HW4

Group BUAN6356002 4

11/11/2019

**CLASS**: “BUAN 6356”  
**GROUP MEMBERS**: “Athisaran Beekar, Femina Pereira, Sinduja Senthil Kumar, Tamanna Kawatra”

### a. Load the packages:

if(!require("pacman")) install.packages("pacman")

## Loading required package: pacman

pacman::p\_load(caret, data.table, ISLR, tidyr, devtools, ggplot2, tidyverse, devtools,gains, leaps, rpart, rpart.plot, gbm,randomForest, tinytex)

## Installing package into 'C:/Users/martin pereira/Documents/R/win-library/3.6'  
## (as 'lib' is unspecified)

## Warning: unable to access index for repository http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.6:  
## cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.6/PACKAGES'

## package 'devtools' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\martin pereira\AppData\Local\Temp\RtmpE3mqX2\downloaded\_packages

##   
## devtools installed

## Installing package into 'C:/Users/martin pereira/Documents/R/win-library/3.6'  
## (as 'lib' is unspecified)

## Warning: unable to access index for repository http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.6:  
## cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.6/PACKAGES'

## package 'devtools' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\martin pereira\AppData\Local\Temp\RtmpE3mqX2\downloaded\_packages

##   
## devtools installed

## Warning in pacman::p\_load(caret, data.table, ISLR, tidyr, devtools, ggplot2, : Failed to install/load:  
## devtools, devtools

search()

## [1] ".GlobalEnv" "package:tinytex" "package:randomForest"  
## [4] "package:gbm" "package:rpart.plot" "package:rpart"   
## [7] "package:leaps" "package:gains" "package:forcats"   
## [10] "package:stringr" "package:dplyr" "package:purrr"   
## [13] "package:readr" "package:tibble" "package:tidyverse"   
## [16] "package:usethis" "package:tidyr" "package:ISLR"   
## [19] "package:data.table" "package:caret" "package:ggplot2"   
## [22] "package:lattice" "package:pacman" "package:stats"   
## [25] "package:graphics" "package:grDevices" "package:utils"   
## [28] "package:datasets" "package:methods" "Autoloads"   
## [31] "package:base"

### b. Exploring the dataset:

dim(Hitters)

## [1] 322 20

str(Hitters)

## 'data.frame': 322 obs. of 20 variables:  
## $ AtBat : int 293 315 479 496 321 594 185 298 323 401 ...  
## $ Hits : int 66 81 130 141 87 169 37 73 81 92 ...  
## $ HmRun : int 1 7 18 20 10 4 1 0 6 17 ...  
## $ Runs : int 30 24 66 65 39 74 23 24 26 49 ...  
## $ RBI : int 29 38 72 78 42 51 8 24 32 66 ...  
## $ Walks : int 14 39 76 37 30 35 21 7 8 65 ...  
## $ Years : int 1 14 3 11 2 11 2 3 2 13 ...  
## $ CAtBat : int 293 3449 1624 5628 396 4408 214 509 341 5206 ...  
## $ CHits : int 66 835 457 1575 101 1133 42 108 86 1332 ...  
## $ CHmRun : int 1 69 63 225 12 19 1 0 6 253 ...  
## $ CRuns : int 30 321 224 828 48 501 30 41 32 784 ...  
## $ CRBI : int 29 414 266 838 46 336 9 37 34 890 ...  
## $ CWalks : int 14 375 263 354 33 194 24 12 8 866 ...  
## $ League : Factor w/ 2 levels "A","N": 1 2 1 2 2 1 2 1 2 1 ...  
## $ Division : Factor w/ 2 levels "E","W": 1 2 2 1 1 2 1 2 2 1 ...  
## $ PutOuts : int 446 632 880 200 805 282 76 121 143 0 ...  
## $ Assists : int 33 43 82 11 40 421 127 283 290 0 ...  
## $ Errors : int 20 10 14 3 4 25 7 9 19 0 ...  
## $ Salary : num NA 475 480 500 91.5 750 70 100 75 1100 ...  
## $ NewLeague: Factor w/ 2 levels "A","N": 1 2 1 2 2 1 1 1 2 1 ...

