Stats 101A Homework 5 Lecture 1B

Charles Liu (304804942)

3/6/2020

Loading Necessary Data/Models:

```
library(tidyverse)
library(dplyr)
library(ggplot2)
library(stringr)
library(car)
library(readr)
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 3.6.3
```

```
library(mice)
```

```
## Warning: package 'mice' was built under R version 3.6.3
```

```
library(gridExtra)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 3.6.3
```

```
FifaNoY <- read.csv("C:/Users/cliuk/Documents/UCLA Works/UCLA Winter 2020/Stats 101A/Project/Project At

FifaTrainNew <- read.csv("C:/Users/cliuk/Documents/UCLA Works/UCLA Winter 2020/Stats 101A/Project/Proje

FifaTrainNew$Overall2 <- FifaTrainNew$Overall
FifaNoY$Overall2 <- FifaNoY$Overall^2


#attach(FifaTrainNew)
RepmeanByClub <- FifaTrainNew %>%
  group_by(Club) %>%
  summarise(
    average.wage = mean(WageNew),
    average.reputation = mean(International.Reputation)
  ) %>%
  arrange(desc(average.wage))

#RepmeanByClub
#attempt to group Club by international
FifaTrainNew <- left_join(FifaTrainNew, RepmeanByClub[,c("Club","average.reputation")], by = c("Club" =
FifaNoY <- left_join(FifaNoY, RepmeanByClub[,c("Club","average.reputation")], by = c("Club" = "Club"))
```

```r
FifaTrainNew$ClubRep2 <- ifelse(FifaTrainNew$average.reputation >= 2, "very high", ifelse(FifaTrainNew$a


FifaNoY$ClubRep2 <- ifelse(FifaNoY$average.reputation >= 2, "very high", ifelse(FifaNoY$average.reputati

#MLR
SLR3 <- lm(log(WageNew) ~ 0 + Overall2 + ClubRep2, data = FifaTrainNew)
summary(SLR3)
```

```
##
## Call:
## lm(formula = log(WageNew) ~ 0 + Overall2 + ClubRep2, data = FifaTrainNew)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.1579 -0.3588  0.0363  0.4342  2.2643
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## Overall2          0.111017   0.001073  103.45   <2e-16 ***
## ClubRep2high      2.474162   0.101645   24.34   <2e-16 ***
## ClubRep2low       1.092261   0.072145   15.14   <2e-16 ***
## ClubRep2mid       1.823892   0.081197   22.46   <2e-16 ***
## ClubRep2mid-high  1.647238   0.085392   19.29   <2e-16 ***
## ClubRep2mid-low   1.067421   0.071791   14.87   <2e-16 ***
## ClubRep2very high 2.500001   0.097744   25.58   <2e-16 ***
## ClubRep2very low  1.051369   0.070170   14.98   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7186 on 11324 degrees of freedom
##   (1413 observations deleted due to missingness)
## Multiple R-squared:  0.9931, Adjusted R-squared:  0.9931
## F-statistic: 2.029e+05 on 8 and 11324 DF,  p-value: < 2.2e-16
```

```r
anova(SLR3)
```

```
## Analysis of Variance Table
##
## Response: log(WageNew)
##              Df Sum Sq Mean Sq    F value    Pr(>F)
## Overall2      1 837237  837237 1621510.09 < 2.2e-16 ***
## ClubRep2      7    930     133     257.34 < 2.2e-16 ***
## Residuals 11324   5847       1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
leverages <- hatvalues(SLR3)
# h_ii > 2 * (p+1)/n
leverage_points <- which(leverages > 2 * mean(leverages))
outliers <- which(abs(rstandard(SLR3)) > 2)
bad_leverages <- intersect(leverage_points, outliers)

SLR3.1 <- lm(log(WageNew) ~ 0 + Overall2 + ClubRep2, data = FifaTrainNew[-bad_leverages,])
summary(SLR3.1)
```

```
##
## Call:
## lm(formula = log(WageNew) ~ 0 + Overall2 + ClubRep2, data = FifaTrainNew[-bad_leverages,
##     ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.1576 -0.3590  0.0363  0.4345  2.2646
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## Overall2           0.111002   0.001076  103.12   <2e-16 ***
## ClubRep2high       2.475272   0.101867   24.30   <2e-16 ***
## ClubRep2low        1.092852   0.072375   15.10   <2e-16 ***
## ClubRep2mid        1.824980   0.081451   22.41   <2e-16 ***
## ClubRep2mid-high   1.648535   0.085642   19.25   <2e-16 ***
## ClubRep2mid-low    1.068248   0.071998   14.84   <2e-16 ***
## ClubRep2very high 2.501157   0.097982   25.53   <2e-16 ***
## ClubRep2very low   1.051946   0.070382   14.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7191 on 11296 degrees of freedom
##   (1410 observations deleted due to missingness)
## Multiple R-squared:  0.9931, Adjusted R-squared:  0.9931
## F-statistic: 2.021e+05 on 8 and 11296 DF,  p-value: < 2.2e-16
```

