Lab 5 - Part 2 - Data Science Example 2

Lauren Jensen

2023-02-16

Overview

I'm going to take the NBA Salary dataset which can be found from Kaggle. I'm doing this because by popular demand I was asked to do another example. I'm going to show GLMs.

1) Import Data and Load Packages

```
library(ggplot2)
library(plotly)
library(magrittr)
library(dplyr)
library(psych)
library(arm)
library(gridExtra)
library(lmtest)
library(effects)
library(tidyverse)
library(alr4)      # Data
library(rlang)     # Non-standard evaluation for missplot function
library(patchwork) # arranging multiple ggplots
library(GGally)    # Pairs plot
library(ggdag)     # To draw causal DAG
library(broom)     # To work with model results
library(ggthemes)
library(scales)
library(mice)
library(fastDummies)
library(ggcorrplot)
library(mlbench)
library(caret)
library(rpart) #tree model library
library(psych)
library(data.table)
library(DiagrammeR)
library(corrplot)
```

Load Data

```
nba <- read.csv("C:/Users/ljens/Desktop/UW Class/R Certification/Class 5/NBA_season1718_salary.csv",as.i

stats <- read.csv("C:/Users/ljens/Desktop/UW Class/R Certification/Class 5/Seasons_Stats.csv",as.is=TRU
```