Hitters.df <- data.frame(Hitters)  
summary(Hitters.df)

## AtBat Hits HmRun Runs   
## Min. : 16.0 Min. : 1 Min. : 0.00 Min. : 0.00   
## 1st Qu.:255.2 1st Qu.: 64 1st Qu.: 4.00 1st Qu.: 30.25   
## Median :379.5 Median : 96 Median : 8.00 Median : 48.00   
## Mean :380.9 Mean :101 Mean :10.77 Mean : 50.91   
## 3rd Qu.:512.0 3rd Qu.:137 3rd Qu.:16.00 3rd Qu.: 69.00   
## Max. :687.0 Max. :238 Max. :40.00 Max. :130.00   
##   
## RBI Walks Years CAtBat   
## Min. : 0.00 Min. : 0.00 Min. : 1.000 Min. : 19.0   
## 1st Qu.: 28.00 1st Qu.: 22.00 1st Qu.: 4.000 1st Qu.: 816.8   
## Median : 44.00 Median : 35.00 Median : 6.000 Median : 1928.0   
## Mean : 48.03 Mean : 38.74 Mean : 7.444 Mean : 2648.7   
## 3rd Qu.: 64.75 3rd Qu.: 53.00 3rd Qu.:11.000 3rd Qu.: 3924.2   
## Max. :121.00 Max. :105.00 Max. :24.000 Max. :14053.0   
##   
## CHits CHmRun CRuns CRBI   
## Min. : 4.0 Min. : 0.00 Min. : 1.0 Min. : 0.00   
## 1st Qu.: 209.0 1st Qu.: 14.00 1st Qu.: 100.2 1st Qu.: 88.75   
## Median : 508.0 Median : 37.50 Median : 247.0 Median : 220.50   
## Mean : 717.6 Mean : 69.49 Mean : 358.8 Mean : 330.12   
## 3rd Qu.:1059.2 3rd Qu.: 90.00 3rd Qu.: 526.2 3rd Qu.: 426.25   
## Max. :4256.0 Max. :548.00 Max. :2165.0 Max. :1659.00   
##   
## CWalks League Division PutOuts Assists   
## Min. : 0.00 A:175 E:157 Min. : 0.0 Min. : 0.0   
## 1st Qu.: 67.25 N:147 W:165 1st Qu.: 109.2 1st Qu.: 7.0   
## Median : 170.50 Median : 212.0 Median : 39.5   
## Mean : 260.24 Mean : 288.9 Mean :106.9   
## 3rd Qu.: 339.25 3rd Qu.: 325.0 3rd Qu.:166.0   
## Max. :1566.00 Max. :1378.0 Max. :492.0   
##   
## Errors Salary NewLeague  
## Min. : 0.00 Min. : 67.5 A:176   
## 1st Qu.: 3.00 1st Qu.: 190.0 N:146   
## Median : 6.00 Median : 425.0   
## Mean : 8.04 Mean : 535.9   
## 3rd Qu.:11.00 3rd Qu.: 750.0   
## Max. :32.00 Max. :2460.0   
## NA's :59

#### Question 1: Remove the observations with unknown salary information. How many observations were removed in this process?

sum(is.na(Hitters.df$Salary))

## [1] 59

Hitters1.df <- Hitters.df %>% drop\_na()  
  
# Verification of no 'NA' values in SALARy  
sapply(Hitters1.df$Salary, function(Salary) sum(length(which(is.na(Salary)))))

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [106] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [141] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [176] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [211] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [246] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

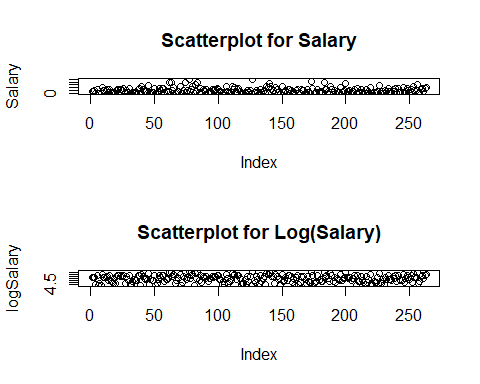
dim(Hitters1.df)

## [1] 263 20

# *Interpretation 1. From the sum function it is evident that the salary information is not present in 59 observations . Therefore these 59 observations out of 322 observation is removed, resulting in total of 263 records in the new dataframe Hitters1.df*