Q1) Report the following from your training data used to create your latest MLR:

1a) State your name and your group name.

Group Kaggle Lec 1A

1b) The dimension of your training data after cleaning the NAs

```
dim(FifaTrainNew)
```

```
## [1] 12745    83
```

12745 rows (observations) , 80 columns (variables)

1c) Summary statistics of your response variable only.

```
summary(FifaTrainNew$WageNew)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       6    2004    3926   11433   10528  650305
```

1d) How many predictors used to create your latest MLR.

Two: Overall2 and ClubRep2, but on our summary, we have 8 categories total since we grouped our predictors into categories and have numerical predictor.

1e) Classify your predictors: Categorical or Numerical: Template Table

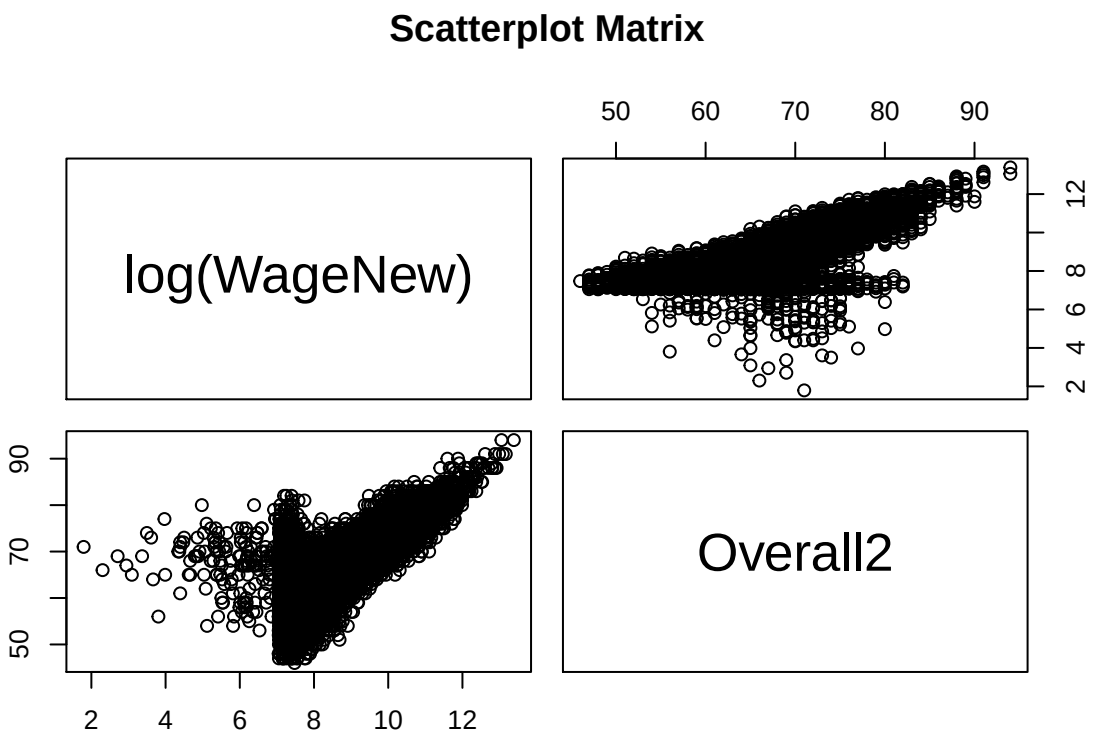
Categorical: ClubRep2 Numerical: Overall2

1f) Report your latest R2 and latest Rank on Kaggle.