2) Exploratory Analysis and Data Cleansing

Summary Statistics

```r
dim(nba)
```

```
## [1] 573   4
```

```r
summary(nba)
```

```
##        X          Player               Tm             season17_18
##  Min.   :  1   Length:573         Length:573         Min.   :   17224
##  1st Qu.:144   Class :character   Class :character   1st Qu.: 1312611
##  Median :287   Mode  :character   Mode  :character   Median : 2386864
##  Mean   :287                                         Mean   : 5858946
##  3rd Qu.:430                                         3rd Qu.: 7936509
##  Max.   :573                                         Max.   :34682550
```

```r
head(nba)
```

```
##   X         Player  Tm season17_18
## 1 1  Stephen Curry GSW    34682550
## 2 2   LeBron James CLE    33285709
## 3 3   Paul Millsap DEN    31269231
## 4 4 Gordon Hayward BOS    29727900
## 5 5  Blake Griffin DET    29512900
## 6 6     Kyle Lowry TOR    28703704
```

```r
dim(stats)
```

```
## [1] 24691    53
```

```r
summary(stats)
```

```
##        X              Year         Player              Pos
##  Min.   :    0   Min.   :1950   Length:24691       Length:24691
##  1st Qu.: 6172   1st Qu.:1981   Class :character   Class :character
##  Median :12345   Median :1996   Mode  :character   Mode  :character
##  Mean   :12345   Mean   :1993
##  3rd Qu.:18518   3rd Qu.:2007
##  Max.   :24690   Max.   :2017
##                  NA's   :67
##       Age             Tm                  G               GS
##  Min.   :18.00   Length:24691       Min.   : 1.00   Min.   : 0.00
##  1st Qu.:24.00   Class :character   1st Qu.:27.00   1st Qu.: 0.00
##  Median :26.00   Mode  :character   Median :58.00   Median : 8.00
##  Mean   :26.66                      Mean   :50.84   Mean   :23.59
##  3rd Qu.:29.00                      3rd Qu.:75.00   3rd Qu.:45.00
##  Max.   :44.00                      Max.   :88.00   Max.   :83.00
##  NA's   :75                         NA's   :67      NA's   :6458
##        MP            PER              TS.            X3PAr
##  Min.   :   0   Min.   :-90.60   Min.   :0.000   Min.   :0.000
##  1st Qu.: 340   1st Qu.:  9.80   1st Qu.:0.458   1st Qu.:0.005
##  Median :1053   Median : 12.70   Median :0.506   Median :0.064
##  Mean   :1210   Mean   : 12.48   Mean   :0.493   Mean   :0.159
##  3rd Qu.:1971   3rd Qu.: 15.60   3rd Qu.:0.544   3rd Qu.:0.288
##  Max.   :3882   Max.   :129.10   Max.   :1.136   Max.   :1.000
##  NA's   :553    NA's   :590      NA's   :153     NA's   :5852
```

```
##       FTr              ORB.              DRB.              TRB.
##  Min.   :0.0000   Min.   :  0.000   Min.   :  0.00   Min.   :  0.000
##  1st Qu.:0.2080   1st Qu.:  2.600   1st Qu.:  8.80   1st Qu.:  5.900
##  Median :0.2960   Median :  5.400   Median : 12.70   Median :  9.200
##  Mean   :0.3255   Mean   :  6.182   Mean   : 13.71   Mean   :  9.949
##  3rd Qu.:0.4000   3rd Qu.:  9.000   3rd Qu.: 18.10   3rd Qu.: 13.500
##  Max.   :6.0000   Max.   :100.000   Max.   :100.00   Max.   :100.000
##  NA's   :166      NA's   :3899      NA's   :3899     NA's   :3120
##       AST.             STL.              BLK.              TOV.
##  Min.   :  0.00   Min.   : 0.000   Min.   : 0.000   Min.   :  0.00
##  1st Qu.:  6.50   1st Qu.: 1.100   1st Qu.: 0.300   1st Qu.: 11.40
##  Median : 10.50   Median : 1.500   Median : 0.900   Median : 14.20
##  Mean   : 13.01   Mean   : 1.648   Mean   : 1.411   Mean   : 15.09
##  3rd Qu.: 17.60   3rd Qu.: 2.100   3rd Qu.: 1.900   3rd Qu.: 17.70
##  Max.   :100.00   Max.   :24.200   Max.   :77.800   Max.   :100.00
##  NA's   :2136     NA's   :3899     NA's   :3899     NA's   :5109
##       USG.          blanl              OWS               DWS
##  Min.   :  0.00   Mode:logical   Min.   :-5.100   Min.   :-1.000
##  1st Qu.: 15.40   NA's:24691     1st Qu.:-0.100   1st Qu.: 0.200
##  Median : 18.60                  Median : 0.400   Median : 0.800
##  Mean   : 18.91                  Mean   : 1.257   Mean   : 1.227
##  3rd Qu.: 22.20                  3rd Qu.: 1.900   3rd Qu.: 1.800
##  Max.   :100.00                  Max.   :18.300   Max.   :16.000
##  NA's   :5051                    NA's   :106      NA's   :106
##        WS             WS.48          blank2             OBPM
##  Min.   :-2.800   Min.   :-2.519   Mode:logical   Min.   :-73.800
##  1st Qu.: 0.200   1st Qu.: 0.031   NA's:24691     1st Qu.: -3.400
##  Median : 1.400   Median : 0.075                  Median : -1.500
##  Mean   : 2.486   Mean   : 0.065                  Mean   : -1.778
##  3rd Qu.: 3.800   3rd Qu.: 0.115                  3rd Qu.:  0.300
##  Max.   :25.400   Max.   : 2.123                  Max.   : 47.800
##  NA's   :106      NA's   :590                     NA's   :3894
##       DBPM              BPM                VORP             FG
##  Min.   :-30.400   Min.   :-86.700   Min.   :-2.60   Min.   :   0.0
##  1st Qu.: -1.700   1st Qu.: -4.200   1st Qu.:-0.20   1st Qu.:  41.0
##  Median : -0.500   Median : -1.800   Median : 0.00   Median : 141.0
##  Mean   : -0.549   Mean   : -2.327   Mean   : 0.56   Mean   : 195.3
##  3rd Qu.:  0.700   3rd Qu.:  0.300   3rd Qu.: 0.90   3rd Qu.: 299.0
##  Max.   : 46.800   Max.   : 36.200   Max.   :12.40   Max.   :1597.0
##  NA's   :3894      NA's   :3894      NA's   :3894    NA's   :67
##       FGA              FG.              X3P              X3PA
##  Min.   :   0.0   Min.   :0.0000   Min.   :  0.00   Min.   :  0.0
##  1st Qu.:  99.0   1st Qu.:0.3930   1st Qu.:  0.00   1st Qu.:  1.0
##  Median : 321.0   Median :0.4390   Median :  2.00   Median : 11.0
##  Mean   : 430.6   Mean   :0.4308   Mean   : 22.21   Mean   : 63.6
##  3rd Qu.: 661.0   3rd Qu.:0.4800   3rd Qu.: 27.00   3rd Qu.: 84.0
##  Max.   :3159.0   Max.   :1.0000   Max.   :402.00   Max.   :886.0
##  NA's   :67       NA's   :166      NA's   :5764     NA's   :5764
##       X3P.             X2P              X2PA              X2P.
##  Min.   :0.000   Min.   :   0.0   Min.   :   0.0   Min.   :0.0000
##  1st Qu.:0.100   1st Qu.:  35.0   1st Qu.:  82.0   1st Qu.:0.4070
##  Median :0.292   Median : 122.0   Median : 270.0   Median :0.4560
##  Mean   :0.249   Mean   : 178.3   Mean   : 381.8   Mean   :0.4453
##  3rd Qu.:0.363   3rd Qu.: 268.0   3rd Qu.: 579.2   3rd Qu.:0.4960
```

```
##  Max.   :1.000   Max.   :1597.0   Max.   :3159.0   Max.   :1.0000  
##  NA's   :9275    NA's   :67       NA's   :67       NA's   :195     
##       eFG.              FT              FTA              FT.        
##  Min.   :0.0000   Min.   :  0.0   Min.   :   0.0   Min.   :0.0000  
##  1st Qu.:0.4140   1st Qu.: 18.0   1st Qu.:  27.0   1st Qu.:0.6570  
##  Median :0.4630   Median : 63.0   Median :  88.0   Median :0.7430  
##  Mean   :0.4507   Mean   :102.4   Mean   : 136.8   Mean   :0.7193  
##  3rd Qu.:0.5010   3rd Qu.:149.0   3rd Qu.: 201.0   3rd Qu.:0.8080  
##  Max.   :1.5000   Max.   :840.0   Max.   :1363.0   Max.   :1.0000  
##  NA's   :166      NA's   :67      NA's   :67       NA's   :925     
##       ORB              DRB              TRB              AST        
##  Min.   :  0.00   Min.   :   0.0   Min.   :   0.0   Min.   :   0.0  
##  1st Qu.: 12.00   1st Qu.:  33.0   1st Qu.:  51.0   1st Qu.:  19.0  
##  Median : 38.00   Median : 106.0   Median : 159.0   Median :  68.0  
##  Mean   : 62.19   Mean   : 147.2   Mean   : 224.6   Mean   : 114.9  
##  3rd Qu.: 91.00   3rd Qu.: 212.0   3rd Qu.: 322.0   3rd Qu.: 160.0  
##  Max.   :587.00   Max.   :1111.0   Max.   :2149.0   Max.   :1164.0  
##  NA's   :3894     NA's   :3894     NA's   :379      NA's   :67      
##       STL             BLK              TOV               PF        
##  Min.   :  0.0   Min.   :  0.00   Min.   :  0.00   Min.   :  0.0  
##  1st Qu.:  9.0   1st Qu.:  3.00   1st Qu.: 18.00   1st Qu.: 39.0  
##  Median : 29.0   Median : 11.00   Median : 55.00   Median :109.0  
##  Mean   : 39.9   Mean   : 24.47   Mean   : 73.94   Mean   :116.3  
##  3rd Qu.: 60.0   3rd Qu.: 29.00   3rd Qu.:112.00   3rd Qu.:182.0  
##  Max.   :301.0   Max.   :456.00   Max.   :464.00   Max.   :386.0  
##  NA's   :3894    NA's   :3894     NA's   :5046     NA's   :67     
##       PTS        
##  Min.   :   0.0  
##  1st Qu.: 106.0  
##  Median : 364.0  
##  Mean   : 510.1  
##  3rd Qu.: 778.0  
##  Max.   :4029.0  
##  NA's   :67      
```

```
head(stats)
```

```
##   X Year          Player Pos Age  Tm  G GS MP PER   TS. X3PAr   FTr ORB. DRB.
## 1 0 1950 Curly Armstrong G-F  31 FTW 63 NA NA  NA 0.368    NA 0.467   NA   NA
## 2 1 1950    Cliff Barker  SG  29 INO 49 NA NA  NA 0.435    NA 0.387   NA   NA
## 3 2 1950   Leo Barnhorst  SF  25 CHS 67 NA NA  NA 0.394    NA 0.259   NA   NA
## 4 3 1950      Ed Bartels   F  24 TOT 15 NA NA  NA 0.312    NA 0.395   NA   NA
## 5 4 1950      Ed Bartels   F  24 DNN 13 NA NA  NA 0.308    NA 0.378   NA   NA
## 6 5 1950      Ed Bartels   F  24 NYK  2 NA NA  NA 0.376    NA 0.750   NA   NA
##   TRB. AST. STL. BLK. TOV. USG. blanl  OWS  DWS   WS WS.48 blank2 OBPM DBPM BPM
## 1   NA   NA   NA   NA   NA   NA    NA -0.1  3.6  3.5    NA     NA   NA   NA  NA
## 2   NA   NA   NA   NA   NA   NA    NA  1.6  0.6  2.2    NA     NA   NA   NA  NA
## 3   NA   NA   NA   NA   NA   NA    NA  0.9  2.8  3.6    NA     NA   NA   NA  NA
## 4   NA   NA   NA   NA   NA   NA    NA -0.5 -0.1 -0.6    NA     NA   NA   NA  NA
## 5   NA   NA   NA   NA   NA   NA    NA -0.5 -0.1 -0.6    NA     NA   NA   NA  NA
## 6   NA   NA   NA   NA   NA   NA    NA  0.0  0.0  0.0    NA     NA   NA   NA  NA
##   VORP  FG FGA   FG. X3P X3PA X3P. X2P X2PA  X2P.  eFG.  FT FTA   FT. ORB DRB
## 1   NA 144 516 0.279  NA   NA   NA 144  516 0.279 0.279 170 241 0.705  NA  NA
## 2   NA 102 274 0.372  NA   NA   NA 102  274 0.372 0.372  75 106 0.708  NA  NA
## 3   NA 174 499 0.349  NA   NA   NA 174  499 0.349 0.349  90 129 0.698  NA  NA
```

```
## 4    NA   22   86 0.256   NA    NA    NA  22    86 0.256 0.256   19   34 0.559   NA   NA
## 5    NA   21   82 0.256   NA    NA    NA  21    82 0.256 0.256   17   31 0.548   NA   NA
## 6    NA    1    4 0.250   NA    NA    NA   1     4 0.250 0.250    2    3 0.667   NA   NA
##    TRB AST STL BLK TOV  PF PTS
## 1   NA 176  NA  NA  NA 217 458
## 2   NA 109  NA  NA  NA  99 279
## 3   NA 140  NA  NA  NA 192 438
## 4   NA  20  NA  NA  NA  29  63
## 5   NA  20  NA  NA  NA  27  59
## 6   NA   0  NA  NA  NA   2   4
```

