Project

#Intro steps  
library(leaps)  
library(MPV)

##   
## Attaching package: 'MPV'

## The following object is masked from 'package:datasets':  
##   
## stackloss

project <- read.csv("C:\\Users\\Bridget\\Documents\\MATH651\\data\\OnlineData.csv")  
names(project)

## [1] "County.Names"   
## [2] "Percent.for.Trump"   
## [3] "Median.Household.Income"   
## [4] "Percent.Third.Party..Johnson..McMullin...Stein."  
## [5] "Median.Age..years."   
## [6] "Percent.White..One.Race."   
## [7] "Unemployment.rate..Population.16.years.and.over"  
## [8] "Bachelors.Degree..25.years.or.older"   
## [9] "Graduate.Professional.Degree..25.years.or.older"  
## [10] "Bachelors.or.Higher..25.years.or.older"   
## [11] "Trump.Won"

head(project)

## County.Names Percent.for.Trump Median.Household.Income  
## 1 Accomack County 54.5 39389  
## 2 Albemarle County 34.3 67958  
## 3 Alleghany County 66.7 45183  
## 4 Amelia County 66.9 55870  
## 5 Amherst County 63.6 44765  
## 6 Appomattox County 71.8 47927  
## Percent.Third.Party..Johnson..McMullin...Stein. Median.Age..years.  
## 1 2.8 44.9  
## 2 6.4 38.5  
## 3 3.6 46.8  
## 4 2.6 43.3  
## 5 3.5 43.1  
## 6 2.7 42.5  
## Percent.White..One.Race. Unemployment.rate..Population.16.years.and.over  
## 1 68.6 7.5  
## 2 81.6 4.4  
## 3 93.3 6.2  
## 4 72.5 6.0  
## 5 76.8 7.6  
## 6 77.1 7.3  
## Bachelors.Degree..25.years.or.older  
## 1 10.5  
## 2 25.9  
## 3 10.2  
## 4 11.5  
## 5 12.6  
## 6 9.4  
## Graduate.Professional.Degree..25.years.or.older  
## 1 7.8  
## 2 26.2  
## 3 6.7  
## 4 4.2  
## 5 5.7  
## 6 5.8  
## Bachelors.or.Higher..25.years.or.older Trump.Won  
## 1 18.3 1  
## 2 52.1 0  
## 3 16.9 1  
## 4 15.7 1  
## 5 18.3 1  
## 6 15.2 1

names(project)<-c("County", "Percent.Trump", "Median.Income", "Percent.Third", "Median.Age", "Percent.White", "Unemployment.Rate", "Bachelors", "Graduate", "Bachelors.Higher", "Trump.Won")  
  
  
  
  
  
  
  
  
#run a Multiple Regression with all variables.   
Test.lm<-lm(Percent.Trump~Percent.Third+Median.Income+Median.Age+Percent.White+Unemployment.Rate+Bachelors+Graduate+Bachelors.Higher, data=project)  
#The summary shows that Bachelors.Higher has NA values. This means that Bachelors.Higher is a linear combination of other variables.   
summary(Test.lm)

##   
## Call:  
## lm(formula = Percent.Trump ~ Percent.Third + Median.Income +   
## Median.Age + Percent.White + Unemployment.Rate + Bachelors +   
## Graduate + Bachelors.Higher, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -16.1285 -3.3974 0.0057 3.1486 16.4768   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.450e+00 6.915e+00 -0.788 0.43204   
## Percent.Third -1.064e+00 3.915e-01 -2.718 0.00749 \*\*   
## Median.Income 2.003e-04 3.903e-05 5.133 1.06e-06 \*\*\*  
## Median.Age 2.885e-01 9.907e-02 2.912 0.00426 \*\*   
## Percent.White 7.150e-01 3.139e-02 22.775 < 2e-16 \*\*\*  
## Unemployment.Rate 4.509e-01 2.488e-01 1.812 0.07231 .   
## Bachelors -1.594e-01 1.873e-01 -0.851 0.39635   
## Graduate -1.299e+00 1.443e-01 -8.998 3.12e-15 \*\*\*  
## Bachelors.Higher NA NA NA NA   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.031 on 125 degrees of freedom  
## Multiple R-squared: 0.914, Adjusted R-squared: 0.9091   
## F-statistic: 189.7 on 7 and 125 DF, p-value: < 2.2e-16

