BUAN6356_Homework4_UdayakumarA

Anjana

4/20/2021

```
library(ISLR)
```

```
## Warning: package ’ISLR’ was built under R version 4.0.5
```

```
library(ggplot2)
library(leaps)
```

```
## Warning: package ’leaps’ was built under R version 4.0.4
```

```
library(rpart)
```

```
## Warning: package ’rpart’ was built under R version 4.0.4
```

```
library(caret)
```

```
## Warning: package ’caret’ was built under R version 4.0.4

## Loading required package: lattice

## Warning: package ’lattice’ was built under R version 4.0.4
```

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

```
## Warning: package ’rpart.plot’ was built under R version 4.0.4
```

```
library(tree)
```

```
## Warning: package ’tree’ was built under R version 4.0.5
```

```
library(gbm)
```

```
## Warning: package ’gbm’ was built under R version 4.0.5

## Loaded gbm 2.1.8
```

```r
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.0.5

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```r
library(tinytex)
tinytex::tlmgr_install("pdfcrop")
```

```
## tlmgr install pdfcrop

## tlmgr update --self

## A new version of TeX Live has been released. If you need to install or update any LaTeX packages, yo

## tlmgr install pdfcrop
```

```r
Sys.setenv(R_GSCMD="C:/Program Files/gs/gs9.53.3/bin/gswin32c.exe")
```

```r
#Question 1
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 ...
```

```
is.na(Hitters$Salary)
```

```
##   [1]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [13] FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE
##  [25] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE
##  [37]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE
##  [49]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
##  [61] FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE
##  [73] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE
##  [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
##  [97] FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE
## [109] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [121] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
## [133] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [145]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [157] FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [169] FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
## [181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [193] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
## [205] FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
## [229]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
## [241] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE
## [253] FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [265] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## [277] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
## [289] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
## [301] FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
## [313] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
```

```
data <-Hitters
data.nona<- data[complete.cases(data[,19]),]
str(data.nona)
```

```
## 'data.frame':    263 obs. of  20 variables:
##  $ AtBat    : int  315 479 496 321 594 185 298 323 401 574 ...
##  $ Hits     : int  81 130 141 87 169 37 73 81 92 159 ...
##  $ HmRun    : int  7 18 20 10 4 1 0 6 17 21 ...
##  $ Runs     : int  24 66 65 39 74 23 24 26 49 107 ...
##  $ RBI      : int  38 72 78 42 51 8 24 32 66 75 ...
##  $ Walks    : int  39 76 37 30 35 21 7 8 65 59 ...
##  $ Years    : int  14 3 11 2 11 2 3 2 13 10 ...
##  $ CAtBat   : int  3449 1624 5628 396 4408 214 509 341 5206 4631 ...
##  $ CHits    : int  835 457 1575 101 1133 42 108 86 1332 1300 ...
##  $ CHmRun   : int  69 63 225 12 19 1 0 6 253 90 ...
##  $ CRuns    : int  321 224 828 48 501 30 41 32 784 702 ...
##  $ CRBI     : int  414 266 838 46 336 9 37 34 890 504 ...
##  $ CWalks   : int  375 263 354 33 194 24 12 8 866 488 ...
##  $ League   : Factor w/ 2 levels "A","N": 2 1 2 2 1 2 1 2 1 1 ...
##  $ Division : Factor w/ 2 levels "E","W": 2 2 1 1 2 1 2 2 1 1 ...
##  $ PutOuts  : int  632 880 200 805 282 76 121 143 0 238 ...
##  $ Assists  : int  43 82 11 40 421 127 283 290 0 445 ...
##  $ Errors   : int  10 14 3 4 25 7 9 19 0 22 ...
##  $ Salary   : num  475 480 500 91.5 750 ...
##  $ NewLeague: Factor w/ 2 levels "A","N": 2 1 2 2 1 1 1 2 1 1 ...
```