Because the one dataset only has 17-18 data I want to limit the stats dataset

```
stats17 <-
  stats %>% filter(Year >= 2017) %>%
  dplyr::select(Year:G, MP, PER, FG:PTS) %>%
  distinct(Player, .keep_all = TRUE) %>%
  mutate(MPG = MP/G, PPG = PTS/G, APG = AST/G,
         RPG = TRB/G, TOPG = TOV/G, BPG = BLK/G, SPG = STL/G)
```

Merge the data

```
nba_final <- merge(stats17, nba, by.x = "Player", by.y = "Player")
names(nba_final)[40] <- "salary17_18"
nba_final <- nba_final[-39]
```

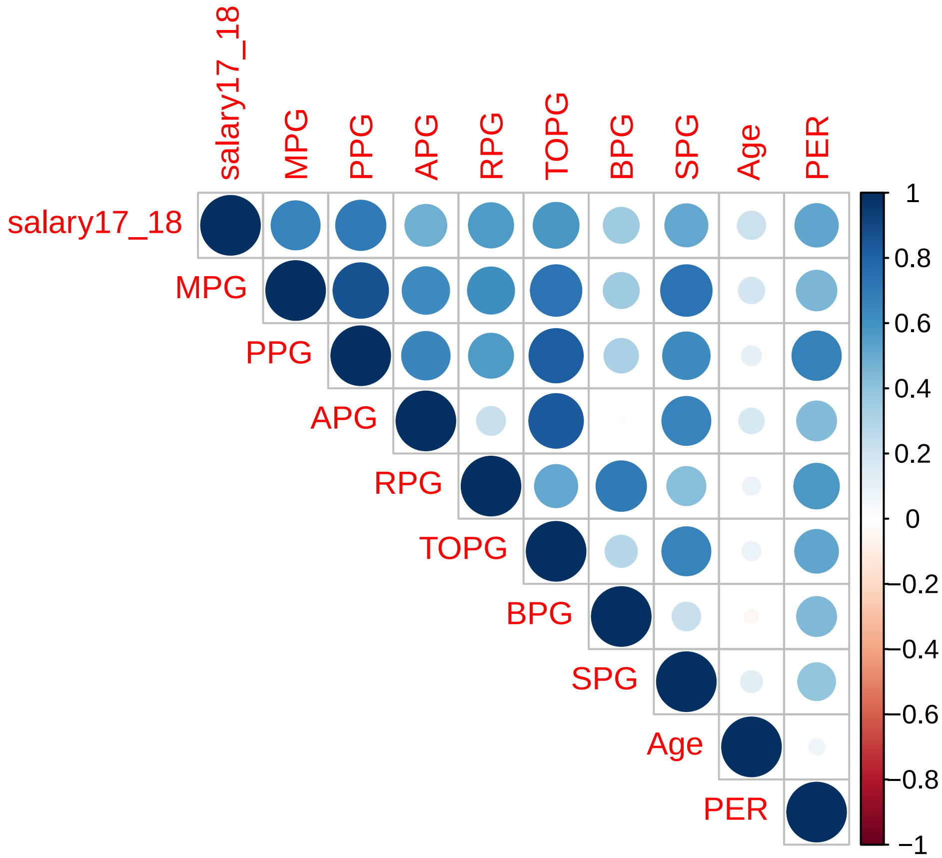
Check for missing values

```
sapply(nba_final, function(x) sum(is.na(x)))
```