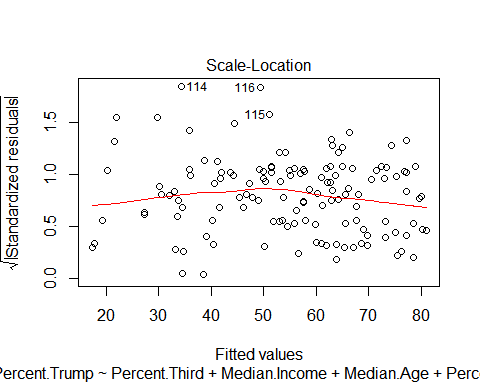
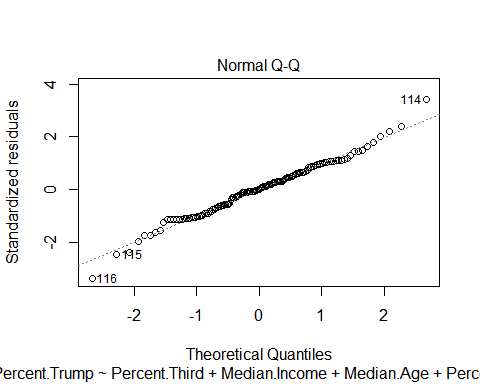
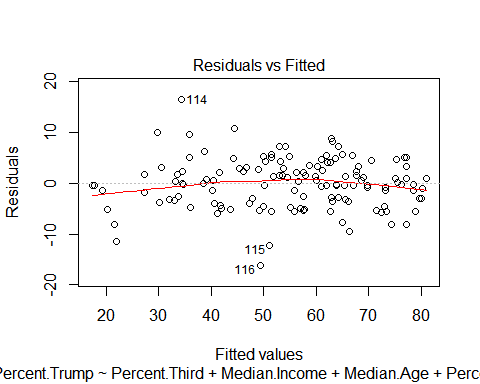
#Examine multicollinearity among the variables, notice that Bachelors.Higher is very highly correlated with Bachelors and Graduate  
cor(project[,-1])

## Percent.Trump Median.Income Percent.Third Median.Age  
## Percent.Trump 1.00000000 -0.25593349 -0.3518481 0.57729154  
## Median.Income -0.25593349 1.00000000 0.4050190 -0.15414165  
## Percent.Third -0.35184812 0.40501901 1.0000000 -0.53312207  
## Median.Age 0.57729154 -0.15414165 -0.5331221 1.00000000  
## Percent.White 0.73297024 0.02284767 0.1267255 0.25431284  
## Unemployment.Rate -0.09812962 -0.53356702 -0.4171118 -0.05348841  
## Bachelors -0.49156722 0.77796753 0.6195283 -0.43466447  
## Graduate -0.58093091 0.68130643 0.5827093 -0.44875211  
## Bachelors.Higher -0.55256836 0.74849068 0.6173090 -0.45423666  
## Trump.Won 0.83320525 -0.08394048 -0.2827862 0.57876349  
## Percent.White Unemployment.Rate Bachelors Graduate  
## Percent.Trump 0.732970238 -0.09812962 -0.49156722 -0.58093091  
## Median.Income 0.022847672 -0.53356702 0.77796753 0.68130643  
## Percent.Third 0.126725513 -0.41711177 0.61952835 0.58270933  
## Median.Age 0.254312843 -0.05348841 -0.43466447 -0.44875211  
## Percent.White 1.000000000 -0.47659732 0.01371791 -0.01600707  
## Unemployment.Rate -0.476597322 1.00000000 -0.49870304 -0.44986810  
## Bachelors 0.013717908 -0.49870304 1.00000000 0.89290360  
## Graduate -0.016007072 -0.44986810 0.89290360 1.00000000  
## Bachelors.Higher -0.001632855 -0.48675594 0.97118873 0.97447724  
## Trump.Won 0.596213551 -0.17859473 -0.34311355 -0.43583425  
## Bachelors.Higher Trump.Won  
## Percent.Trump -0.552568356 0.83320525  
## Median.Income 0.748490676 -0.08394048  
## Percent.Third 0.617309006 -0.28278617  
## Median.Age -0.454236657 0.57876349  
## Percent.White -0.001632855 0.59621355  
## Unemployment.Rate -0.486755936 -0.17859473  
## Bachelors 0.971188727 -0.34311355  
## Graduate 0.974477240 -0.43583425  
## Bachelors.Higher 1.000000000 -0.40175356  
## Trump.Won -0.401753556 1.00000000