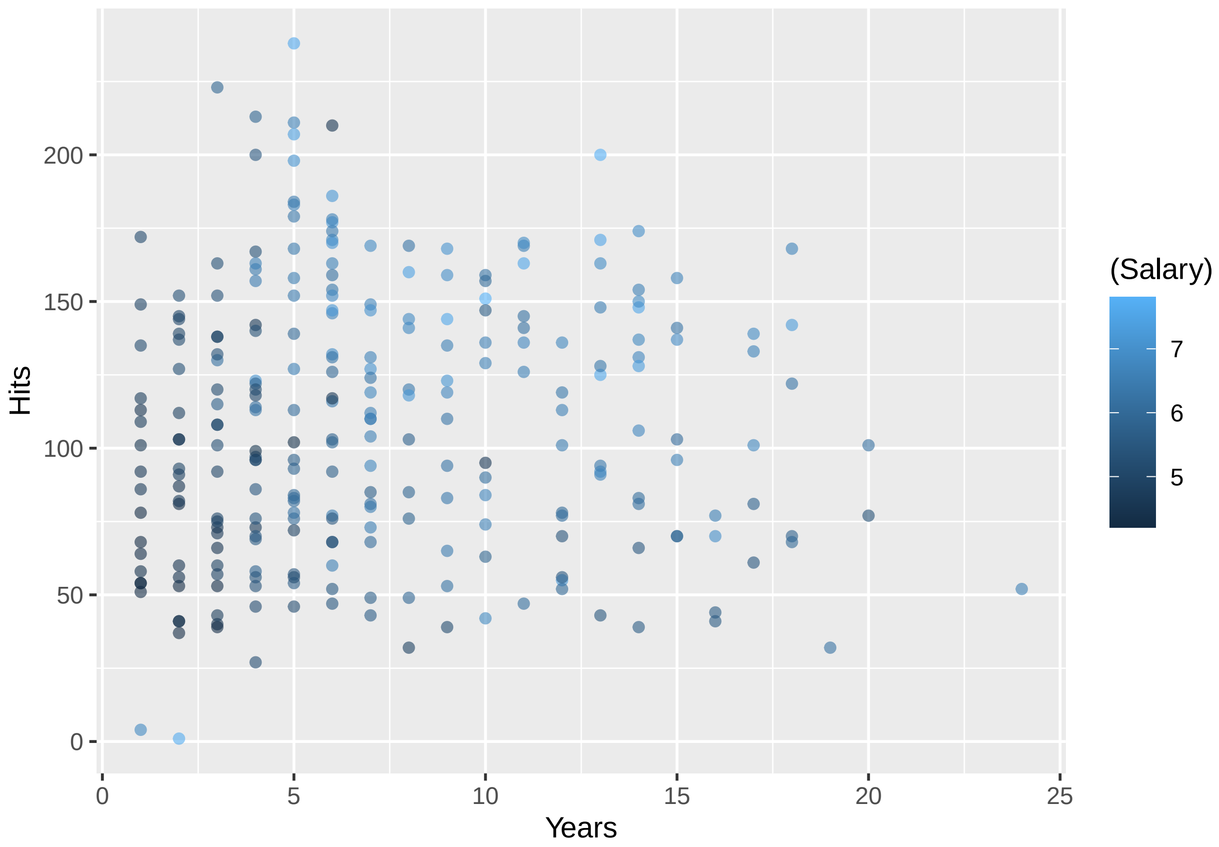
1)59 Observations were removed by removing the observations with no salary record

```
#Question:2
data.nona$Salary <- log(data.nona$Salary)
str(data.nona)
```

```
## 'data.frame':    263 obs. of  20 variables:
##  $ AtBat    : int  315 479 496 321 594 185 298 323 401 574 ...
##  $ Hits     : int  81 130 141 87 169 37 73 81 92 159 ...
##  $ HmRun    : int  7 18 20 10 4 1 0 6 17 21 ...
##  $ Runs     : int  24 66 65 39 74 23 24 26 49 107 ...
##  $ RBI      : int  38 72 78 42 51 8 24 32 66 75 ...
##  $ Walks    : int  39 76 37 30 35 21 7 8 65 59 ...
##  $ Years    : int  14 3 11 2 11 2 3 2 13 10 ...
##  $ CAtBat   : int  3449 1624 5628 396 4408 214 509 341 5206 4631 ...
##  $ CHits    : int  835 457 1575 101 1133 42 108 86 1332 1300 ...
##  $ CHmRun   : int  69 63 225 12 19 1 0 6 253 90 ...
##  $ CRuns    : int  321 224 828 48 501 30 41 32 784 702 ...
##  $ CRBI     : int  414 266 838 46 336 9 37 34 890 504 ...
##  $ CWalks   : int  375 263 354 33 194 24 12 8 866 488 ...
##  $ League   : Factor w/ 2 levels "A","N": 2 1 2 2 1 2 1 2 1 1 ...
##  $ Division : Factor w/ 2 levels "E","W": 2 2 1 1 2 1 2 2 1 1 ...
##  $ PutOuts  : int  632 880 200 805 282 76 121 143 0 238 ...
##  $ Assists  : int  43 82 11 40 421 127 283 290 0 445 ...
##  $ Errors   : int  10 14 3 4 25 7 9 19 0 22 ...
##  $ Salary   : num  6.16 6.17 6.21 4.52 6.62 ...
##  $ NewLeague: Factor w/ 2 levels "A","N": 2 1 2 2 1 1 1 2 1 1 ...
```

2)Logarithmic transformations are carried out to normalize a highly skewed data variable.

```
#Question 3
scatter_plot <- ggplot(data.nona,aes(y=Hits,x=Years,color =(Salary)))+geom_point(alpha =0.6)
scatter_plot
```



3)From the plot we notice that the log salaries become higher as the number of years increase.

```
#Question 4
hitters.lm <- lm(Salary~.,data=data.nona)
summary(hitters.lm)
```

```
##
## Call:
## lm(formula = Salary ~ ., data = data.nona)
##
## 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
```

```
search <- regsubsets(Salary~.,data=data.nona,nbest=1,
                     nvmax=dim(data.nona)[2],method="exhaustive")
sum <- summary(search)
sum$which
```

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

```
##  [1] -117.0304 -156.4291 -159.2777 -159.2182 -159.0885 -157.9207 -157.1229
##  [8] -156.1954 -152.7649 -148.8061 -144.5962 -140.6541 -136.5480 -131.0939
## [15] -125.7112 -120.1995 -114.7125 -109.1859 -103.6145
```

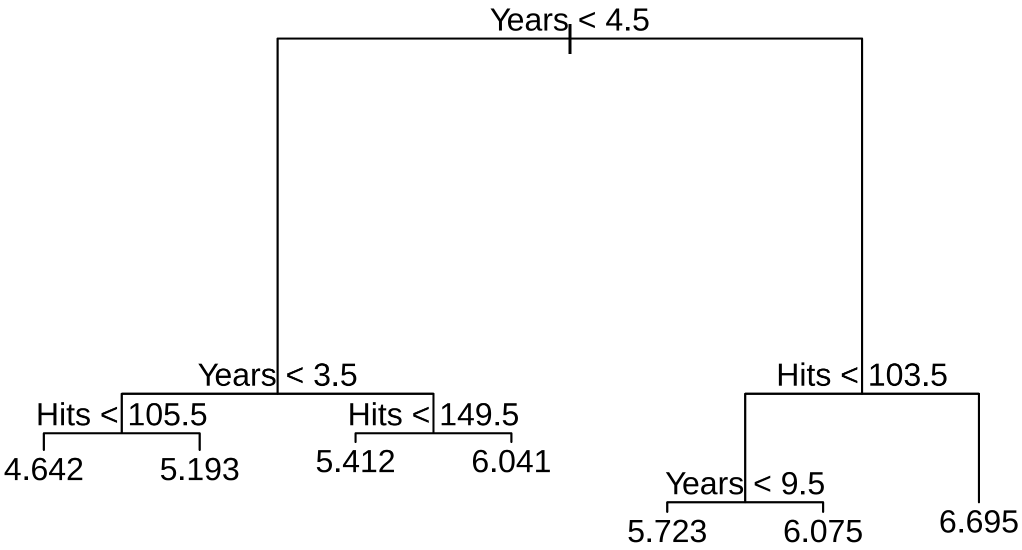
4)The 3rd model gives us the lowest BIC hence that is considered to be the best subset.The predictor variables included in the best model are Hits,Walks and years.

```
#Question 5
set.seed(42)
train.index <- sample(c(1:263),210)
train.df <- data.nona[train.index,]
valid.df <- data.nona[-train.index,]
```