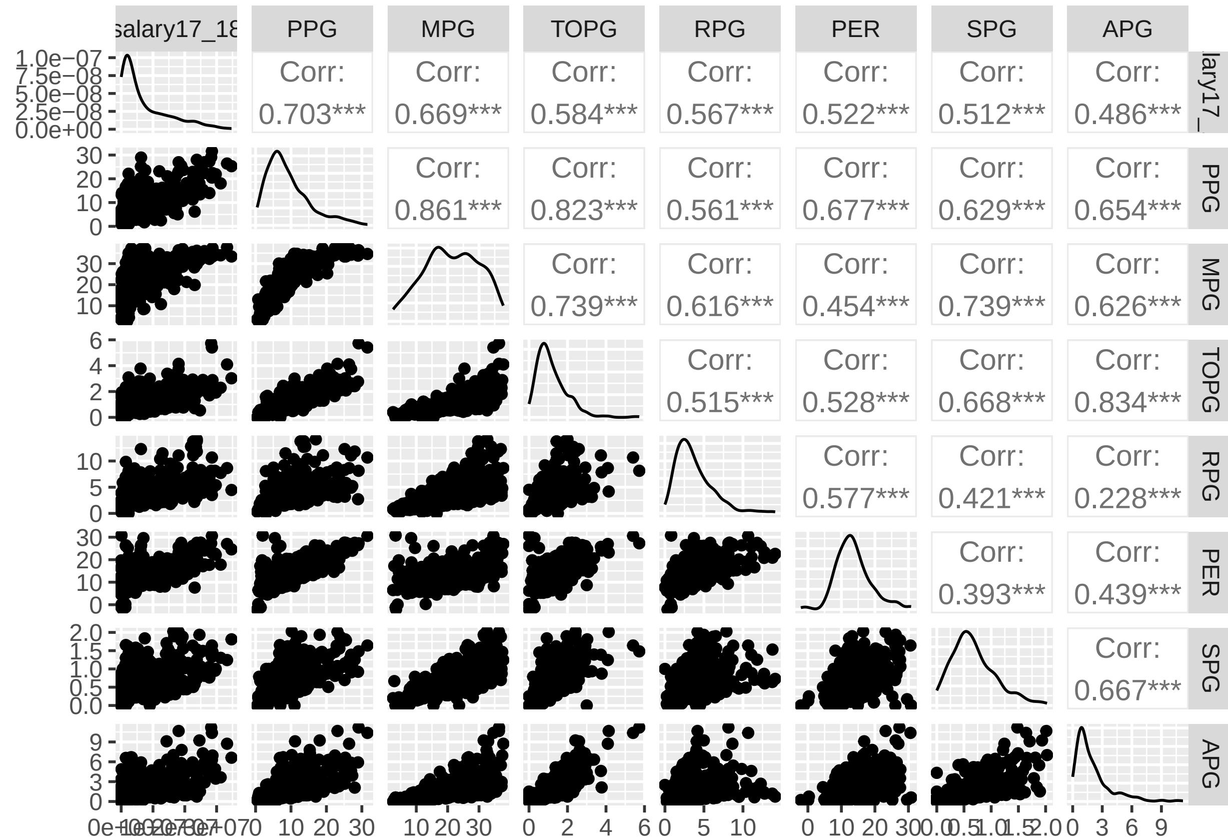
```
##      Player        Year         Pos         Age        Tm.x           G
##           0           0           0           0           0           0
##          MP         PER          FG         FGA         FG.         X3P
##           0           0           0           0           0           0
##        X3PA        X3P.         X2P        X2PA        X2P.        eFG.
##           0          28           0           0           0           0
##          FT         FTA         FT.         ORB         DRB         TRB
##           0           0           4           0           0           0
##         AST         STL         BLK         TOV          PF         PTS
##           0           0           0           0           0           0
##         MPG         PPG         APG         RPG        TOPG         BPG
##           0           0           0           0           0           0
##         SPG           X salary17_18
##           0           0           0
```

Correlation and Variable Importance Selection

```
corrplot(cor(nba_final %>%
               dplyr::select(salary17_18, MPG:SPG,
                      Age, PER, contains("%")),
             use = "complete.obs"),
         method = "circle",type = "upper")
```




```
nba_final2 <-
  nba_final %>%
  dplyr::select(salary17_18, PPG, MPG, TOPG, RPG, PER, SPG, APG)
ggpairs(nba_final2)
```



What does this tell me?

This is a really small dataset (shudder) as it is 573 records. In an ideal world I so wouldn't use this dataset. It's also weirdly clean.. as there are no missing values.

Let's do some plots anyways

```
head(nba_final)
```

```
##              Player Year Pos Age Tm.x  G   MP  PER  FG FGA   FG. X3P X3PA  X3P.
## 1    A.J. Hammons 2017   C  24  DAL 22  163  8.4  17  42 0.405   5   10 0.500
## 2    Aaron Brooks 2017  PG  32  IND 65  894  9.5 121 300 0.403  48  128 0.375
## 3    Aaron Gordon 2017  SF  21  ORL 80 2298 14.4 393 865 0.454  77  267 0.288
## 4 Al-Farouq Aminu 2017  SF  26  POR 61 1773 11.3 183 466 0.393  70  212 0.330
## 5      Al Horford 2017   C  30  BOS 68 2193 17.7 379 801 0.473  86  242 0.355
## 6    Al Jefferson 2017   C  32  IND 66  931 18.9 235 471 0.499   0    1 0.000
##   X2P X2PA  X2P.  eFG.  FT FTA   FT. ORB DRB TRB AST STL BLK TOV  PF  PTS
## 1  12   32 0.375 0.464   9  20 0.450   8  28  36   4   1  13  10  21   48
## 2  73  172 0.424 0.483  32  40 0.800  18  51  69 125  25   9  66  93  322
## 3 316  598 0.528 0.499 156 217 0.719 116 289 405 150  64  40  89 172 1019
## 4 113  254 0.445 0.468  96 136 0.706  77 374 451  99  60  44  94 102  532
## 5 293  559 0.524 0.527 108 135 0.800  95 369 464 337  52  87 116 138  952
## 6 235  470 0.500 0.499  65  85 0.765  75 203 278  57  19  16  33 125  535
##         MPG       PPG       APG      RPG      TOPG       BPG        SPG   X
## 1  7.409091  2.181818 0.1818182 1.636364 0.4545455 0.5909091 0.04545455 411
## 2 13.753846  4.953846 1.9230769 1.061538 1.0153846 0.1384615 0.38461538 319
## 3 28.725000 12.737500 1.8750000 5.062500 1.1125000 0.5000000 0.80000000 190
## 4 29.065574  8.721311 1.6229508 7.393443 1.5409836 0.7213115 0.98360656 154
```

```
## 5 32.250000 14.000000 4.9558824 6.823529 1.7058824 1.2794118 0.76470588  11
## 6 14.106061  8.106061 0.8636364 4.212121 0.5000000 0.2424242 0.28787879 128
##   salary17_18
## 1     1312611
## 2     2116955
## 3     5504420
## 4     7319035
## 5    27734405
## 6     9769821
```

So what does this tell me? I'm not going to do all the talking so what do you think this tells you?

1) What did you learn?
2) What do you wish to know that you don't?
3) Are there any concerns?
4) How you should you divide out the predictions? Should you treat all players the same? All teams the same? Why? Or Why not?

You could do this in 1 of 2 ways. You could either break it out and consider teams as a factor but the model going to do that somewhat anyways for you. Or you could break out the super high players from everyone else by basically creating buckets.

Bucket Your Data You can probably get away with doing 2 buckets, high and everyone else.

Question for the group Why can't I use "x" and "season17_18" in a correlation problem? What is the problem with "x"?

Train/Test Time

Another thing that makes this dataset uck, is that there is no 3rd dataset. But such is life.

```
#make this example reproducible
set.seed(42)

#use 70% of dataset as training set and 30% as test set
sample <- sample(c(TRUE, FALSE), nrow(nba), replace=TRUE, prob=c(0.7,0.3))
train  <- nba[sample, ]
test   <- nba[!sample, ]
```

Model Time

Let's start with linear regression. Yes you can use the ln call and if anything that's probably easier. But to make the comparision easier I'm not going to.

```
lr1 <- glm(season17_18 ~ Tm, data = train)
lr2 <- glm(season17_18 ~ Tm + X, data = train)
```

Poisson Time

```
poisson1 <- glm(season17_18 ~ Tm, data = train, family = poisson)
poisson2 <- glm(season17_18 ~ Tm + X, data = train,
                family = poisson)
```

Negative Binomial

```
nb1 <- glm.nb(season17_18 ~ Tm, data = train)
nb2 <- glm.nb(season17_18 ~ Tm + X, data = train)
```