R-Squared = 0.94183 Rank: 43

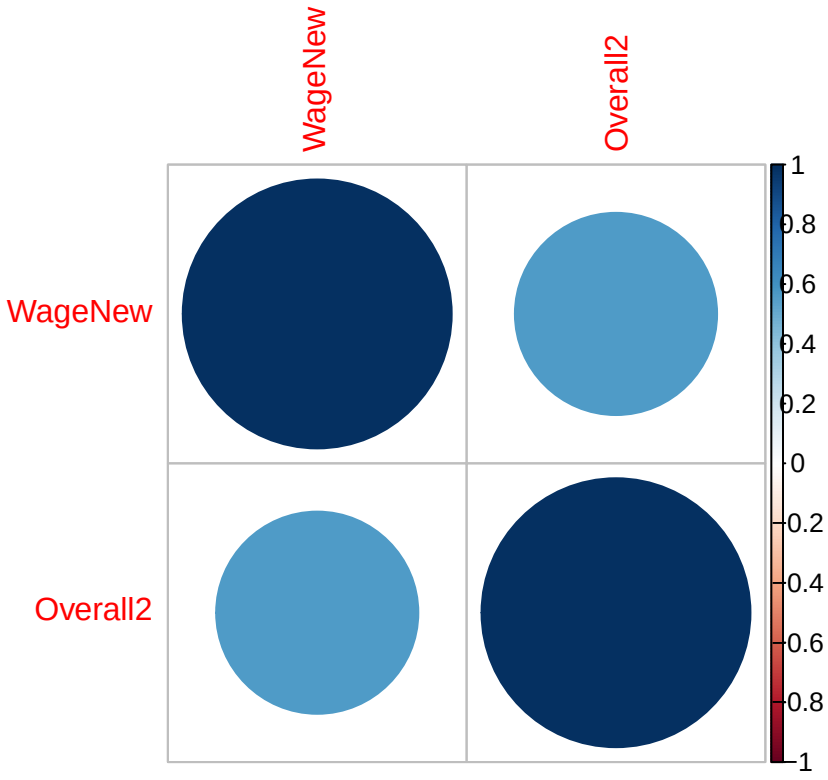
1g) Create matrix plot for your variables.

```
pairs(~log(WageNew) + Overall2, data = FifaTrainNew, main = "Scatterplot Matrix")
```