#run a new Regression without Bachelors.Higher  
NewTest.lm<-lm(Percent.Trump~Percent.Third+Median.Income+Median.Age+Percent.White+Unemployment.Rate+Bachelors + Graduate, data=project)  
#Note that this summary gives different values, but that the adjusted R2 is still over .9  
summary(NewTest.lm)

##   
## Call:  
## lm(formula = Percent.Trump ~ Percent.Third + Median.Income +   
## Median.Age + Percent.White + Unemployment.Rate + Bachelors +   
## Graduate, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -16.1285 -3.3974 0.0057 3.1486 16.4768   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.450e+00 6.915e+00 -0.788 0.43204   
## Percent.Third -1.064e+00 3.915e-01 -2.718 0.00749 \*\*   
## Median.Income 2.003e-04 3.903e-05 5.133 1.06e-06 \*\*\*  
## Median.Age 2.885e-01 9.907e-02 2.912 0.00426 \*\*   
## Percent.White 7.150e-01 3.139e-02 22.775 < 2e-16 \*\*\*  
## Unemployment.Rate 4.509e-01 2.488e-01 1.812 0.07231 .   
## Bachelors -1.594e-01 1.873e-01 -0.851 0.39635   
## Graduate -1.299e+00 1.443e-01 -8.998 3.12e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.031 on 125 degrees of freedom  
## Multiple R-squared: 0.914, Adjusted R-squared: 0.9091   
## F-statistic: 189.7 on 7 and 125 DF, p-value: < 2.2e-16

#Check the residual assumptions- are the independent, constant variance, mean 0 and normal variance? Yes!  
plot(NewTest.lm)



#Now, using adjusted R2 and Mallow's Cp statistic from the leaps package, compare subsets of the models to see if there are any better to examine. We do not include the Bachelors.Higher variable because, as it is a linear combination of other variables, it will not provide any new information.  
#Comparing Regressions  
project.leaps<-leaps(y=project$Percent.Trump, x=project[,c(3:9)])  
project.leaps.cp<-leaps(y=project$Percent.Trump, x=project[,c(3:9)])  
project.leaps.r2<-leaps(y=project$Percent.Trump, x=project[,c(3:9)], method="adjr2")  
project.leapsfull<-cbind(project.leaps.cp$which, project.leaps.r2$adjr2, project.leaps.cp$Cp)  
colnames(project.leapsfull)<-c("Percent.Third", "Median.Income", "Median.Age", "Percent.White", "Unemployment.Rate", "Bachelors", "Graduate","adjR2", "Cp")  
  
  
  
  
#When comparing Adjusted R2 values, the higher values are best. Because adjusted R2 penalizes for inclusion of variables, it only increases with more variables if these are significant. When comparing Cp values, Cp should be near the number of parameters used. For example, if looking at a regression model with 4 explanatory variables, Cp should be near 4.   
#The top 3 regression models for adjR2 and Cp are the same as follows, from the best down:  
#Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate  
#Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Bachelors + Graduate  
#Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Graduate  
#project.leapsfull  
#seeing what the top 6 adjR2 values are  
leaps.frame<-as.data.frame(project.leapsfull)  
tail(leaps.frame[order(leaps.frame$adjR2),])

## Percent.Third Median.Income Median.Age Percent.White Unemployment.Rate  
## 5.8 1 0 1 1 1  
## 6.4 1 0 1 1 1  
## 6.5 1 1 1 1 0  
## 5.9 1 1 1 1 0  
## 7 1 1 1 1 1  
## 6.6 1 1 1 1 1  
## Bachelors Graduate adjR2 Cp  
## 5.8 0 1 0.9038570 13.371497  
## 6.4 1 1 0.9045231 13.389832  
## 6.5 1 1 0.9074834 9.285036  
## 5.9 0 1 0.9075652 8.188900  
## 7 1 1 0.9091314 8.000000  
## 6.6 0 1 0.9093301 6.724347

#Look at PRESS and AIC for the top 3 models found. For both of these, smaller values means better models. For both of these, as well, Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate is the best model.  
  
