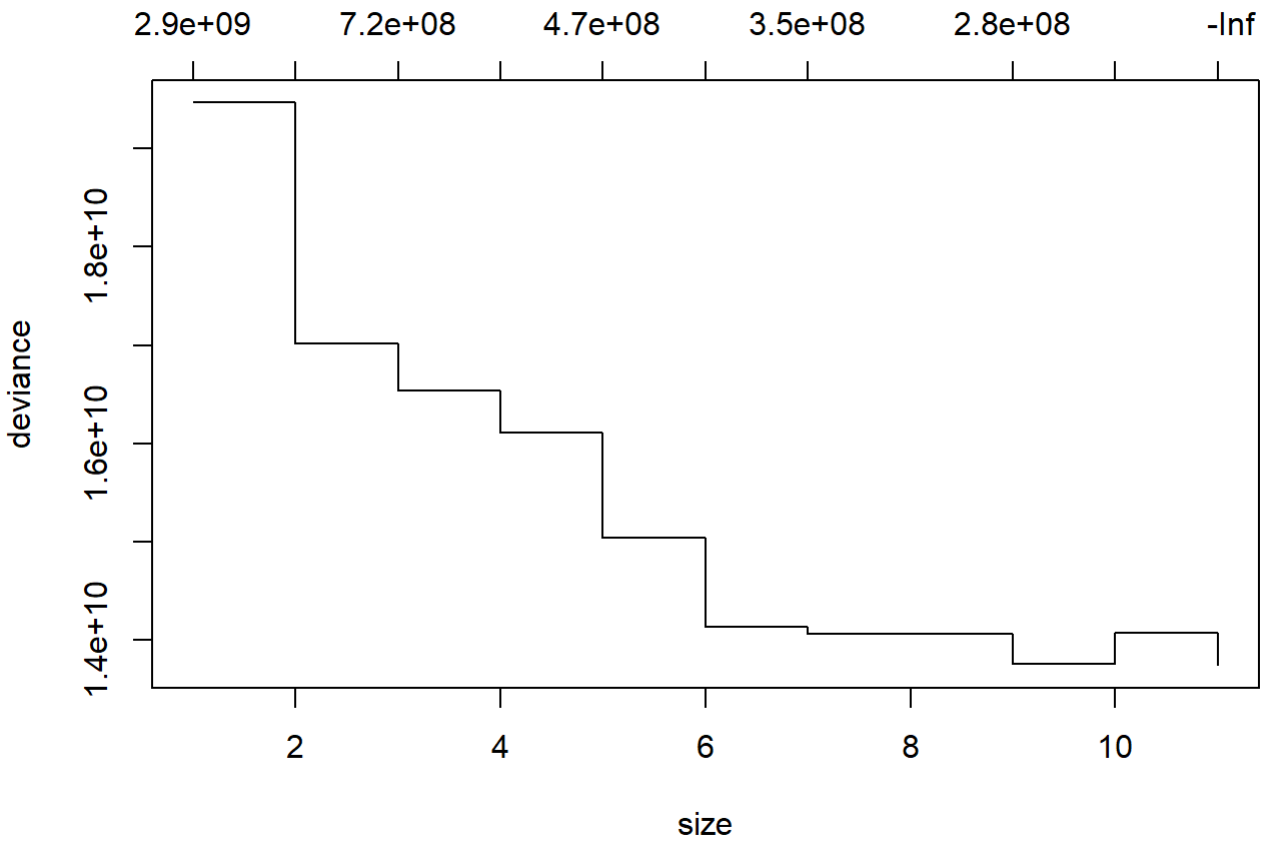
Full Tree with rPART

```
library(rpart)
library(rpart.plot)

tree2 <- rpart(attendence ~ weekday + month + temp + time, data = train)
rpart.plot(tree2, box.palette="RdBu", shadow.col="gray", nn=TRUE, digits=5)
```

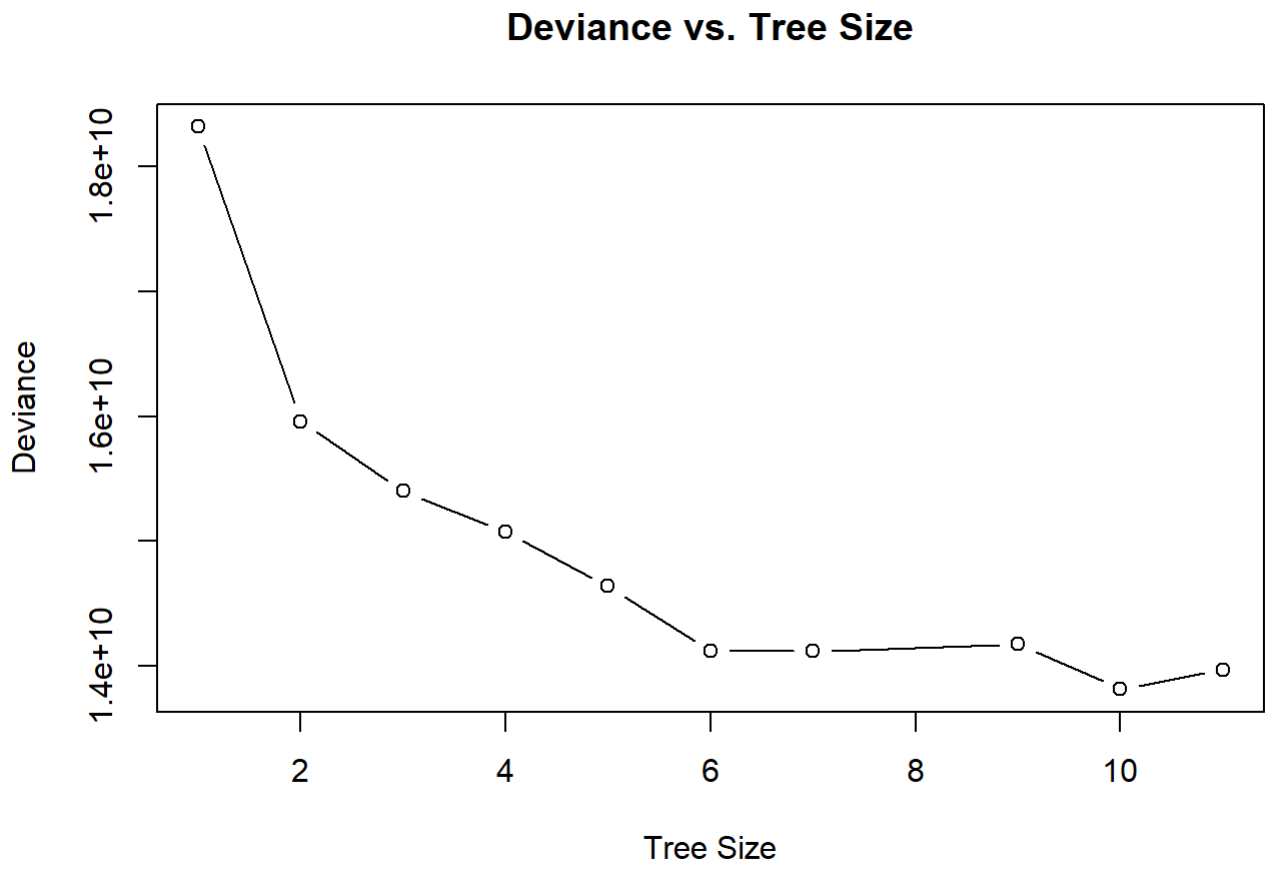


Plots of Deviation

```
library(tree)
tree1 <- tree(attendence ~ weekday + month + temp + time, data = train)

#plot of deviation
cvpst <- cv.tree(tree1, K=90)