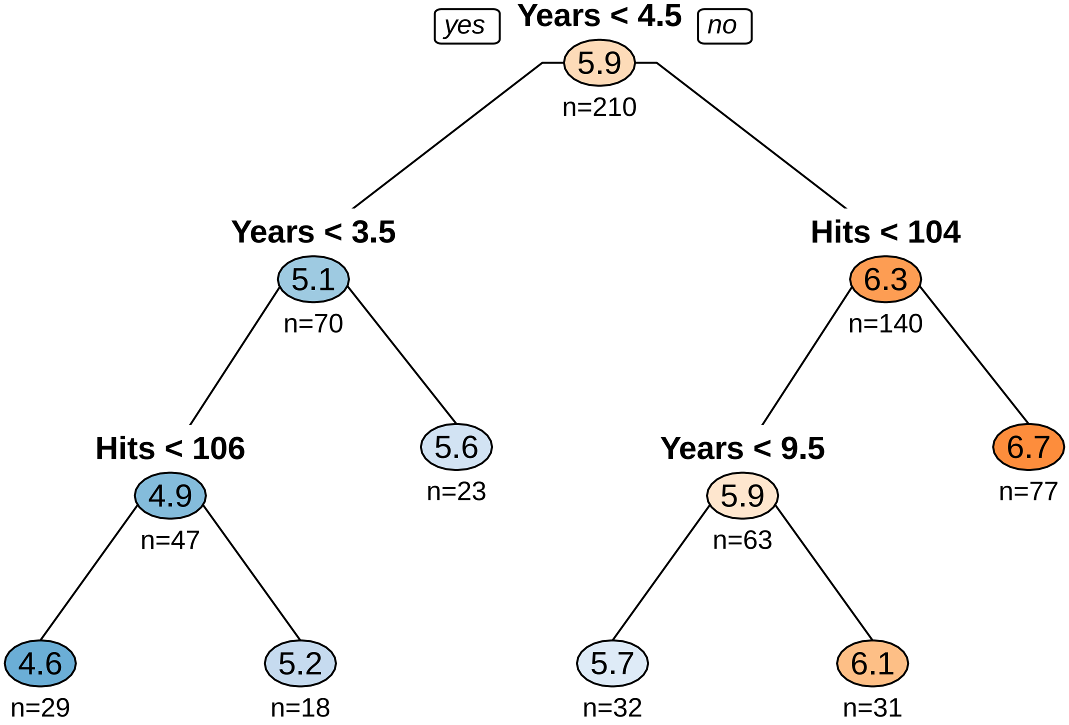
```
#Question 6
#using tree package
tree.hitters <- tree(Salary~Hits+Years,data.nona,subset = train.index)
summary(tree.hitters)
```

```
##
## Regression tree:
## tree(formula = Salary ~ Hits + Years, data = data.nona, subset = train.index)
## Number of terminal nodes:  7
## Residual mean deviance:  0.2436 = 49.45 / 203
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -2.19500 -0.29810 -0.03641  0.00000  0.22790  2.18200
```

```r
plot(tree.hitters)
text(tree.hitters,pretty = 0)
```



```r
#using rpart package
reg_tree <- rpart(Salary~Hits+Years,data = train.df,method ="anova")
prp(reg_tree, type = 1, extra = 1, under = TRUE, roundint = FALSE,
    split.font = 2, varlen = -10, box.palette = "BuOr")
```

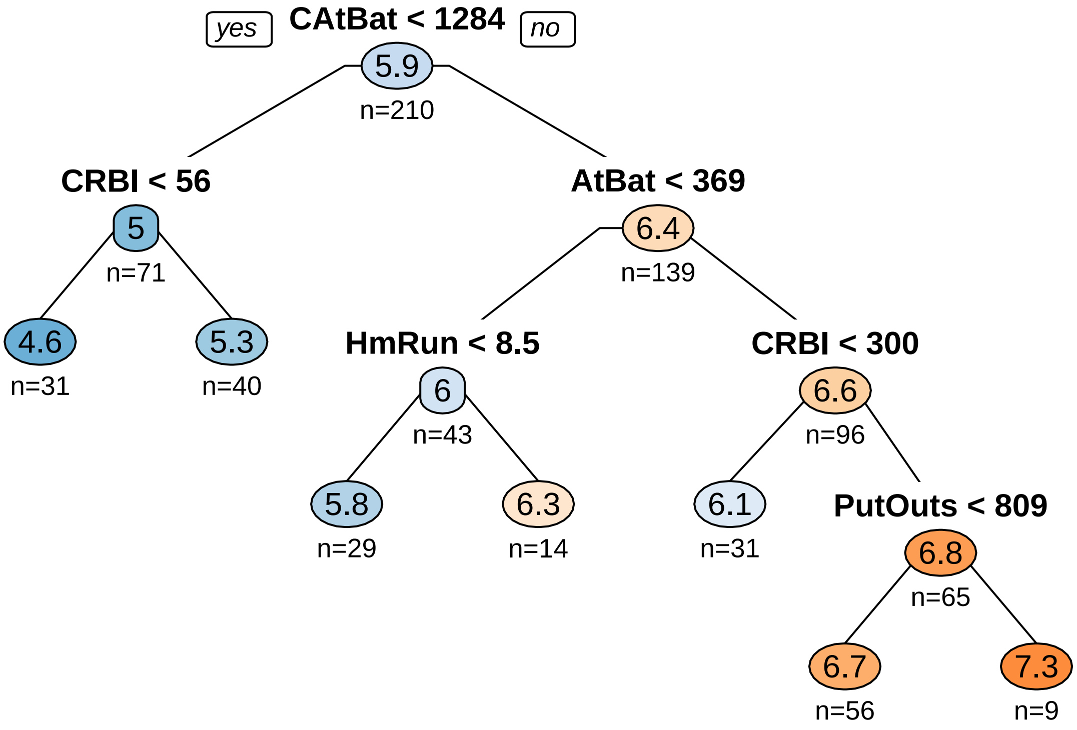


```r
rpart.rules(reg_tree, cover = TRUE)
```

```
##  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%
```