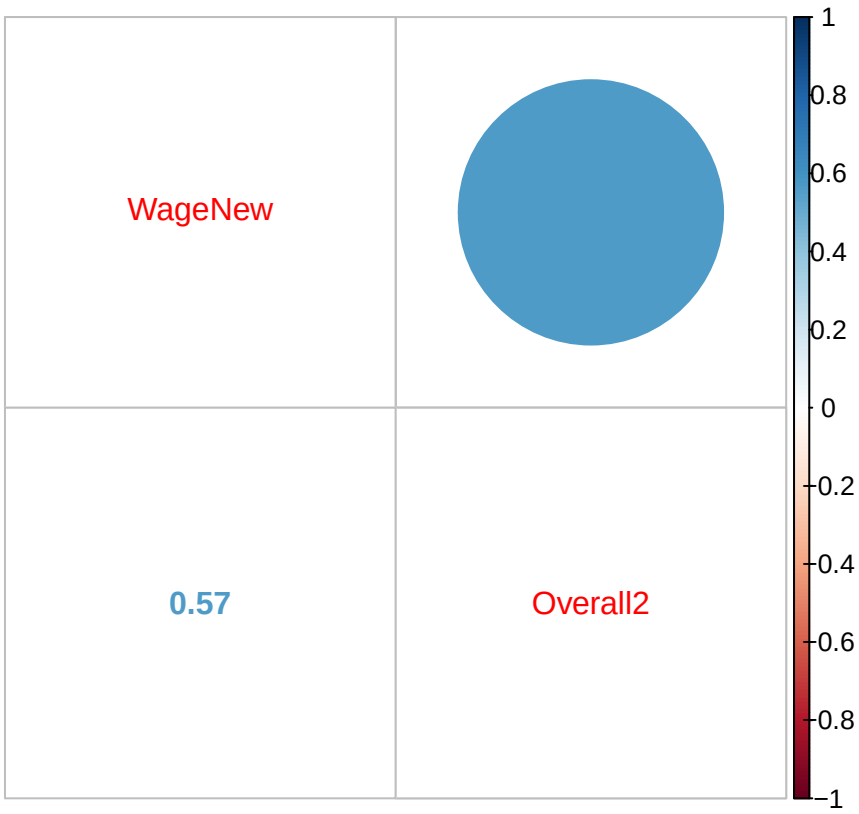


1h) Create corrplot of your numerical variables.

```
corrplot(cor(FifaTrainNew[ ,c('WageNew', 'Overall2')]), method = "circle")
```




```
corrplot.mixed(cor(FifaTrainNew[ ,c('WageNew', 'Overall2')]))
```



Q2) Have you used any transformation on your predictors or on your response

variable?

2a) If yes, explain how did you decide what transformation to be used. List the variables and the transformation function used in your latest MLR. (Provide proofs of your work).

```
summary(powerTransform(cbind(WageNew, Overall) ~ 1, data = FifaTrainNew))
```

```
## bcPower Transformations to Multinormality
##         Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## WageNew    0.0002        0.00      -0.0092       0.0095
## Overall    2.1935        2.19       2.0801       2.3069
##
## Likelihood ratio test that transformation parameters are equal to 0
##  (all log transformations)
##                               LRT df       pval
## LR test, lambda = (0 0) 1450.382  2 < 2.22e-16
##
## Likelihood ratio test that no transformations are needed
##                               LRT df       pval
## LR test, lambda = (1 1) 43075.41  2 < 2.22e-16
```

2b) If no, explain why the suggested transformations did not work out for your latest MLR. (Provide proofs of your work).

N/A

Q3) Report the following from your latest MLR:

3a) Anova table of your MLR

```
anova(SLR3.1)
```

```
## Analysis of Variance Table
##
## Response: log(WageNew)
##              Df Sum Sq Mean Sq    F value    Pr(>F)
## Overall2      1 835178  835178 1615163.10 < 2.2e-16 ***
## ClubRep2      7    929     133     256.72 < 2.2e-16 ***
## Residuals 11296   5841       1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3b) Sort your predictors by their importance or contributions

```
# 1. Overall2
# 2. ClubRep2
anova(SLR3.1)[order(anova(SLR3.1)$`Sum Sq`, decreasing=TRUE),]
```

```
## Analysis of Variance Table
##
## Response: log(WageNew)
##              Df Sum Sq Mean Sq    F value    Pr(>F)
## Overall2      1 835178  835178 1615163.10 < 2.2e-16 ***
## Residuals 11296   5841       1
## ClubRep2      7    929     133     256.72 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3c) Report R2 and your R2-Adjusted of your MLR using the training data.

R^2 = 0.9934 R^2-adj = 0.9934

3d) Report the VIF of every predictor in your MLR make sure you have no multicollinearity violation (No predictor has a VIF exceeding five). Use the following template

```
vif(lm(log(WageNew)~ Overall2 + ClubRep2, data = FifaTrainNew))
```

```
##              GVIF Df GVIF^(1/(2*Df))
## Overall2 1.226263  1        1.107368
## ClubRep2 1.226263  6        1.017143
```