plot(cvpst)
```

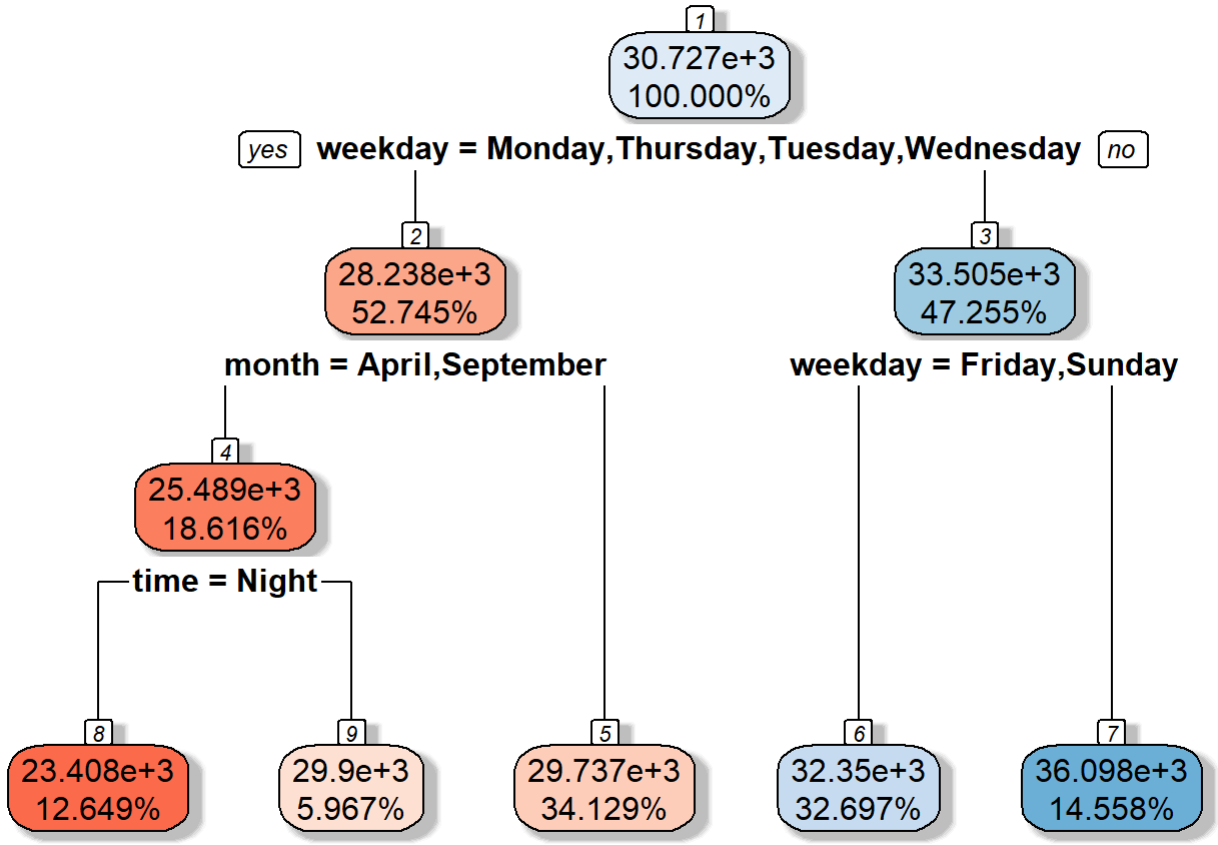



```
cvTree <-  cv.tree(tree1, FUN=prune.tree)
plot(cvTree$size, cvTree$dev, type="b", main = "Deviance vs. Tree Size", xlab ="Tree Size", ylab
= "Deviance")
```



Pruned Trees

```
cp.select <- function(big.tree) {
  min.x <- which.min(big.tree$cptable[, 4])
  for(i in 1:nrow(big.tree$cptable)) {
    if(big.tree$cptable[i, 4] < big.tree$cptable[min.x, 4] + big.tree$cptable[min.x, 5]) return