6)When the player has more than or equal to 4.5 years of experience and hits more than or equal to 104 he gets high salary

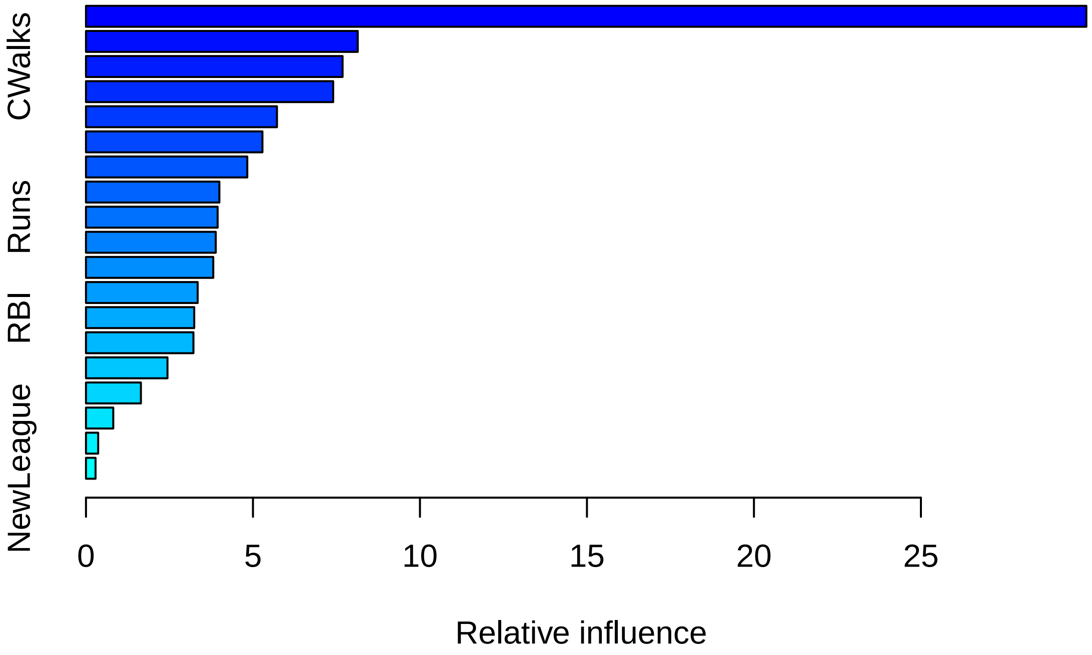
```r
#Question 7
reg_tree_all <- rpart(Salary~.,data=train.df,method="anova")
prp(reg_tree_all, type = 1, extra = 1, under = TRUE, roundint = FALSE,
    split.font = 2, varlen = -10, box.palette = "BuOr")
```



```
rpart.rules(reg_tree_all, cover = TRUE)
```

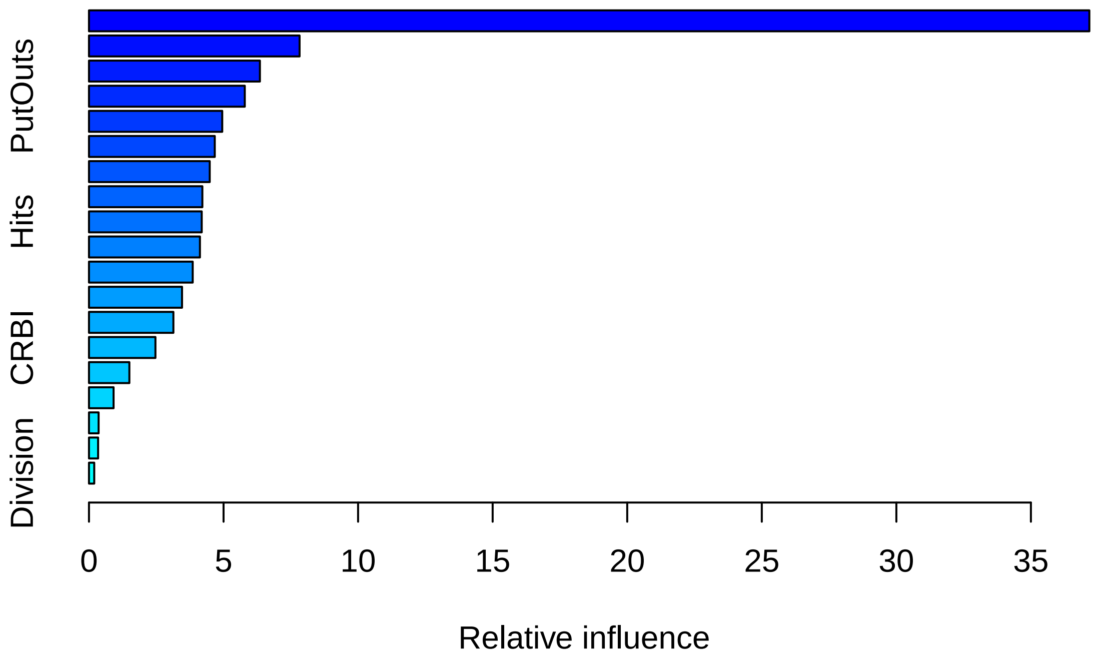
```
##  Salary                                                                                  cover
##     4.6 when CAtBat <  1284 & CRBI <   56                                                  15%
##     5.3 when CAtBat <  1284 & CRBI >=  56                                                  19%
##     5.8 when CAtBat >= 1284               & AtBat <  369 & HmRun <  9                      14%
##     6.1 when CAtBat >= 1284 & CRBI <  300 & AtBat >= 369                                   15%
##     6.3 when CAtBat >= 1284               & AtBat <  369 & HmRun >= 9                       7%
##     6.7 when CAtBat >= 1284 & CRBI >= 300 & AtBat >= 369              & PutOuts <  809     27%
##     7.3 when CAtBat >= 1284 & CRBI >= 300 & AtBat >= 369              & PutOuts >= 809      4%
```

```
boost.hitters1 <- gbm(Salary~.,data = train.df,distribution = "gaussian",
                      shrinkage = 0.2,n.trees = 1000,interaction.depth = 4)
summary(boost.hitters1)
```