#### Question 2: Generate log-transform the salaries. Can you justify this transformation?

par(mfrow = c(2,1))  
plot(Hitters1.df$Salary,main = "Scatterplot for Salary", ylab ="Salary")  
logSalary <-log(Hitters1.df$Salary)  
plot(logSalary, main = "Scatterplot for Log(Salary)")

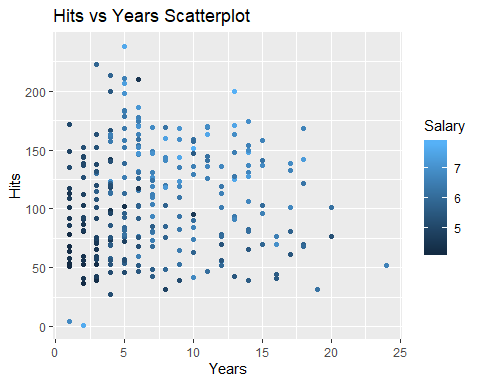


Hitters1.df$Salary <- log(Hitters1.df$Salary)

# *Interpretation 2. In the plot of salary, all points are skewed to the bottom and there are also outliers and hence log transformation is used which gives a better distribution. This is eveident from the scatter plots over SALARY with and without log transformation.The log transformation can be used to make highly skewed distributions less skewed. This can be valuable both for making patterns in the data more interpretable and for helping to meet the assumptions of inferential statistics.*

#### Question 3:Create a scatterplot with Hits on the y-axis and Years on the x-axis using all the observations. Color code the observations using the log Salary variable. What patterns do you notice on this chart, if any?

ggplot(Hitters1.df, aes(x = Years, y = Hits, color = Salary))+  
 geom\_point()+  
 ggtitle("Hits vs Years Scatterplot")

 # *Interpretation 3. The scatterplot clearly indicates that the player who has been playing for large number of years in the major leagues and who gave the higher number of hits in the year 1986 has higher salary to the player who has been playing for lesser number of years in the major leagues and who gave the lesser number of hits in the year 1986*

#### Question 4: Run a linear regression model of Log Salary on all the predictors using the entire dataset. Use regsubsets() function to perform best subset selection from the regression model. Identify the best model using BIC. Which predictor variables are included in this (best) model?

linear\_regression <- lm(Salary ~ ., data = Hitters1.df)  
summary(linear\_regression)

##   
## Call:  
## lm(formula = Salary ~ ., data = Hitters1.df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.22870 -0.45350 0.09424 0.40474 2.77223   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.618e+00 1.765e-01 26.171 < 2e-16 \*\*\*  
## AtBat -2.984e-03 1.232e-03 -2.421 0.01620 \*   
## Hits 1.308e-02 4.622e-03 2.831 0.00503 \*\*   
## HmRun 1.179e-02 1.205e-02 0.978 0.32889   
## Runs -1.419e-03 5.794e-03 -0.245 0.80670   
## RBI -1.675e-03 5.056e-03 -0.331 0.74063   
## Walks 1.096e-02 3.554e-03 3.082 0.00229 \*\*   
## Years 5.696e-02 2.413e-02 2.361 0.01902 \*   
## CAtBat 1.283e-04 2.629e-04 0.488 0.62596   
## CHits -4.414e-04 1.311e-03 -0.337 0.73670   
## CHmRun -7.809e-05 3.144e-03 -0.025 0.98020   
## CRuns 1.513e-03 1.459e-03 1.037 0.30072   
## CRBI 1.312e-04 1.346e-03 0.097 0.92246   
## CWalks -1.466e-03 6.377e-04 -2.298 0.02239 \*   
## LeagueN 2.825e-01 1.541e-01 1.833 0.06797 .   
## DivisionW -1.656e-01 7.847e-02 -2.111 0.03580 \*   
## PutOuts 3.389e-04 1.505e-04 2.251 0.02526 \*   
## Assists 6.214e-04 4.300e-04 1.445 0.14970   
## Errors -1.197e-02 8.537e-03 -1.402 0.16225   
## NewLeagueN -1.742e-01 1.536e-01 -1.134 0.25788   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6135 on 243 degrees of freedom  
## Multiple R-squared: 0.5586, Adjusted R-squared: 0.524   
## F-statistic: 16.18 on 19 and 243 DF, p-value: < 2.2e-16