(big.tree$cptable[i, 1])
  }
}
pruned.tree <- prune(tree2, cp = cp.select(tree2))
rpart.plot(pruned.tree, box.palette="RdBu", shadow.col="gray", nn=TRUE, digits=5)
```



Predictions

```
predPruned <-  predict(pruned.tree, test)
predTree2 <-  predict(tree2, test)

RMSE_ = function(a, b){
  sqrt((mean((a - b)^2)))
}

RMSE_(predPruned, test$attendence)
```

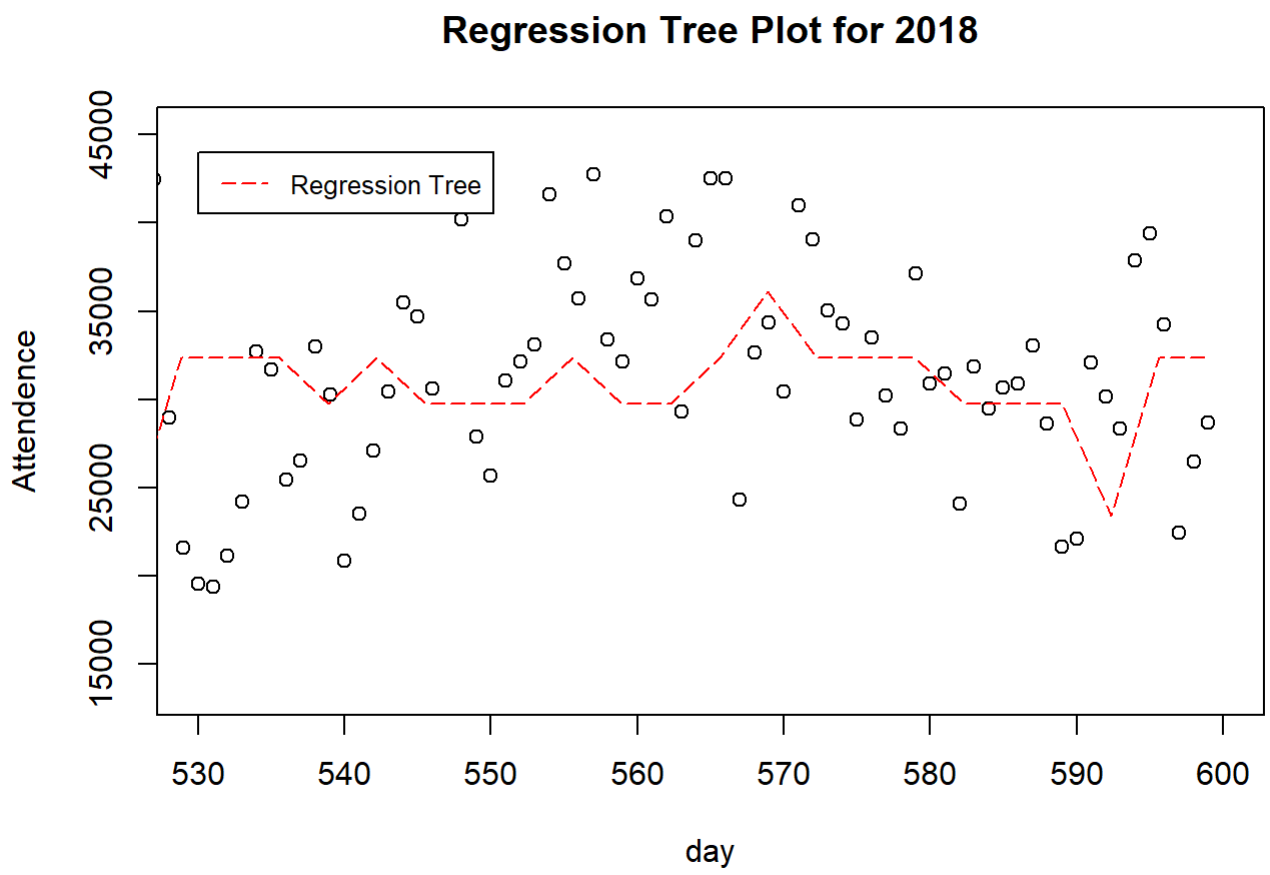
```
## [1] 5731.939
```

```
RMSE_(predTree2, test$attendence)
```