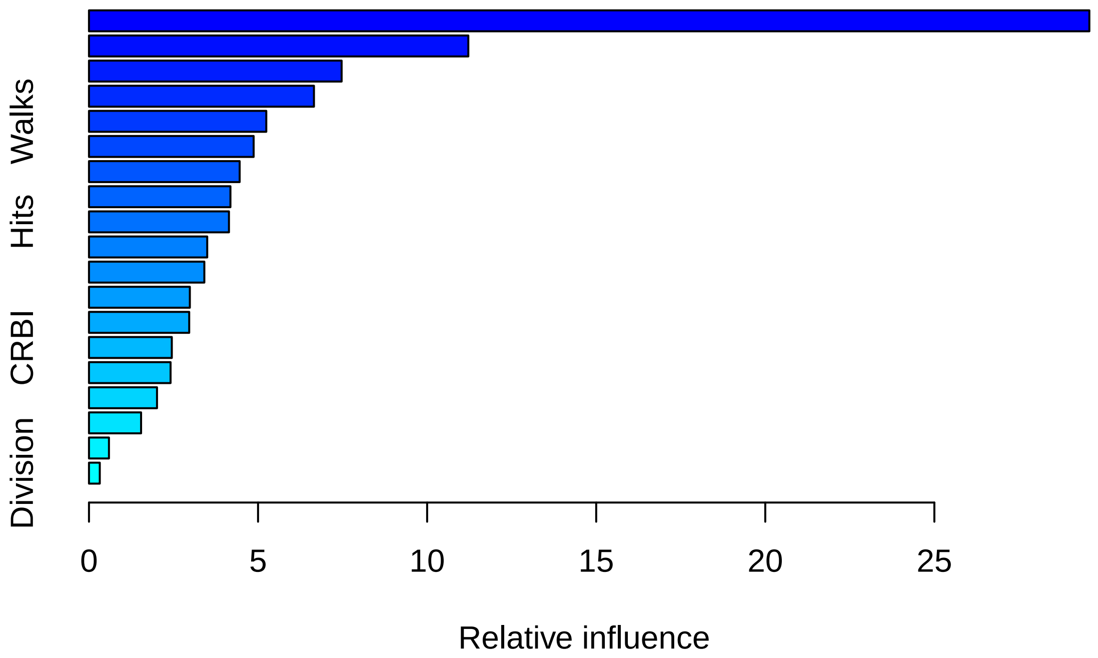
```
##                 var    rel.inf
## CAtBat       CAtBat 29.9536299
## CRBI           CRBI  8.1380122
## CWalks       CWalks  7.6828786
## CHmRun       CHmRun  7.4019172
## AtBat         AtBat  5.7189766
## PutOuts     PutOuts  5.2836204
## CRuns         CRuns  4.8299937
## HmRun         HmRun  3.9941825
## Runs           Runs  3.9418922
## Walks         Walks  3.8863946
## Years         Years  3.8109649
## Assists     Assists  3.3449673
## RBI             RBI  3.2407968
## Errors       Errors  3.2176918
## Hits           Hits  2.4405844
## CHits         CHits  1.6446300
## League       League  0.8171153
## Division   Division  0.3647234
## NewLeague NewLeague  0.2870281
```

```
boost.hitters2 <- gbm(Salary~.,data = train.df,distribution = "gaussian",
                      shrinkage = 0.4,n.trees = 1000,interaction.depth = 4)
summary(boost.hitters2)
```



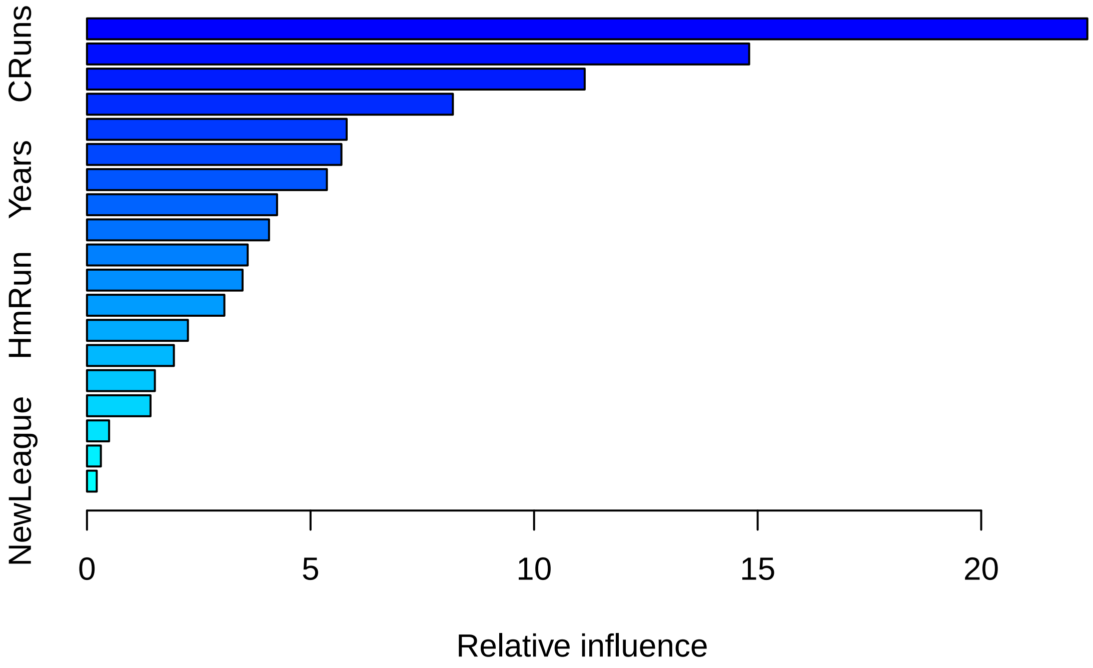
```
##                 var    rel.inf
## CAtBat       CAtBat 37.1717184
## CHmRun       CHmRun  7.8251839
## CRuns         CRuns  6.3515200
## PutOuts     PutOuts  5.7890263
## HmRun         HmRun  4.9519234
## Walks         Walks  4.6738234
## AtBat         AtBat  4.4869908
## Assists     Assists  4.2131627
## Hits           Hits  4.1890760
## Years         Years  4.1225387
## CWalks       CWalks  3.8541090
## RBI             RBI  3.4578867
## Errors       Errors  3.1360647
## CRBI           CRBI  2.4697327
## Runs           Runs  1.5006629
## CHits         CHits  0.9142512
## NewLeague NewLeague  0.3581985
## League       League  0.3386043
## Division   Division  0.1955265
```

```
boost.hitters3 <- gbm(Salary~.,data = train.df,distribution = "gaussian",
                      shrinkage = 0.6,n.trees = 1000,interaction.depth = 4)
summary(boost.hitters3)
```