  
  
  
PRESS(lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate, data=project))

## [1] 3691.668

PRESS(lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Bachelors + Graduate, data=project))

## [1] 3704.112

PRESS(lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Graduate, data=project))

## [1] 3742.957

AIC(lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate, data=project))

## [1] 815.7276

AIC(lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Bachelors + Graduate, data=project))

## [1] 816.9591

AIC(lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Graduate, data=project))

## [1] 817.343

#Now that we have the best model to use, we examine it  
NewTest2.lm<-lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate, data=project)  
#Note that Unemployment.Rate is statistically significant only with a p-value of 0.1, not the standard 0.5. However, the statistics above (adjR2, Cp, PRESS, AIC) show that the model that does not include this model is not as good.   
summary(NewTest2.lm)

##   
## Call:  
## lm(formula = Percent.Trump ~ Percent.Third + Median.Income +   
## Median.Age + Percent.White + Unemployment.Rate + Graduate,   
## data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -16.079 -3.411 0.127 3.043 15.993   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.415e+00 6.814e+00 -0.942 0.34824   
## Percent.Third -1.129e+00 3.836e-01 -2.944 0.00386 \*\*   
## Median.Income 1.833e-04 3.348e-05 5.476 2.27e-07 \*\*\*  
## Median.Age 3.019e-01 9.771e-02 3.089 0.00247 \*\*   
## Percent.White 7.145e-01 3.135e-02 22.788 < 2e-16 \*\*\*  
## Unemployment.Rate 4.624e-01 2.481e-01 1.863 0.06474 .   
## Graduate -1.383e+00 1.049e-01 -13.186 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.026 on 126 degrees of freedom  
## Multiple R-squared: 0.9135, Adjusted R-squared: 0.9093   
## F-statistic: 221.6 on 6 and 126 DF, p-value: < 2.2e-16

#Check the residual assumptions- are the independent, constant variance, mean 0 and normal variance? Yes!  
#plot(NewTest2.lm)  
library(lmtest)

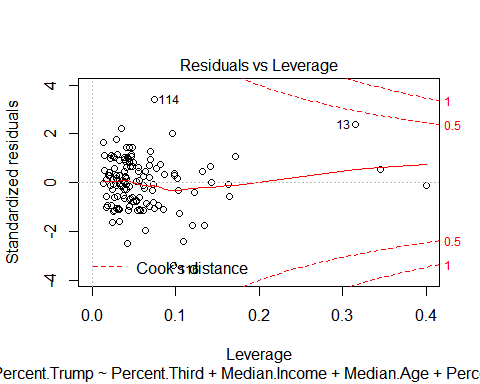
## Warning: package 'lmtest' was built under R version 3.3.2

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.3.2

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric



#non-constant variance from the bptest  
bptest(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate, data=project,studentize = FALSE)

##   
## Breusch-Pagan test  
##   
## data: Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate  
## BP = 24.553, df = 6, p-value = 0.000413

library(nortest)

## Warning: package 'nortest' was built under R version 3.3.2

# Lilliefors test shows the residuals are normally distributed  
lillie.test(NewTest2.lm$resid)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: NewTest2.lm$resid  
## D = 0.056598, p-value = 0.3719

library(stats)  
shapiro.test(NewTest2.lm$resid)

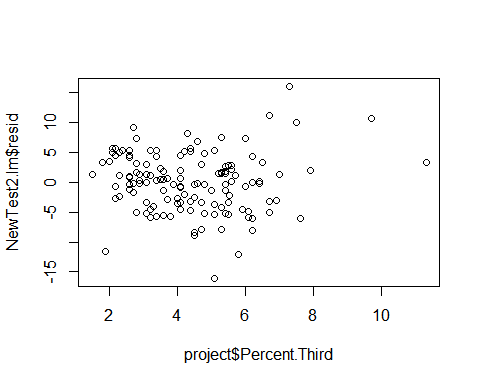
##   
## Shapiro-Wilk normality test  
##   
## data: NewTest2.lm$resid  
## W = 0.98984, p-value = 0.4422

library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:MPV':  
##   
## cement