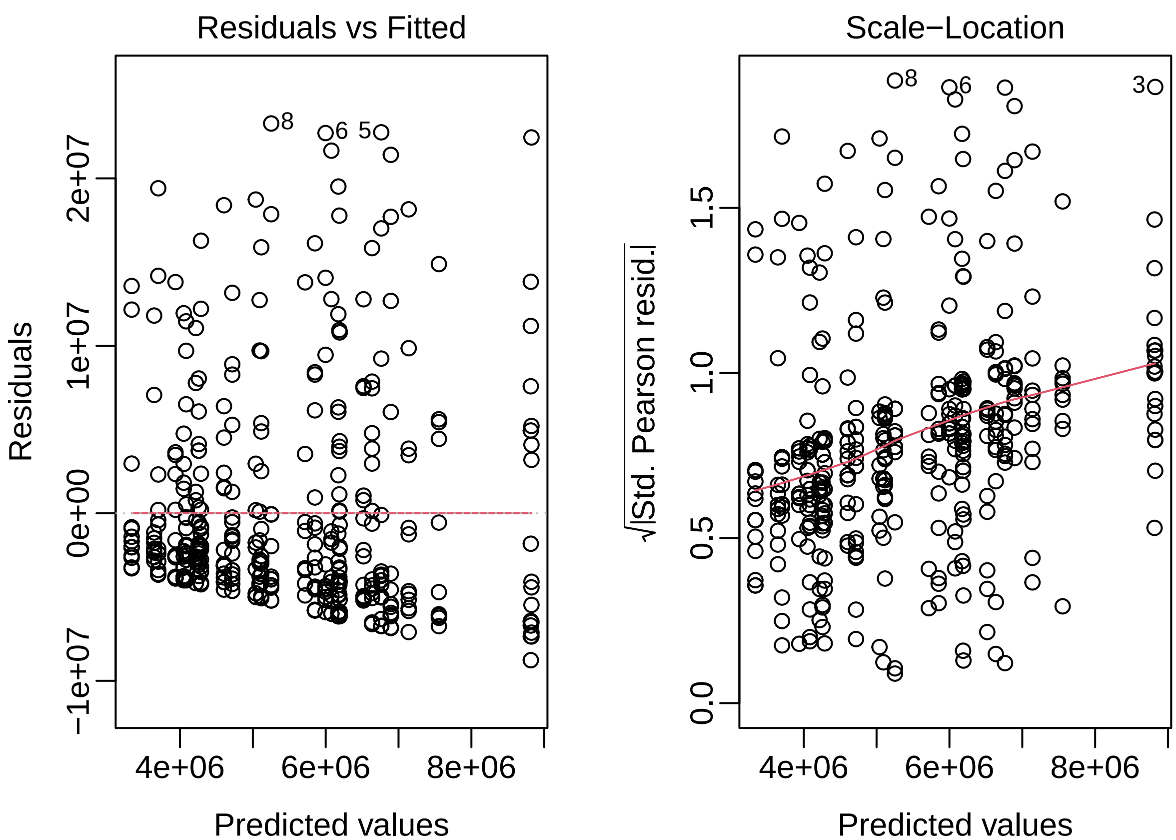
Quasi-Likelihood

```r
quasi1 <- glm(season17_18 ~ Tm, data = train,
              family = quasipoisson)
quasi2 <- glm(season17_18 ~ Tm + X, data = train,
              family = quasipoisson)
```

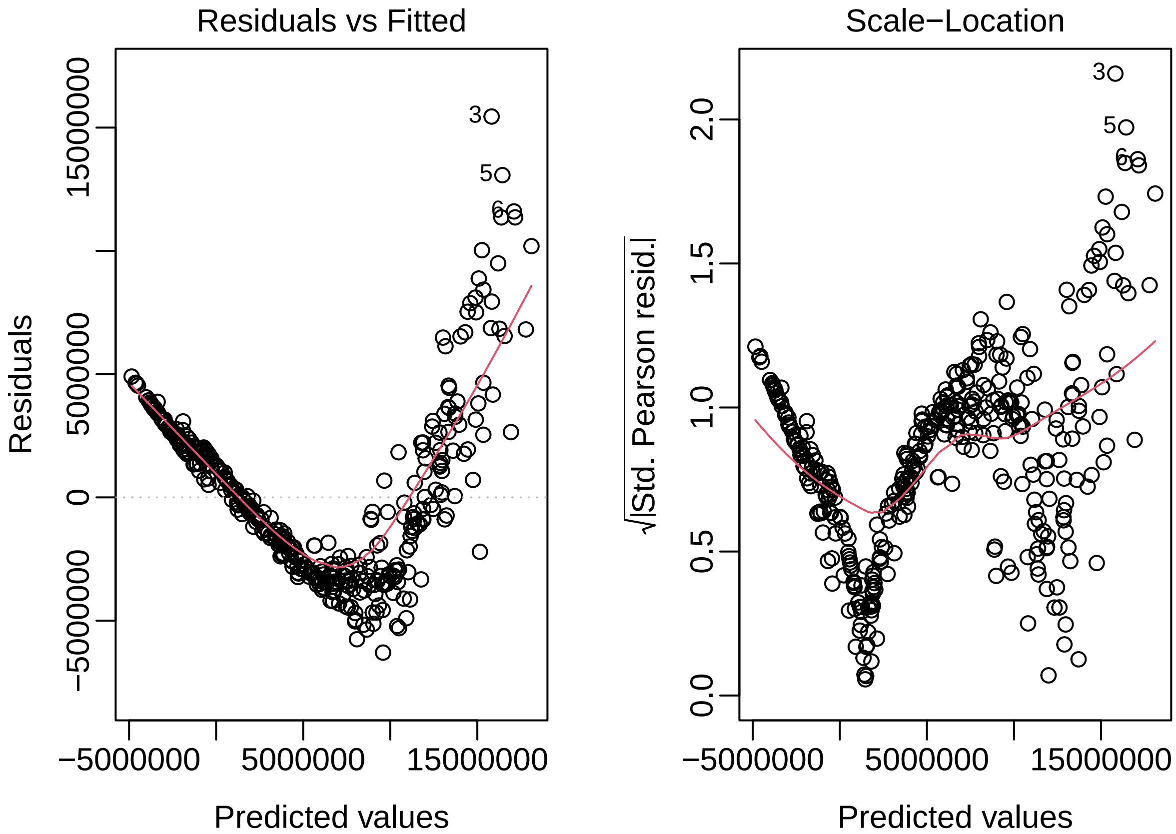
Ok I built a bunch of models now what? Now I'm decently confident these are bad but which one is the "least bad". How would I know? What should I look for?