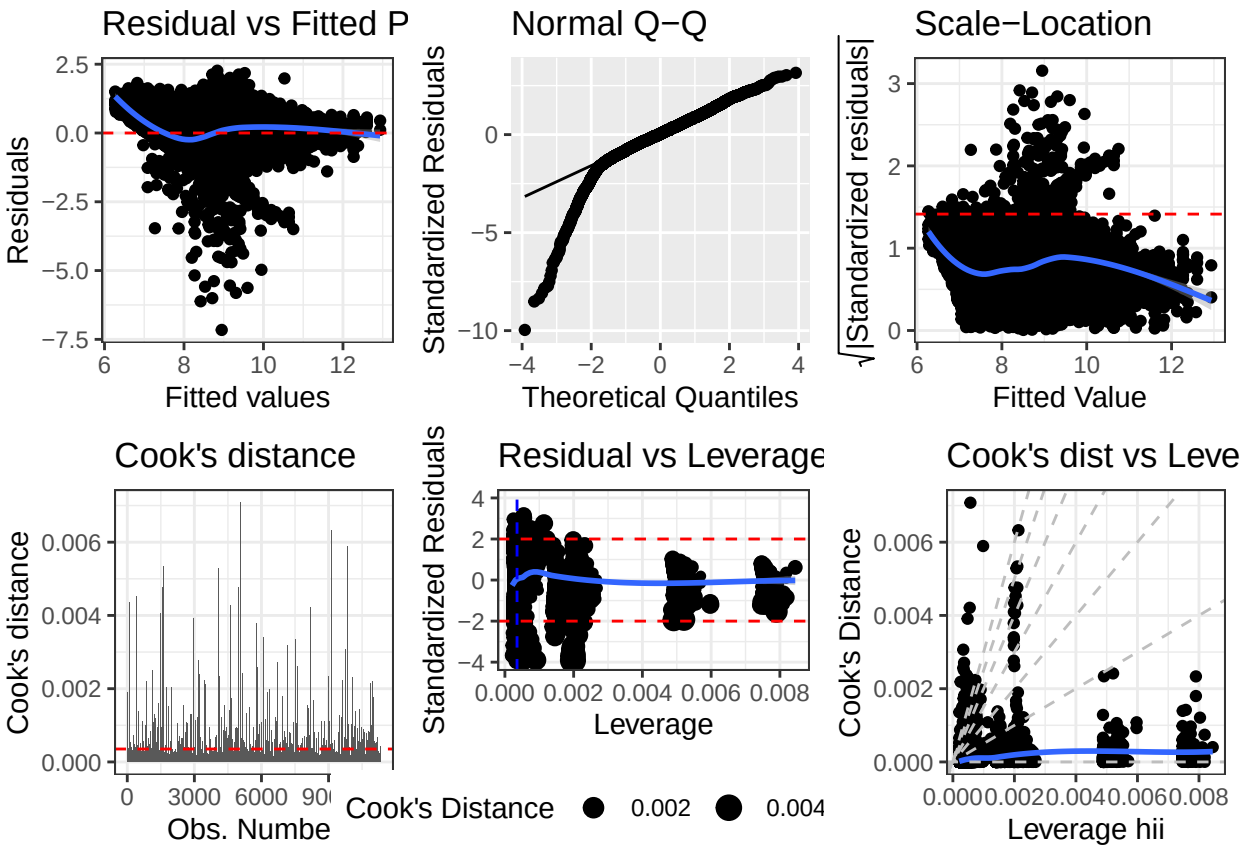
Q4) Report the following:

4a) Diagnostics six plots of your latest MLR. Comment on how well or how bad your MLR.

```
diagPlot<-function(model){
  p1<-ggplot(model, aes(model$fitted,
                        model$residuals),label=rownames(bonds))+geom_point()
  p1<-p1+stat_smooth(method="loess")+geom_hline(yintercept=0, col="red", linetype="dashed")
  p1<-p1+xlab("Fitted values")+ylab("Residuals")
  p1<-p1+ggtitle("Residual vs Fitted Plot")+theme_bw()
  p2<-ggplot(model,aes(sample=rstandard(model))) + stat_qq() + stat_qq_line()
  p2<-p2+xlab("Theoretical Quantiles")+ylab("Standardized Residuals")
  p2<-p2+ggtitle("Normal Q-Q")
  p3<-ggplot(model, aes(model$fitted,sqrt(abs(rstandard(model)))))+geom_point(na.rm=TRUE)
  p3<-p3+stat_smooth(method="loess", na.rm = TRUE)+xlab("Fitted Value")
  p3<-p3+ylab(expression(sqrt("|Standardized residuals|")))
  p3<-p3+ggtitle("Scale-Location")+theme_bw()+geom_hline(yintercept=sqrt(2),col="red", linetype="dashed
  p4<-ggplot(model, aes(seq_along(cooks.distance(model)),cooks.distance(model)))+geom_bar(stat="identity
  p4<-p4+xlab("Obs. Number")+ylab("Cook's distance")
  p4<-p4+ggtitle("Cook's distance")+theme_bw()+geom_hline(yintercept=4/(length(model$residuals-2)), col=
  p5<-ggplot(model,aes(hatvalues(model),rstandard(model)))+geom_point(aes(size=cooks.distance(model)), r
  p5<-p5+stat_smooth(method="loess", na.rm=TRUE)
  p5<-p5+xlab("Leverage")+ylab("Standardized Residuals")
  p5<-p5+ggtitle("Residual vs Leverage Plot")
  p5<-p5+scale_size_continuous("Cook's Distance", range=c(1,5))
  p5<-p5+theme_bw()+theme(legend.position="bottom")+geom_hline(yintercept=c(-2,2), col="red", linetype=
  p6<-ggplot(model,aes(hatvalues(model),cooks.distance(model)))+geom_point(na.rm=TRUE)+stat_smooth(metho
  p6<-p6+xlab("Leverage hii")+ylab("Cook's Distance")
  p6<-p6+ggtitle("Cook's dist vs Leverage")
  p6<-p6+geom_abline(slope=seq(0,3,0.5), color="gray", linetype="dashed")
  p6<-p6+theme_bw()
  return(grid.arrange(p1,p2,p3,p4,p5,p6,ncol=3)) }



diagPlot(SLR3)
```



We notice that for the most part our residuals are mostly randomly distributed, with a slight dip and increase, with a few violations with large negative residuals. Our errors are also about normally distributed, except at the lower end of the theoretical quantiles. We wouuld need to investigate further on how to make our residuals more normally distributed there. For the most part, we have constant variance in our models, which is a good sign. There seem to be a few violaters with extermely high standard residuals that we would need to further investigate. We notice that we have a few points we would neet to address that are bad leverages. We would need to invsetigate further to see how influential those points are.

4b) Identify your bad leverage points. How many and what are you planning to do to fix this problem.

```
leverages <- hatvalues(SLR3.1)
# h_ii > 2 * (p+1)/n
leverage_points <- which(leverages > 2 * mean(leverages))
outliers <- which(abs(rstandard(SLR3.1)) > 2)
bad_leverages <- intersect(leverage_points, outliers)
bad_leverages
```

```
##  [1]    83   397   478  1464  1551  1609  2259  2522  3175  4072  4366  4586
## [13]  4886  4952  5069  5755  6056  6142  6255  6611  7129  7488  7575  8101
## [25]  8297  9111  9582  9838  9850 10122 10905
```

```
length(bad_leverages)
```

```
## [1] 31
```

We plan on fixing this problem by deleting the rows of the observations that are bad leverages.

4c) Identify all good leverage points based on your latest MLR. Any comments?

```
good_leverages <- leverage_points[which(!(leverage_points %in% bad_leverages))]
length(good_leverages)
```

```
## [1] 1316
```

We have 1316 good leverages. This could mean that we are not predicting the wages of the players that have more extreme (lower or higher) Overall scores or international reputation very well. But this is a good sign it means they do not affect our model negatively.

4d) Report your summary statistics of the predicted response variable in both training and testing data sets. Any comments?

```
#FifaTrainNew
new_data = data.frame(FifaTrainNew$Overall2, FifaTrainNew$ClubRep2)
names(new_data) = c("Overall2", "ClubRep2")
predicts <- exp(predict(SLR3.1, new_data)) #%>% as.data.frame()
predicts <- ifelse(is.na(predicts), mean(predicts, na.rm=TRUE), predicts)
#predictions <- cbind(FifaTrainNew$Ob, FifaTrainNew$WageNew, predicts)
summary(predicts)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     528    3118    5432    9441    9441  414722
```

```
#For FifaNoY
new_data = data.frame(FifaNoY$Overall2,FifaNoY$ClubRep2)
names(new_data) = c("Overall2", "ClubRep2")
predicts <- exp(predict(SLR3.1, new_data)) #%>% as.data.frame()
predicts <- ifelse(is.na(predicts), mean(predicts, na.rm=TRUE), predicts)
#predictions <- cbind(FifaNoY$Ob, predicts)
summary(predicts)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## 8.862e+106 1.267e+186 7.260e+216        Inf 2.319e+250        Inf
```

```
## We notice that we have the same minimum wage predictions. Most of the values are very similar, excep
```