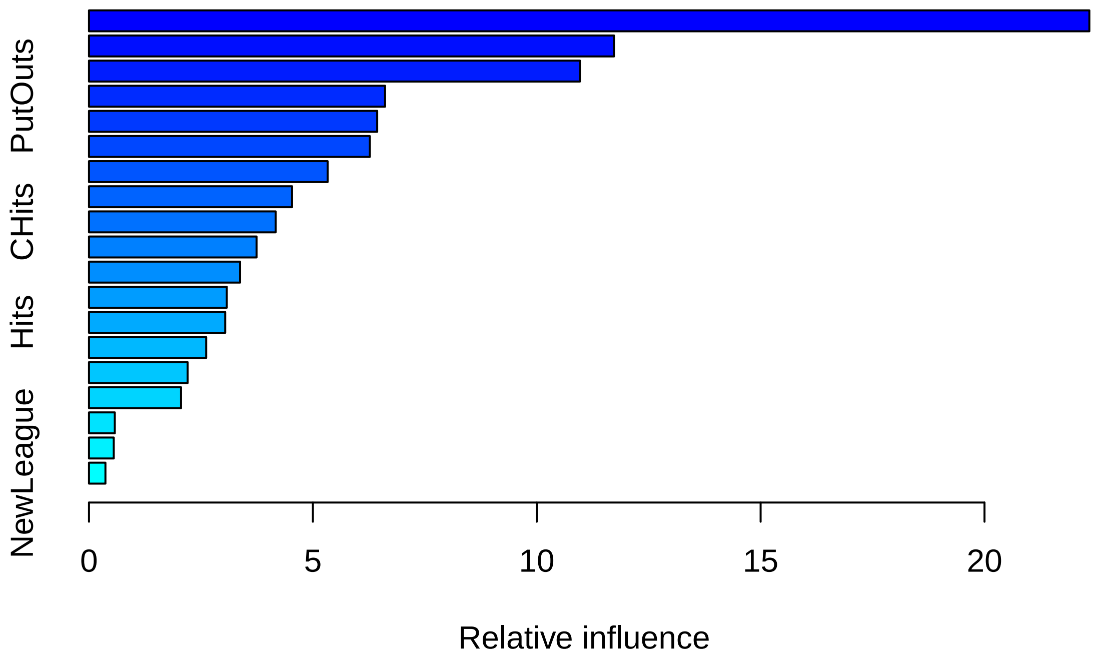
```
##                   var    rel.inf
## CRuns           CRuns 29.5840844
## CAtBat         CAtBat 11.2183324
## AtBat           AtBat  7.4711804
## PutOuts       PutOuts  6.6536370
## Walks           Walks  5.2436526
## Errors         Errors  4.8691959
## Assists       Assists  4.4561914
## RBI               RBI  4.1829045
## Hits             Hits  4.1401195
## CHmRun         CHmRun  3.4965370
## CWalks         CWalks  3.4114881
## Runs             Runs  2.9831977
## HmRun           HmRun  2.9639143
## CRBI             CRBI  2.4498169
## Years           Years  2.4129578
## CHits           CHits  2.0116112
## League         League  1.5407362
## NewLeague NewLeague  0.5908351
## Division     Division  0.3196076
```

```
boost.hitters4 <- gbm(Salary~.,data = train.df,distribution = "gaussian",
                      shrinkage = 0.01,n.trees = 1000,interaction.depth = 4)
summary(boost.hitters4)
```



```
##                 var    rel.inf
## CAtBat       CAtBat 22.3739580
## CRuns         CRuns 14.8105227
## CRBI           CRBI 11.1319227
## CWalks       CWalks  8.1834583
## CHits         CHits  5.8079736
## PutOuts     PutOuts  5.6910572
## Years         Years  5.3653720
## CHmRun       CHmRun  4.2515434
## AtBat         AtBat  4.0719784
## Walks         Walks  3.5942114
## Hits           Hits  3.4796736
## HmRun         HmRun  3.0730736
## Errors       Errors  2.2579230
## RBI             RBI  1.9438171
## Assists     Assists  1.5182732
## Runs           Runs  1.4215855
## League       League  0.4947809
## Division   Division  0.3106599
## NewLeague NewLeague  0.2182155
```

```
boost.hitters5 <- gbm(Salary~.,data = train.df,distribution = "gaussian",
                      shrinkage = 0.02,n.trees = 1000,interaction.depth = 4)
summary(boost.hitters5)
```