Let's plot some residuals

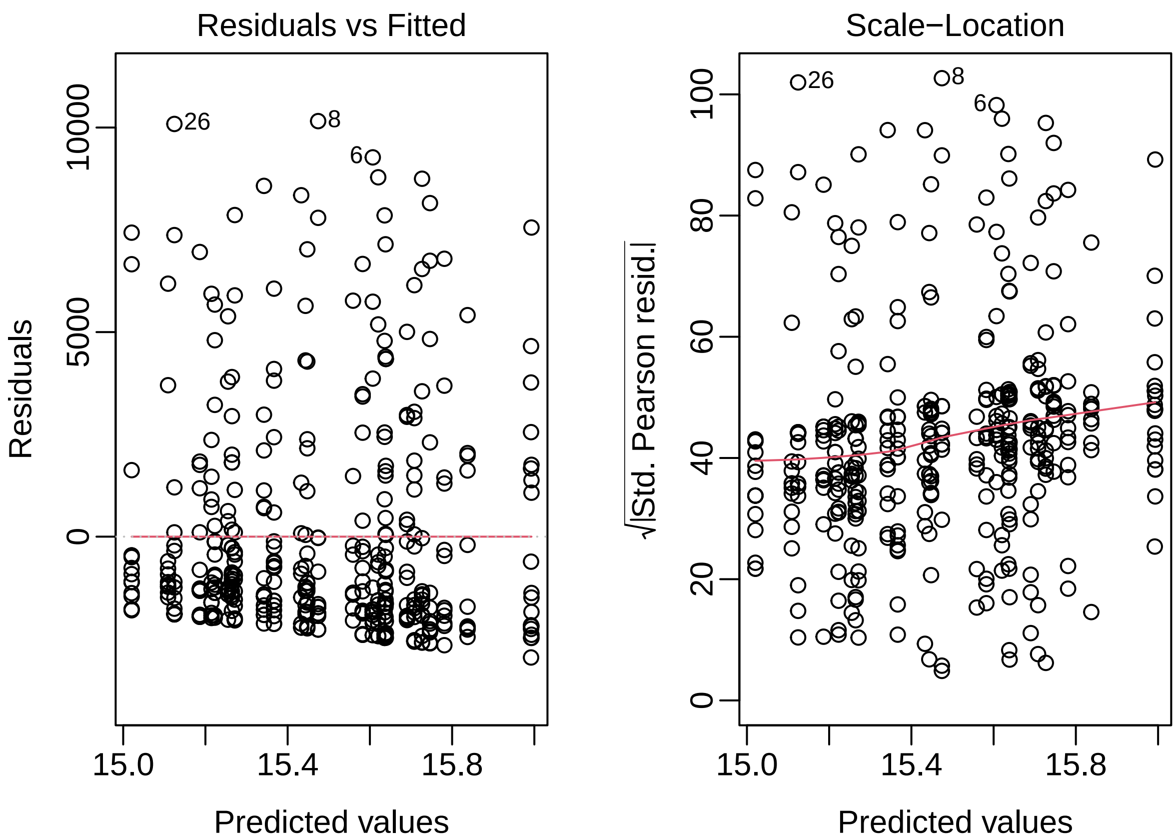
```r
par(mfrow=c(1,2),mar=c(3,3,2,2),mgp=c(2,0.5,0))
plot(lr1, which=c(1,3)) #linear regression 1
```



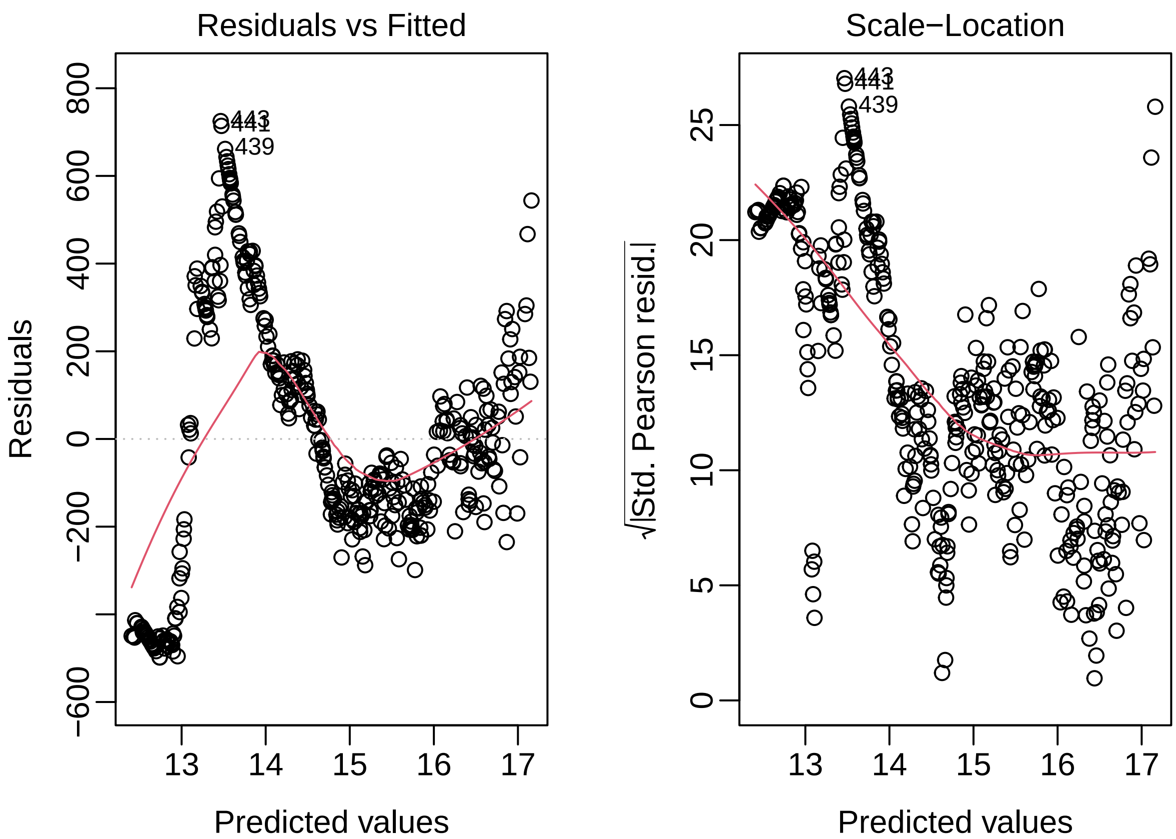

```r
plot(lr2, which=c(1,3)) #linear regression 2
```