Q5) Apply the step function and regsubsets function in r on your latest MLR and use it to answer the following: Need library "leaps"

5a) Identify the optimal model or models based on R2 adj , AIC, BIC from the approach based on all possible subsets.

```
attach(FifaTrainNew)

MLR3.1 <- lm(log(WageNew) ~ 0 + I(Overall^2), data = FifaTrainNew)
MLR3.2 <- lm(log(WageNew) ~ 0 + ClubRep2, data = FifaTrainNew)
MLR3.3 <- lm(log(WageNew) ~ 0 + I(Overall^2) + ClubRep2, data = FifaTrainNew)

summary(MLR3.1)$adj.r.squared
```

```
## [1] 0.9827157
```

```r
summary(MLR3.2)$adj.r.squared
```

```
## [1] 0.9865176
```

```r
summary(MLR3.3)$adj.r.squared
```

```
## [1] 0.9932659
```

```r
extractAIC(MLR3.1, k=2)
```

```
## [1]    1.000 3125.844
```

```r
extractAIC(MLR3.2, k=2)
```

```
## [1]  7.00000 54.22318
```

```r
extractAIC(MLR3.3, k=2)
```

```
## [1]     8.000 -7811.401
```

```r
extractAIC(MLR3.1, k=log(length(WageNew)))
```

```
## [1]    1.000 3133.297
```

```r
extractAIC(MLR3.2, k=log(length(WageNew)))
```

```
## [1]   7.0000 106.3934
```

```r
extractAIC(MLR3.3, k=log(length(WageNew)))
```

```
## [1]     8.000 -7751.778
```

5b) Identify the optimal model or models based on AIC and BIC from the approach based on

```r
# backward selection.

backAIC <- step(SLR3,direction="backward", data=FifaTrainNew)
```

```
## Start:  AIC=-7482.52
## log(WageNew) ~ 0 + Overall2 + ClubRep2
##
##            Df Sum of Sq     RSS     AIC
## <none>                   5846.9 -7482.5
## - ClubRep2  7     930.1  6777.0 -5823.7
## - Overall2  1    5525.4 11372.3    54.2
```

```r
backBIC <- step(SLR3,direction="backward", data=FifaTrainNew, k=log(length(WageNew)))
```

```
## Start:  AIC=-7422.9
## log(WageNew) ~ 0 + Overall2 + ClubRep2
##
##            Df Sum of Sq     RSS     AIC
## <none>                   5846.9 -7422.9
## - ClubRep2  7     930.1  6777.0 -5816.2
## - Overall2  1    5525.4 11372.3   106.4
```

5c) Identify the optimal model or models based on AIC and BIC from the approach based on forward selection.

```
mint <- lm(log(WageNew) ~ 0 + 1, data=FifaTrainNew)

forardAIC <- step(mint, scope=list(lower = ~1, upper = ~I(Overall^2) + Overall2), direction="forward", c
```

```
## Start:  AIC=3998.97
## log(WageNew) ~ 0 + 1
##
##                 Df Sum of Sq     RSS     AIC
## + I(Overall^2)   1   10397.4  7042.3 -7556.4
## + Overall2       1    9998.3  7441.5 -6853.7
## <none>                        17439.7  3999.0
##
## Step:  AIC=-7556.41
## log(WageNew) ~ I(Overall^2)
##
##             Df Sum of Sq    RSS     AIC
## + Overall2   1    516.12 6526.2 -8524.5
## <none>                   7042.3 -7556.4
##
## Step:  AIC=-8524.48
## log(WageNew) ~ I(Overall^2) + Overall2
```

```
forwardBIC <- step(mint, scope=list(lower = ~1, upper = ~I(Overall^2) + Overall2), direction="forward",
```

```
## Start:  AIC=4006.42
## log(WageNew) ~ 0 + 1
##
##                 Df Sum of Sq     RSS     AIC
## + I(Overall^2)   1   10397.4  7042.3 -7541.5
## + Overall2       1    9998.3  7441.5 -6838.8
## <none>                        17439.7  4006.4
##
## Step:  AIC=-7541.51
## log(WageNew) ~ I(Overall^2)
##
##             Df Sum of Sq    RSS     AIC
## + Overall2   1    516.12 6526.2 -8502.1
## <none>                   7042.3 -7541.5
##
## Step:  AIC=-8502.12
## log(WageNew) ~ I(Overall^2) + Overall2
```