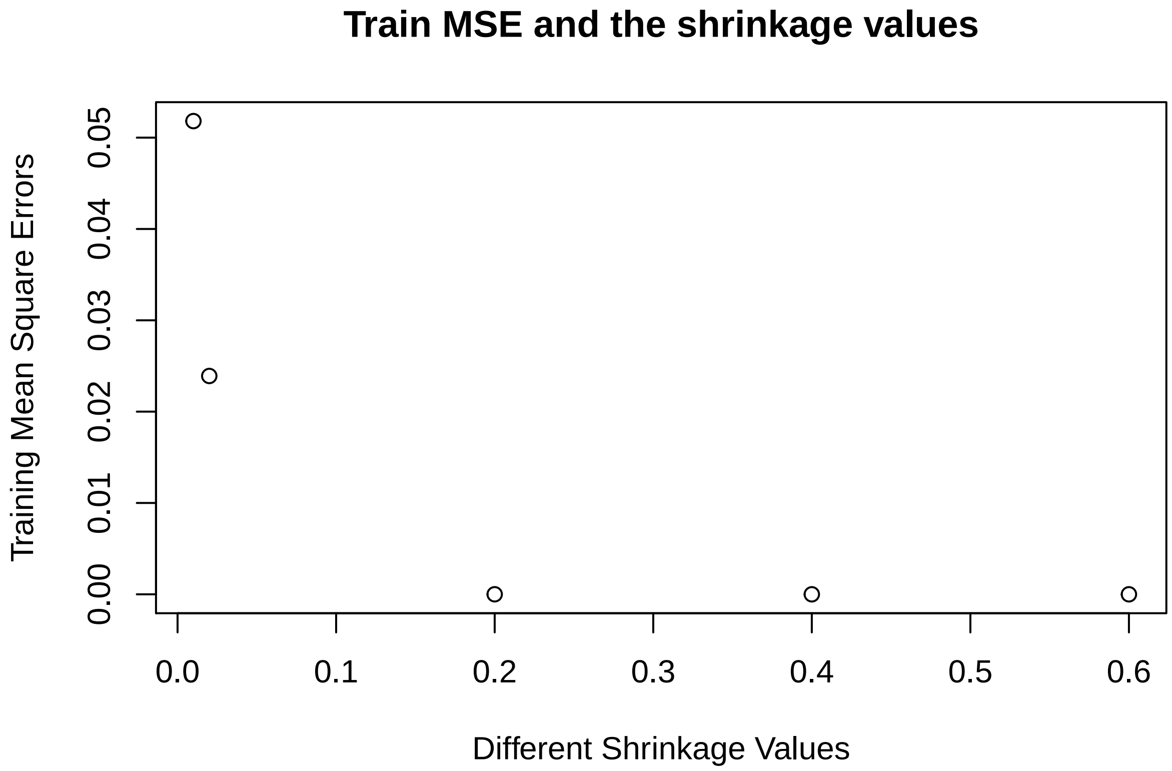


```
##                 var    rel.inf
## CAtBat       CAtBat 22.3415257
## CRuns         CRuns 11.7261489
## CRBI           CRBI 10.9647167
## PutOuts     PutOuts  6.6121951
## CWalks       CWalks  6.4370365
## CHmRun       CHmRun  6.2708049
## Years         Years  5.3295279
## AtBat         AtBat  4.5360799
## CHits         CHits  4.1686028
## Walks         Walks  3.7420610
## HmRun         HmRun  3.3753418
## RBI             RBI  3.0776240
## Hits           Hits  3.0428474
## Errors       Errors  2.6181498
## Assists     Assists  2.2027160
## Runs           Runs  2.0564324
## League       League  0.5769805
## Division   Division  0.5533210
## NewLeague NewLeague  0.3678877
```

```
MSE_train <- c(boost.hitters1$train.error[1000],boost.hitters2$train.error[1000]
              ,boost.hitters3$train.error[1000],boost.hitters4$train.error[1000]
              ,boost.hitters5$train.error[1000])
MSE_train
```

```
## [1] 2.492576e-06 1.541333e-10 2.567661e-13 5.181151e-02 2.390377e-02
```

```
Shrinkage_values <- c(0.2,0.4,0.6,0.01,0.02)
plot(Shrinkage_values,MSE_train,xlab = "Different Shrinkage Values",ylab =
       "Training Mean Square Errors",main = "Train MSE and the shrinkage values"
     )
```



```
#Question 8
Shrinkage_values <- c(0.2,0.4,0.6,0.01,0.02)
hitter.test <- data.nona[-train.index,"Salary"]
yhat.boost1 <- predict(boost.hitters1,newdata = valid.df,n.trees = 1000)
a <- mean((yhat.boost1-hitter.test)^2)
a
```

```
## [1] 0.3829805
```

```
yhat.boost2 <- predict(boost.hitters2,newdata = valid.df,n.trees = 1000)
b <- mean((yhat.boost2-hitter.test)^2)
b
```

```
## [1] 0.4186236
```

```
yhat.boost3 <- predict(boost.hitters3,newdata = valid.df,n.trees = 1000)
c <- mean((yhat.boost3-hitter.test)^2)
c
```

```
## [1] 0.4649825
```

```
yhat.boost4<- predict(boost.hitters4,newdata = valid.df,n.trees = 1000)
d <- mean((yhat.boost1-hitter.test)^2)
d
```

```
## [1] 0.3829805
```