Subset\_selection <- regsubsets(Hitters1.df$Salary ~ ., data = Hitters1.df, nbest = 1, nvmax = dim(Hitters1.df)[2],  
 method = "exhaustive")  
  
sum <- summary(Subset\_selection)  
  
sum$which

## (Intercept) AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits  
## 1 TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## 2 TRUE FALSE TRUE FALSE FALSE FALSE FALSE FALSE TRUE FALSE  
## 3 TRUE FALSE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 4 TRUE TRUE TRUE FALSE FALSE FALSE TRUE FALSE TRUE FALSE  
## 5 TRUE FALSE TRUE FALSE FALSE FALSE TRUE TRUE FALSE TRUE  
## 6 TRUE TRUE TRUE FALSE FALSE FALSE TRUE TRUE FALSE TRUE  
## 7 TRUE TRUE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 8 TRUE TRUE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 9 TRUE TRUE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 10 TRUE TRUE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 11 TRUE TRUE TRUE TRUE FALSE FALSE TRUE TRUE FALSE FALSE  
## 12 TRUE TRUE TRUE TRUE FALSE FALSE TRUE TRUE FALSE FALSE  
## 13 TRUE TRUE TRUE TRUE FALSE FALSE TRUE TRUE FALSE FALSE  
## 14 TRUE TRUE TRUE TRUE FALSE FALSE TRUE TRUE TRUE FALSE  
## 15 TRUE TRUE TRUE TRUE FALSE FALSE TRUE TRUE TRUE TRUE  
## 16 TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE TRUE TRUE  
## 17 TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## 18 TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## 19 TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## CHmRun CRuns CRBI CWalks LeagueN DivisionW PutOuts Assists Errors  
## 1 FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## 2 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## 3 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## 4 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## 5 FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE  
## 6 FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE  
## 7 FALSE TRUE FALSE TRUE FALSE FALSE TRUE FALSE FALSE  
## 8 FALSE TRUE FALSE TRUE FALSE TRUE TRUE FALSE FALSE  
## 9 FALSE TRUE FALSE TRUE TRUE TRUE TRUE FALSE FALSE  
## 10 FALSE TRUE FALSE TRUE TRUE TRUE TRUE FALSE FALSE  
## 11 FALSE TRUE FALSE TRUE TRUE TRUE TRUE FALSE FALSE  
## 12 FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE  
## 13 FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE  
## 14 FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE  
## 15 FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE  
## 16 FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE  
## 17 FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE  
## 18 FALSE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## 19 TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
## NewLeagueN  
## 1 FALSE  
## 2 FALSE  
## 3 FALSE  
## 4 FALSE  
## 5 FALSE  
## 6 FALSE  
## 7 FALSE  
## 8 FALSE  
## 9 FALSE  
## 10 TRUE  
## 11 TRUE  
## 12 FALSE  
## 13 TRUE  
## 14 TRUE  
## 15 TRUE  
## 16 TRUE  
## 17 TRUE  
## 18 TRUE  
## 19 TRUE

sum$rsq

## [1] 0.3857520 0.4822942 0.4986075 0.5090077 0.5190638 0.5270507 0.5355590  
## [8] 0.5436891 0.5473898 0.5501579 0.5524819 0.5552470 0.5577193 0.5579177  
## [15] 0.5582361 0.5583376 0.5584807 0.5585572 0.5585583

sum$adjr2

## [1] 0.3833985 0.4783118 0.4927999 0.5013954 0.5097071 0.5159660 0.5228097  
## [8] 0.5293171 0.5312890 0.5323071 0.5328696 0.5338989 0.5346284 0.5329615  
## [15] 0.5314083 0.5296116 0.5278447 0.5259917 0.5240423

sum$cp

## [1] 79.124523 27.981036 21.001033 17.276086 13.740484 11.343922 8.660386  
## [8] 6.185015 6.147901 6.624133 7.344849 7.822749 8.461825 10.352605  
## [15] 12.177361 14.121460 16.042709 18.000617 20.000000

Subset\_selection\_lm <- lm(Salary ~ (AtBat+Hits+Walks+Years+CRuns+CWalks+PutOuts), data = Hitters1.df)  
  