5d) Compare and contrast the models chosen in (A) (B) & (C). Check those which are similar and those which are different "maybe".

We would choose model MLR3.3 from part (5a) because it yields the highest R-Squared (Adjusted), and the AIC and BIC are the smaller differences, giving us the best model choice. MLR3.3 has the highest R-Squared (adjusted), and has the best AIC and BIC. We can see MLR3.1 and MLR3.2 have similar R-Squared (adjusted). As for AIC and BIC, MLR3.3 yields the best results for not overfitting nor underfitting.

5e) Recommend a final model. Give detailed reasons to support your choice on final model.

For our final model, I would recommend MLR3.3 because it yields the highest R-Squared (Adjusted), and the AIC and BIC are the smaller differences, giving us the best model choice. We can also see it satisfies the assumptions for the plot, and it works. It has undergone the necessary transformations as well to make it normal as possible.

5f) Interpret the regression coefficients in the final model. Is it necessary to be cautious about taking these results too literally?

$Y = B0 + B1x1 + B2x2 + B3x3 + B4x4 + B5x5 + B6x6 + B7x7 + B8x8$

```
summary(MLR3.3)
```

```
##
## Call:
## lm(formula = log(WageNew) ~ 0 + I(Overall^2) + ClubRep2, data = FifaTrainNew)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.1595 -0.3458  0.0389  0.4236  2.2726
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## I(Overall^2)      8.575e-04  8.049e-06  106.53   <2e-16 ***
## ClubRep2high      5.914e+00  7.646e-02   77.34   <2e-16 ***
## ClubRep2low       4.650e+00  3.779e-02  123.03   <2e-16 ***
## ClubRep2mid       5.338e+00  4.914e-02  108.62   <2e-16 ***
## ClubRep2mid-high  5.131e+00  5.427e-02   94.54   <2e-16 ***
## ClubRep2mid-low   4.629e+00  3.796e-02  121.95   <2e-16 ***
## ClubRep2very high 5.879e+00  7.014e-02   83.82   <2e-16 ***
## ClubRep2very low  4.614e+00  3.545e-02  130.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7082 on 11324 degrees of freedom
##   (1413 observations deleted due to missingness)
## Multiple R-squared:  0.9933, Adjusted R-squared:  0.9933
## F-statistic: 2.089e+05 on 8 and 11324 DF,  p-value: < 2.2e-16
```

For every 1 unit of increase in log(WageNew), we see, on average, an increase of ("some number" = slope) in ("predictor"). For every unit increase in `Overall^2`, `log(WageNew)` is expected to increase by 8.603e-04, on average. A player with a `ClubRep2` of high is expected to increase `log(WageNew)` by 5.898, on average. A player with a `ClubRep2` of low is expected to increase `log(WageNew)` by 4.637, on average. A player with a `ClubRep2` of mid is expected to increase `log(WageNew)` by 5.323, on average. A player with a `ClubRep2` of mid-high is expected to increase `log(WageNew)` by 5.115, on average. A player with a `ClubRep2` of mid-low is expected to increase `log(WageNew)` by 4.617, on average. A player with a `ClubRep2` of NA is expected to increase `log(WageNew)` by 4.660, on average. A player with a `ClubRep2` of very high is expected to increase `log(WageNew)` by 5.862, on average. A player with a `ClubRep2` of very low is expected to increase `log(WageNew)` by 4.602, on average.