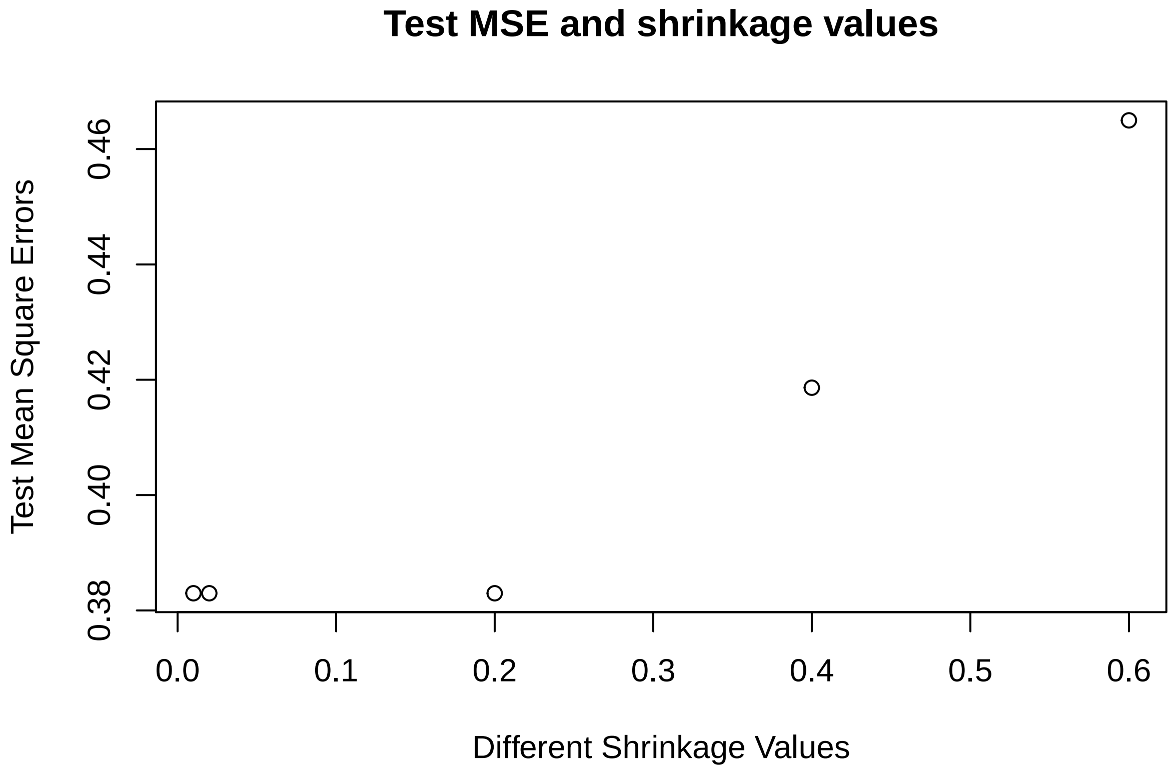
```
yhat.boost5 <- predict(boost.hitters5,newdata = valid.df,n.trees = 1000)
e <- mean((yhat.boost1-hitter.test)^2)
e
```

```
## [1] 0.3829805
```

```
MSE_test <- c(a,b,c,d,e)
MSE_test
```

```
## [1] 0.3829805 0.4186236 0.4649825 0.3829805 0.3829805
```

```
plot(Shrinkage_values,MSE_test,xlab = "Different Shrinkage Values",ylab =
       "Test Mean Square Errors",main = "Test MSE and shrinkage values")
```

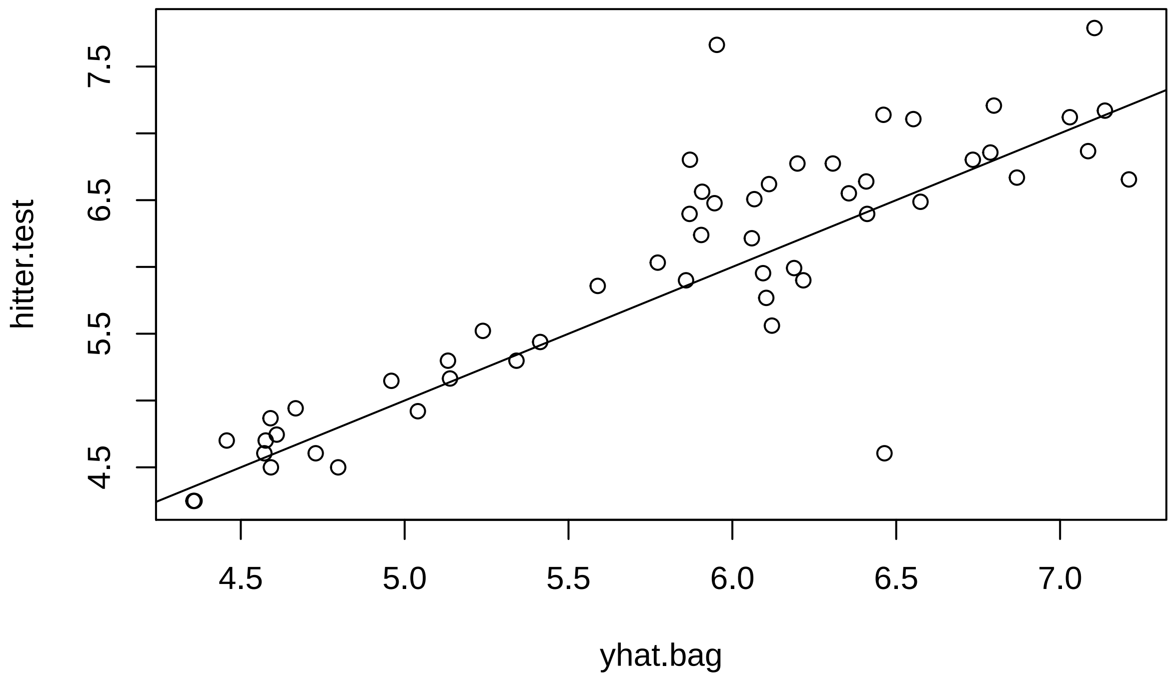


9) By altering te shrinkage parameters to different values we see that the most important predictors are CAtBat and CRuns

```
#Question 10
bag.hitters <- randomForest(Salary~., data=train.df,
                            mtry = 19, importance = TRUE)
bag.hitters
```

```
##
## Call:
##  randomForest(formula = Salary ~ ., data = 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: 0.2033149
##                     % Var explained: 73.15
```

```
yhat.bag <- predict(bag.hitters, newdata=valid.df)
plot(yhat.bag, hitter.test)
abline(0,1)
```



```
MSE_test <- mean((yhat.bag-hitter.test)^2)
MSE_test
```

```
## [1] 0.2369779
```

10)The MSE_test is 0.24

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.