BIC(linear\_regression)

## [1] 585.5431

BIC(Subset\_selection\_lm)

## [1] 532.0347

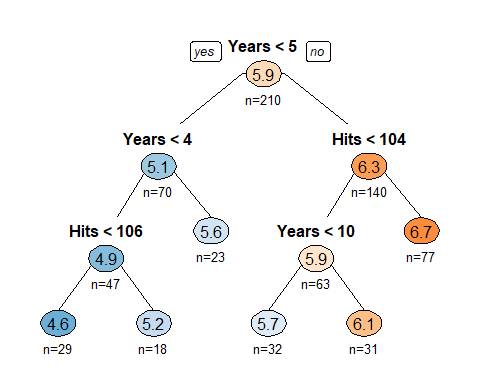
# *Interpretation 4. The best model from BIC is linear regression obtained through exhaustive search because on comparing the BIC values between subset selection(BIC = 532.03) and linear regression(BIC = 585.54), we can see that the BIC value is low for exhaustive search. Also the 7 predictor variables included from the best model are CRuns, Hits, Cwalks, Walks, Putouts, Years, AtBat.(Note : Based on the Cp value we decide to go with 7 predictors from the exhaustive search as 7th Cp = 8.6 which is closer to p+1)*

#### Question 5: Now create a training data set consisting of 80 percent of the observations, and a test data set consisting of the remaining observations.

set.seed(42)  
train\_index <- sample(c(1:263),(0.8\*263))  
Hitters1.train.df <- Hitters1.df[train\_index,]  
Hitters1.valid.df <- Hitters1.df[-train\_index,]

#### Question 6:Generate a regression tree of log Salary using only Years and Hits variables from the training data set. Which players are likely to receive highest salaries according to this model? Write down the rule and elaborate on it.

set.seed(42)  
salary\_regtree <- rpart(Salary ~ (Hits+Years), data = Hitters1.train.df, method = "anova")  
prp(salary\_regtree,type = 1, extra = 1, under = TRUE, split.font = 2,   
 varlen = -10, box.palette = "BuOr")



rpart.rules(salary\_regtree, cover = TRUE)

## Salary cover  
## 4.6 when Years < 4 & Hits < 106 14%  
## 5.2 when Years < 4 & Hits >= 106 9%  
## 5.6 when Years is 4 to 5 11%  
## 5.7 when Years is 5 to 10 & Hits < 104 15%  
## 6.1 when Years >= 10 & Hits < 104 15%  
## 6.7 when Years >= 5 & Hits >= 104 37%

# *Interpretation 6. The players who are likey to receive highest salaries are those whose satisfy this rule:*

# *The players whose years are >= 5 and whose hits are >= 118 receive the highest salary. These players would have their log(salary) = 6.7 which means their salary is around 900 dollars*

# *The rule indicates that the player having his number of years in the major leagues greater than or equal to 5 and whose number of hits in the year 1986 is greater than or equal to 118 are the players whose salary will be the highest*

#### Question 7a:Now create a regression tree using all the variables in the training data set. Perform boosting on the training set with 1,000 trees for a range of values of the shrinkage parameter λ.

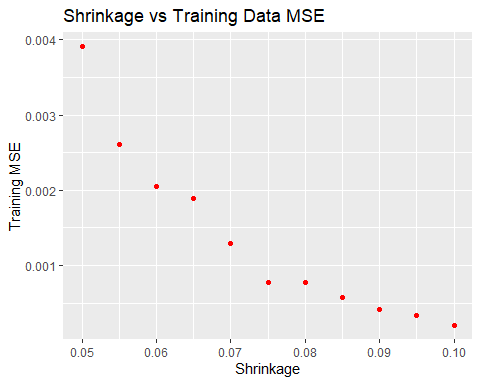
set.seed(42)  
shrinkage1 <- seq(0.05,0.1,0.005)  
training.mse <- array(NA,length(shrinkage1))  
test.mse <- array(NA,length(shrinkage1))  
  
for (i in 1:length(shrinkage1))  
{  
sal.boost <- gbm(Salary ~., data = Hitters1.train.df,distribution ="gaussian",n.trees = 1000, shrinkage = shrinkage1[i],interaction.depth = 4)  
   
pred.train = predict(sal.boost, Hitters1.train.df, n.trees = 1000)  
training.mse[i] = mean((pred.train - Hitters1.train.df$Salary)^2)  
  
pred.test = predict(sal.boost, Hitters1.train.df, n.trees = 1000)  
test.mse[i] = mean((pred.test - Hitters1.valid.df$Salary)^2)  
  
  
  
}

#### Question 7b:Produce a plot with different shrinkage values on the x-axis and the corresponding training set MSE on the y-axis.