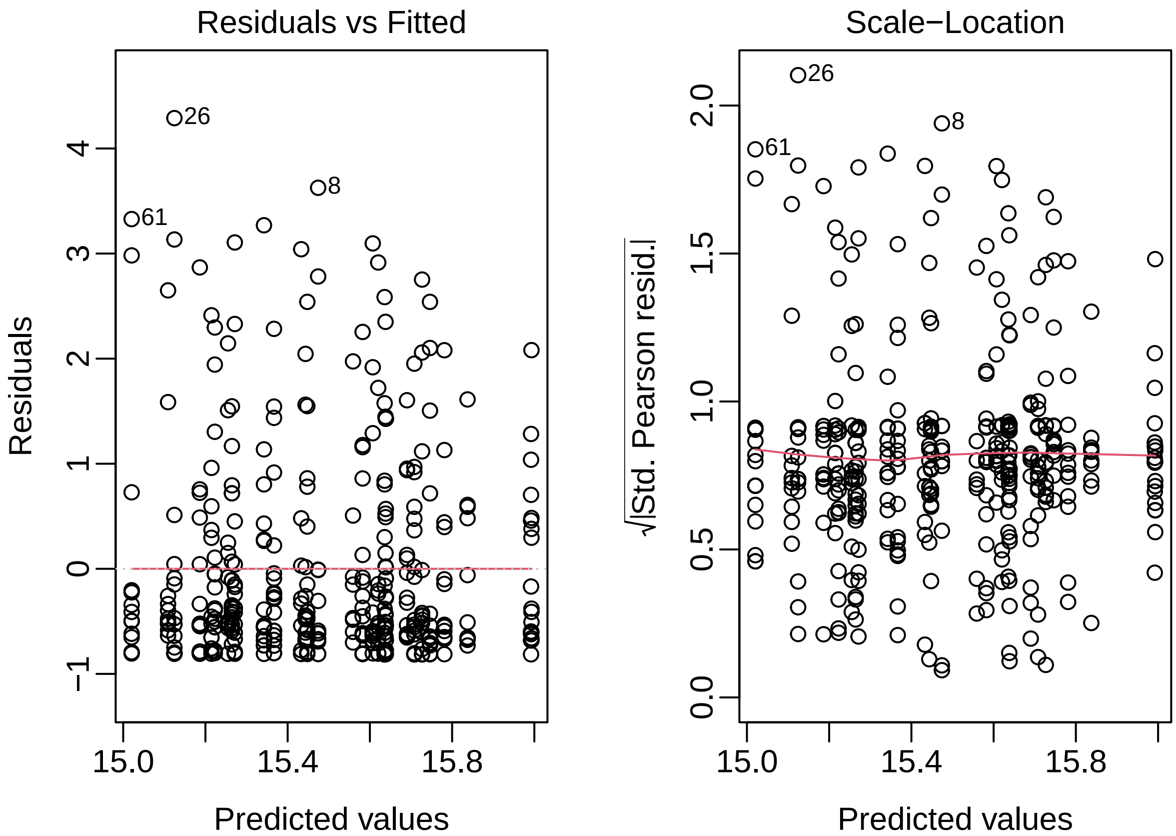

```
plot(poisson1, which=c(1,3)) #poisson regression 1
```



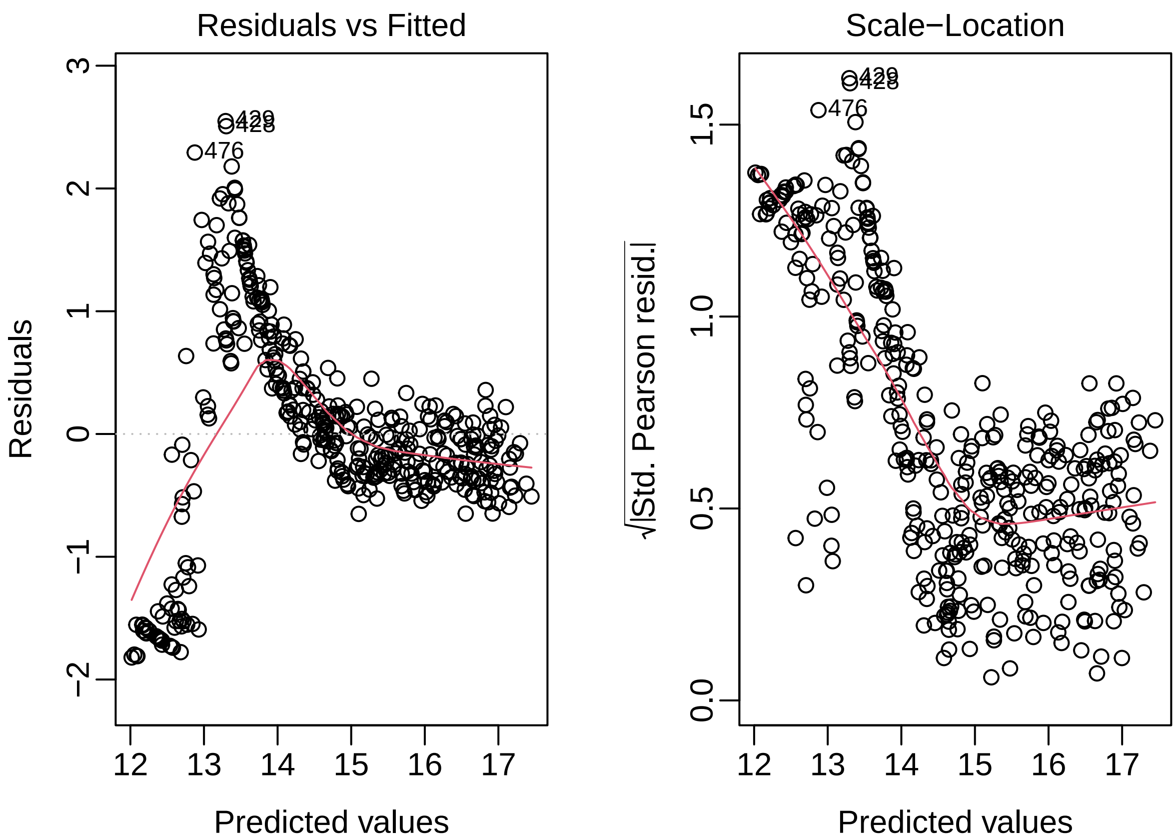

```
plot(poisson2, which=c(1,3)) #poisson regression 2
```