#boxcox(NewTest2.lm)   
#boxcox shows that the best value for lambda is close to 1  
  
  
plot(project$Percent.Third,NewTest2.lm$resid)



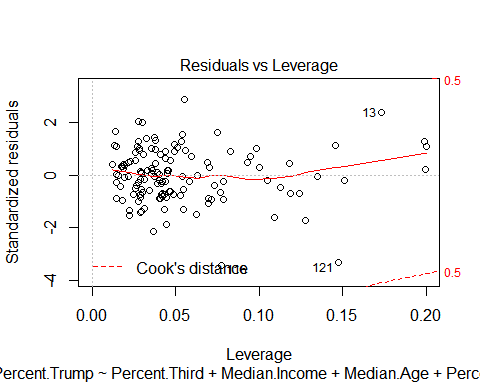
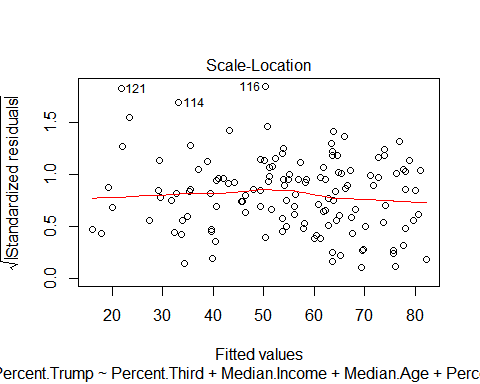
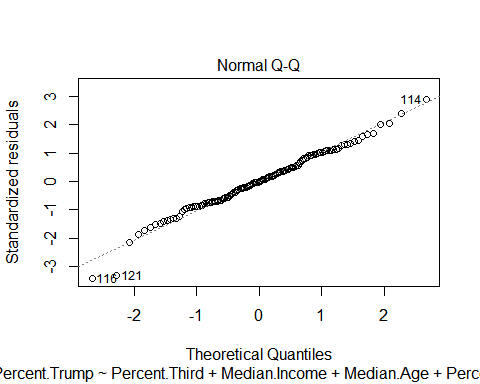
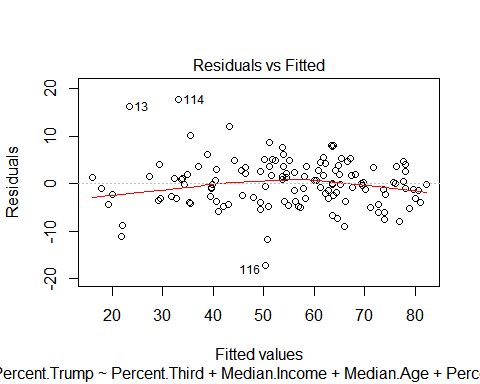
#the plot of the residuals against Percent Third seems to show a megaphone effect.Maybe do a Weighted Least Squared on Percent Third  
  
  
residuals.lm <- lm(abs(NewTest2.lm$resid) ~ project$Percent.Third)  
  
  
NewTest3.wls <- lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate, data=project,weights=1/((residuals.lm$fit)^2))  
#summary(NewTest3.wls)  
#plot(NewTest3.wls)  
bptest(NewTest3.wls)

##   
## studentized Breusch-Pagan test  
##   
## data: NewTest3.wls  
## BP = 17.778, df = 6, p-value = 0.006812

residuals2.lm <- lm(abs(NewTest2.lm$resid) ~ project$Percent.Third)  
  
  
NewTest4.wls <- lm(Percent.Trump ~ Percent.Third + Median.Income + Median.Age + Percent.White + Unemployment.Rate + Graduate, data=project,weights=1/((residuals2.lm$fit)^2))  
summary(NewTest4.wls)