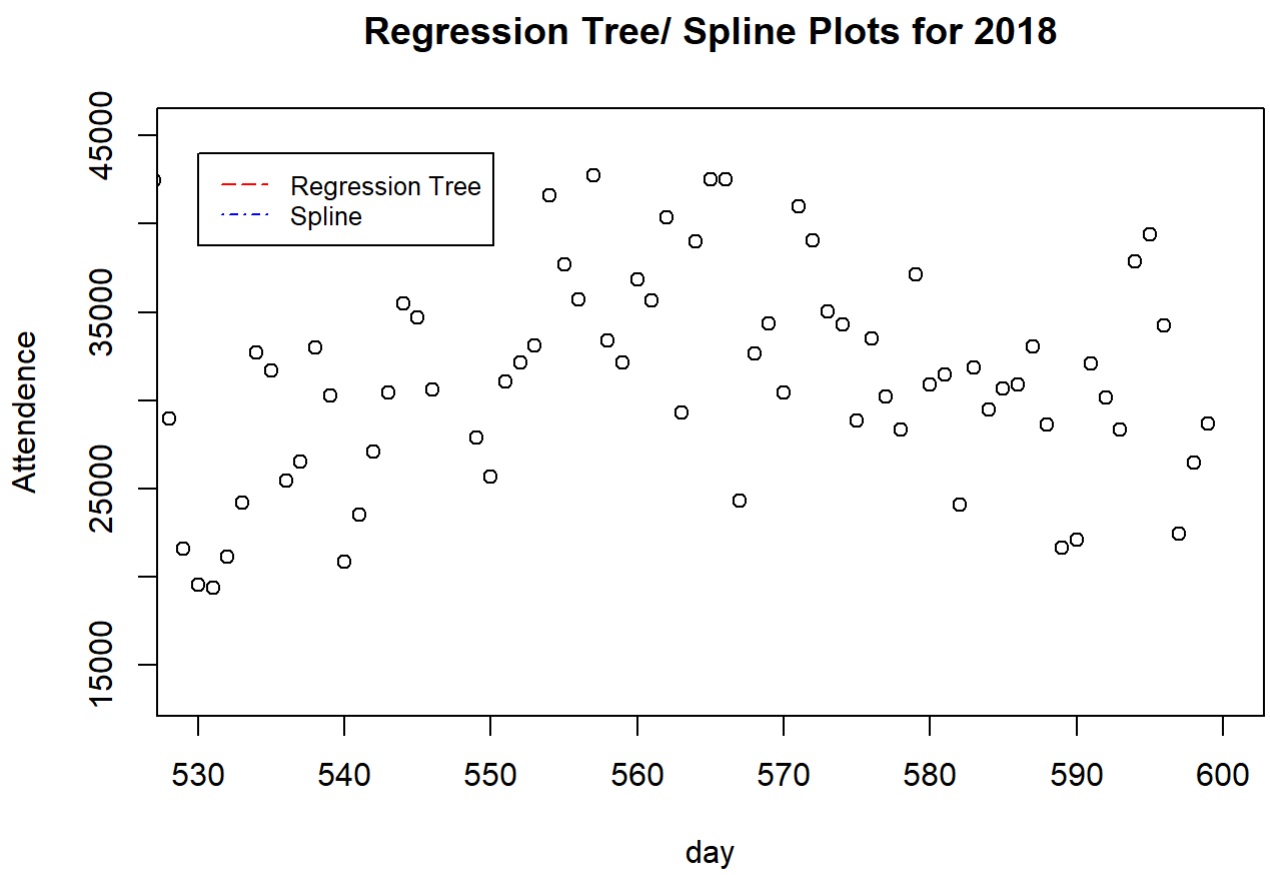
```
## [1] 5677.867
```

Plots of both Trees

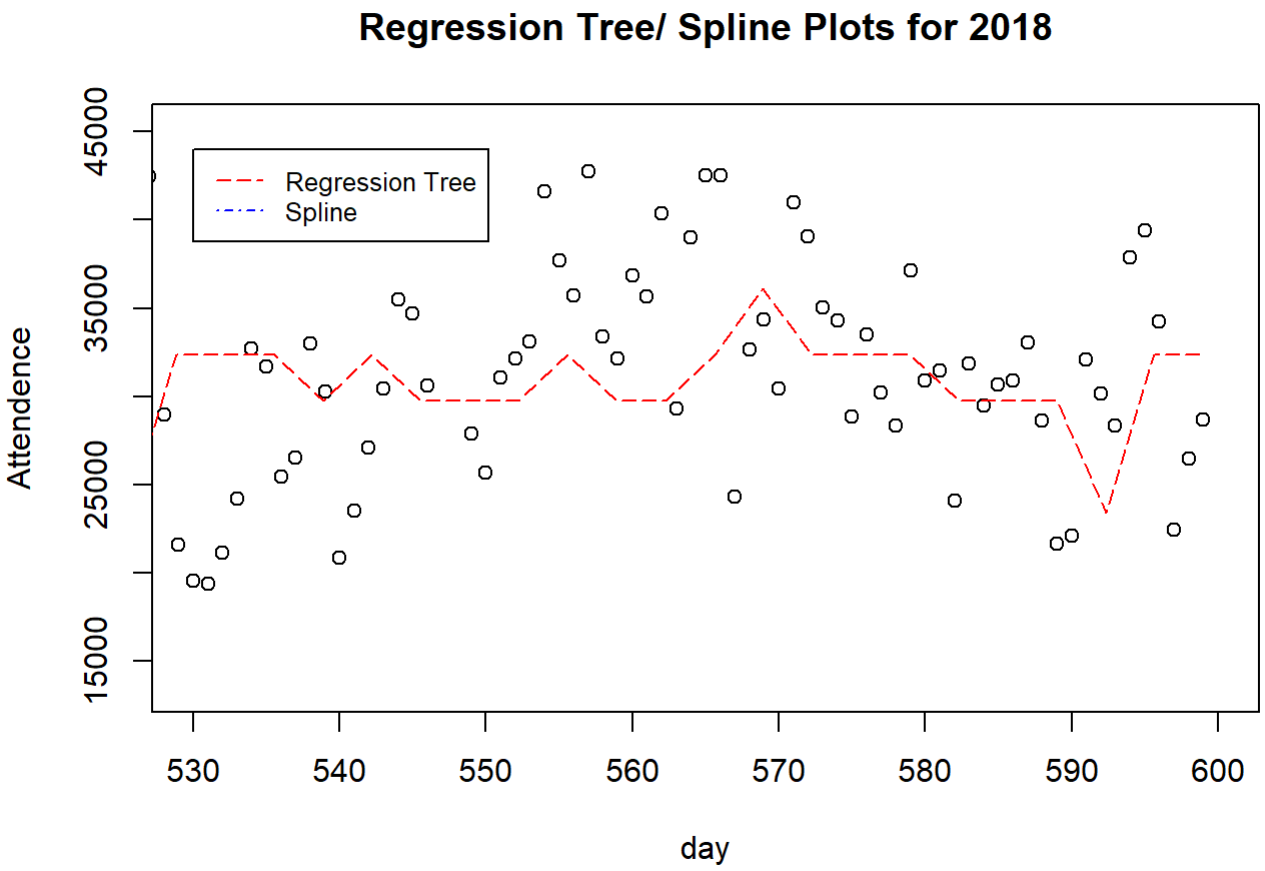
```
plot(data$day, data$attendence, xlim=c(530,600), main = "Regression Tree Plot for 2018", xlab =
"day", ylab = "Attendence")
legend(530, 44000, legend="Regression Tree", col="red", lty=5, cex=0.8)
g <- seq(1, 599, length = length(predPruned))

matlines(g, predPruned, col = 2, lty = 5)
```




```