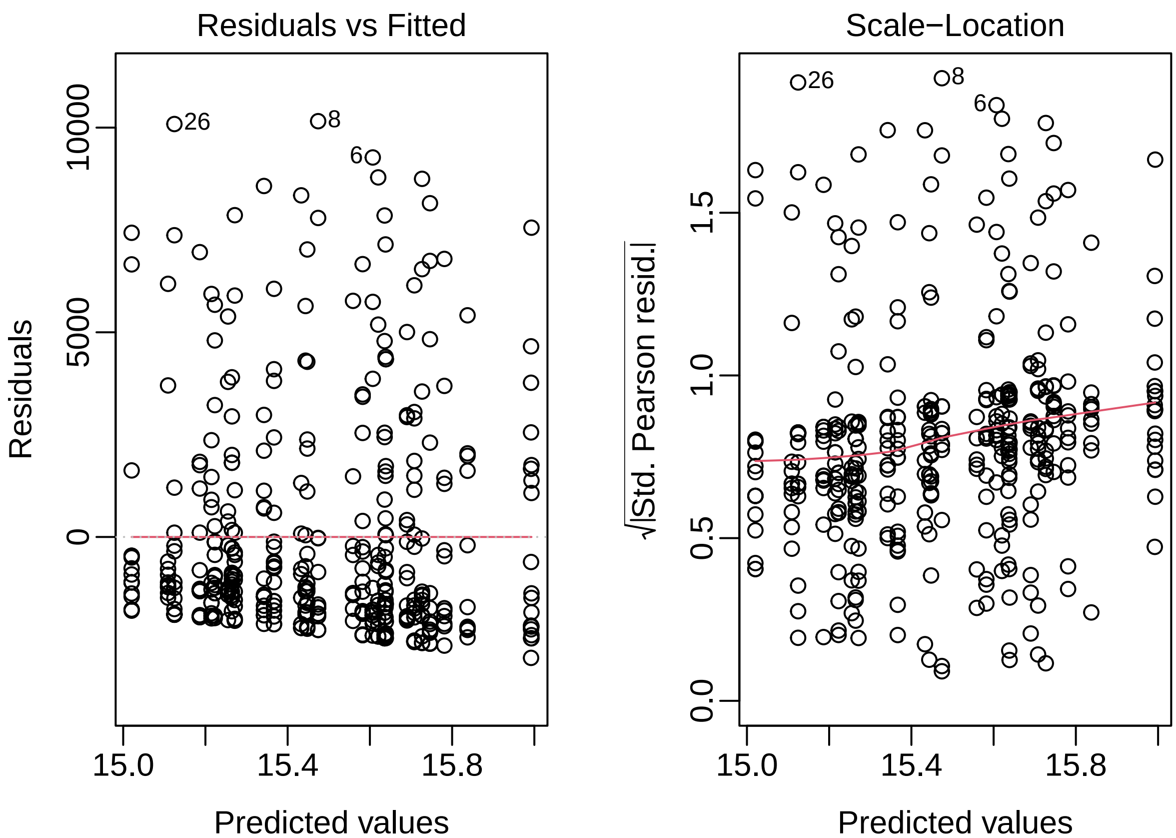

```r
plot(nb1, which=c(1,3)) #NB 1
```



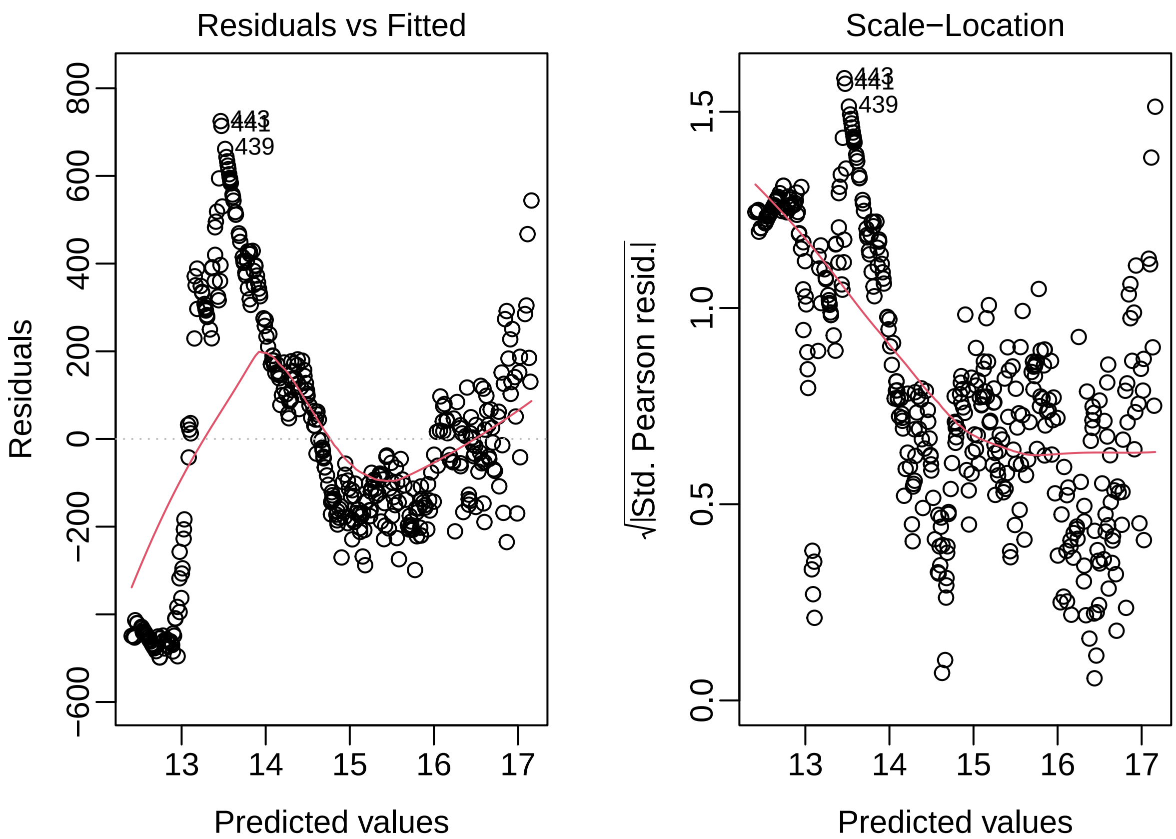

```
plot(nb2, which=c(1,3)) #NB 2
```




```r
plot(quasi1, which=c(1,3)) #Quasi 1
```




```r
plot(quasi2, which=c(1,3)) #Quasi 2
```



From a Residual Standpoint which has the "best" or "least worst" result?

Let's Look at AIC and BIC. Are there any of these models that I can't use AIC or BIC for? Do you remember why?

```
AIC(lr1, lr2, poisson1, poisson2, nb1, nb2)
```

```
##          df          AIC
## lr1      31 1.388836e+04
## lr2      32 1.335347e+04
## poisson1 30 2.595151e+09
## poisson2 31 3.461092e+07
## nb1      31 1.332277e+04
## nb2      32 1.240942e+04
```

```
BIC(lr1, lr2, poisson1, poisson2, nb1, nb2)
```

```
##          df          BIC
## lr1      31 1.401241e+04
## lr2      32 1.348152e+04
## poisson1 30 2.595151e+09
## poisson2 31 3.461105e+07
## nb1      31 1.344681e+04
## nb2      32 1.253747e+04
```

Ok what model would you pick? Why?

Stepwise Selection

So I manually did this but there is actually a trick. There are pros and cons to the trick but I wanted you to experience building manually first. I already gave you one trick of building a tree but you can also get R to do the stepwise for you.