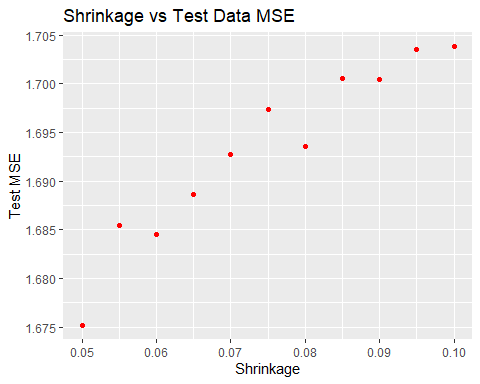
#### Question 8:Produce a plot with different shrinkage values on the x-axis and the corresponding test set MSE on the y-axis.

### Solution 7b and 8:

ggplot(data.frame(x=shrinkage1,y=training.mse))+  
 geom\_point(aes(x=x,y=y),colour = "red")+  
 xlab("Shrinkage")+  
 ylab("Training MSE")+  
 ggtitle("Shrinkage vs Training Data MSE")

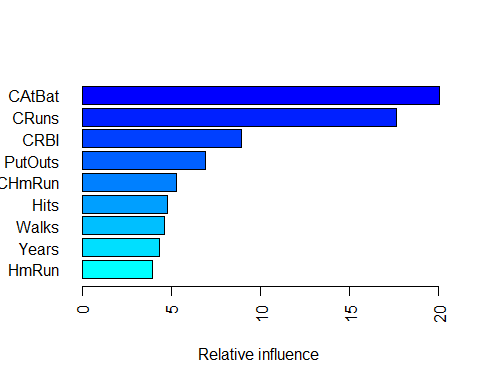


ggplot(data.frame(x=shrinkage1,y=test.mse))+  
 geom\_point(aes(x=x,y=y),colour = "red")+  
 xlab("Shrinkage")+  
 ylab("Test MSE")+  
 ggtitle("Shrinkage vs Test Data MSE")



#### Question 9:Which variables appear to be the most important predictors in the boosted model?

summary(sal.boost, cBars = 9,las = 2)



## var rel.inf  
## CAtBat CAtBat 20.0454182  
## CRuns CRuns 17.6228106  
## CRBI CRBI 8.9428663  
## PutOuts PutOuts 6.9132304  
## CHmRun CHmRun 5.2837183  
## Hits Hits 4.8020514  
## Walks Walks 4.5866501  
## Years Years 4.3508594  
## HmRun HmRun 3.9563671  
## CWalks CWalks 3.7584287  
## AtBat AtBat 3.5287168  
## Errors Errors 3.1917524  
## RBI RBI 3.1645594  
## Assists Assists 2.9426524  
## Runs Runs 2.6571421  
## CHits CHits 2.5012528  
## League League 0.6974682  
## Division Division 0.6702535  
## NewLeague NewLeague 0.3838018

# *Interpretation 9:From the above graph we can see that the following predictors have the highest relative influence on the output:CRuns, CHits, CRBI, CAtBat, Walks, CHmRun, Hits*

#### Question 10:Now apply bagging to the training set. What is the test set MSE for this approach?

bag.salary <- randomForest(Salary~., data=Hitters1.train.df,   
 mtry = 19, importance = TRUE)   
bag.salary

##   
## Call:  
## randomForest(formula = Salary ~ ., data = Hitters1.train.df, mtry = 19, importance = TRUE)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 19  
##   
## Mean of squared residuals: 0.2056712  
## % Var explained: 72.84

yhat.bag <- predict(bag.salary, newdata=Hitters1.valid.df)  
mean((yhat.bag-Hitters1.valid.df$Salary)^2)

## [1] 0.2498306

# *Interpretation 10. The MSE for the test = 0.199*