plot(data$day, data$attendence, xlim=c(530,600), main = "Regression Tree/ Spline Plots for 2018"
, xlab ="day", ylab = "Attendence")
legend(530, 44000, legend=c("Regression Tree", "Spline"), col=c("red", "blue"), lty=5:1, cex=0.8
)
```



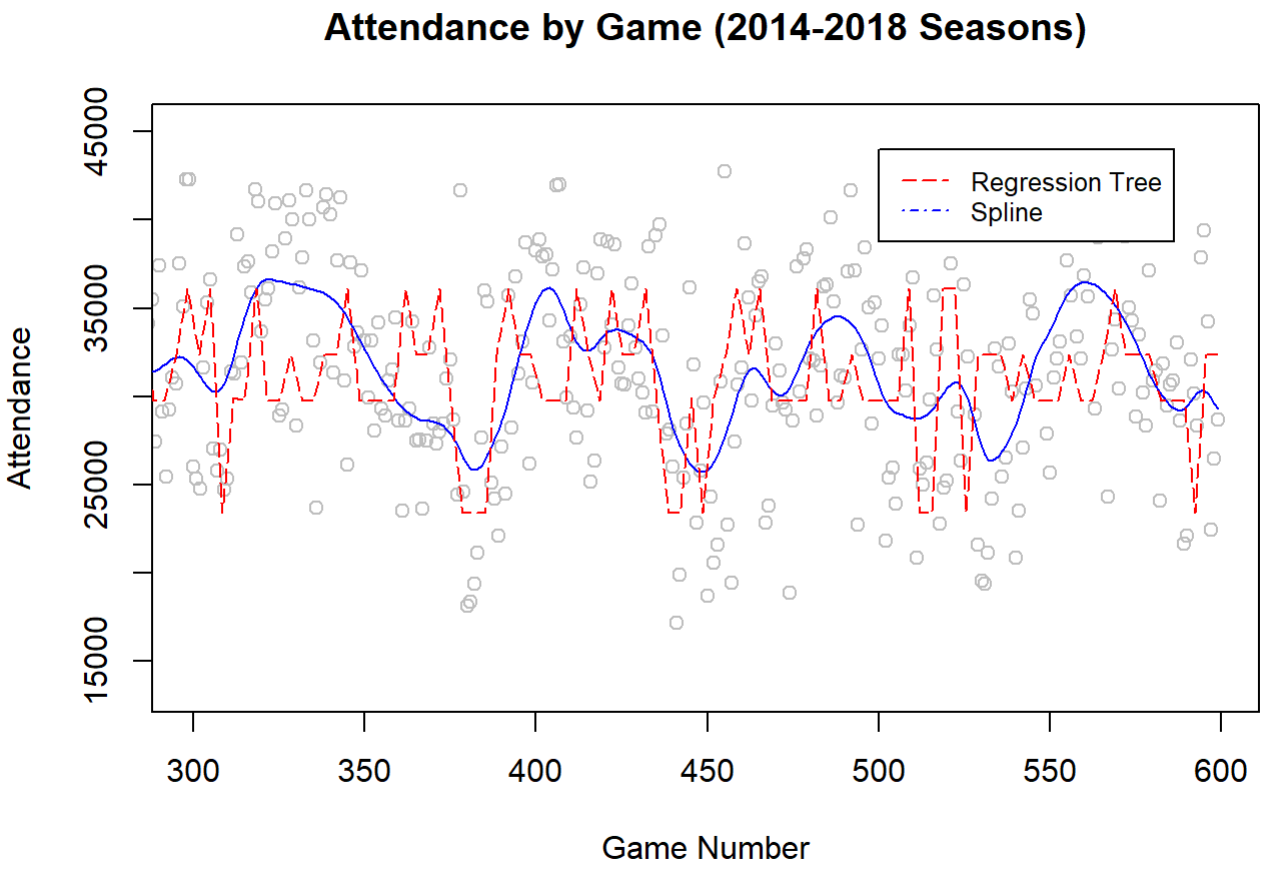

```
g <- seq(1, 599, length = length(predPruned))
plot(data$day, data$attendence, xlim=c(530,600), main = "Regression Tree/ Spline Plots for 2018"
, xlab ="day", ylab = "Attendence")
legend(530, 44000, legend=c("Regression Tree", "Spline"), col=c("red", "blue"), lty=5:1, cex=0.8
)
matlines(g, predPruned, col = 2, lty = 5)
```




```
#COMBINING regression tree and spline
spline3 <- smooth.spline(seq(1, 599, length.out = 599), data$attendence, seq(1, 599, length.out
 = 599))

spline.pred3 <- predict(spline3, newdata = data)
RMSE_ = function(a, b){
  sqrt((mean((a - b)^2)))
}
RMSE_(spline.pred3$y, data$attendence)
```

```
## [1] 5249.825
```

```
{plot(data$attendence ~ seq(1, 599, length.out = 599), col = "gray", xlim = c(300, 599), main =
"Attendance by Game (2014-2018 Seasons)", xlab = "Game Number", ylab = "Attendance")
lines(spline.pred3$y ~ spline.pred3$x, col = "blue")
legend(500, 44000, legend=c("Regression Tree", "Spline"), col=c("red", "blue"), lty=5:1, cex=0.8
)
g <- seq(1, 599, length = length(predPruned))
matlines(g, predPruned, col = 2, lty = 5)}
```




```
g <- seq(1, 599, length = length(predPruned))
```