##   
## Call:  
## lm(formula = Percent.Trump ~ Percent.Third + Median.Income +   
## Median.Age + Percent.White + Unemployment.Rate + Graduate,   
## data = project, weights = 1/((residuals2.lm$fit)^2))  
##   
## Weighted Residuals:  
## Min 1Q Median 3Q Max   
## -4.2274 -0.8480 -0.0137 0.8719 3.6049   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.213e+00 6.392e+00 0.346 0.7298   
## Percent.Third -1.979e+00 4.410e-01 -4.488 1.60e-05 \*\*\*  
## Median.Income 1.852e-04 3.503e-05 5.286 5.34e-07 \*\*\*  
## Median.Age 1.947e-01 9.788e-02 1.989 0.0488 \*   
## Percent.White 7.266e-01 2.700e-02 26.914 < 2e-16 \*\*\*  
## Unemployment.Rate 2.280e-01 2.154e-01 1.058 0.2919   
## Graduate -1.355e+00 1.116e-01 -12.140 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.29 on 126 degrees of freedom  
## Multiple R-squared: 0.9238, Adjusted R-squared: 0.9202   
## F-statistic: 254.5 on 6 and 126 DF, p-value: < 2.2e-16

plot(NewTest4.wls)



bptest(NewTest4.wls)

##   
## studentized Breusch-Pagan test  
##   
## data: NewTest4.wls  
## BP = 17.778, df = 6, p-value = 0.006812

#Set up the new variables for the Box Cox transformation.   
library(caret)

## Warning: package 'caret' was built under R version 3.3.2

## Loading required package: lattice

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.3.2

library(e1071)

## Warning: package 'e1071' was built under R version 3.3.2

dist\_trump=predict(BoxCoxTrans(project$Percent.Trump), project$Percent.Trump)  
BoxCoxTrans(project$Percent.Trump)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 10.60 43.70 57.20 54.98 67.80 82.30   
##   
## Largest/Smallest: 7.76   
## Sample Skewness: -0.559   
##   
## Estimated Lambda: 1.5

dist\_third=predict(BoxCoxTrans(project$Percent.Third), project$Percent.Third)  
BoxCoxTrans(project$Percent.Third)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.500 3.100 4.200 4.381 5.400 11.300   
##   
## Largest/Smallest: 7.53   
## Sample Skewness: 0.848   
##   
## Estimated Lambda: 0.2

dist\_income=predict(BoxCoxTrans(project$Median.Income), project$Median.Income)  
BoxCoxTrans(project$Median.Income)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 27750 38480 47800 53050 63160 124000   
##   
## Largest/Smallest: 4.47   
## Sample Skewness: 1.36   
##   
## Estimated Lambda: -0.8

dist\_age=predict(BoxCoxTrans(project$Median.Age), project$Median.Age)  
BoxCoxTrans(project$Median.Age)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 21.6 37.8 42.5 41.1 44.8 58.6   
##   
## Largest/Smallest: 2.71   
## Sample Skewness: -0.708   
##   
## Estimated Lambda: 2

dist\_white=predict(BoxCoxTrans(project$Percent.White), project$Percent.White)  
BoxCoxTrans(project$Percent.White)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 18.10 64.80 77.50 75.53 90.20 99.80   
##   
## Largest/Smallest: 5.51   
## Sample Skewness: -0.736   
##   
## Estimated Lambda: 2

dist\_employ=predict(BoxCoxTrans(project$Unemployment.Rate), project$Unemployment.Rate)  
BoxCoxTrans(project$Unemployment.Rate)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.900 5.900 7.500 7.803 9.500 17.200   
##   
## Largest/Smallest: 9.05   
## Sample Skewness: 0.736   
##   
## Estimated Lambda: 0.3

dist\_grad=predict(BoxCoxTrans(project$Graduate), project$Graduate)  
BoxCoxTrans(project$Graduate)

## Box-Cox Transformation  
##   
## 133 data points used to estimate Lambda  
##   
## Input data summary:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.700 5.300 7.200 9.668 12.200 43.700   
##   
## Largest/Smallest: 16.2   
## Sample Skewness: 2.19   
##   
## Estimated Lambda: -0.3

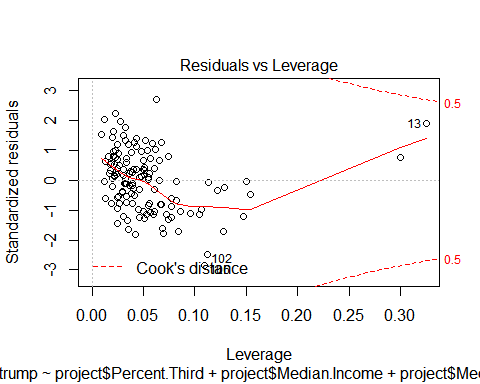
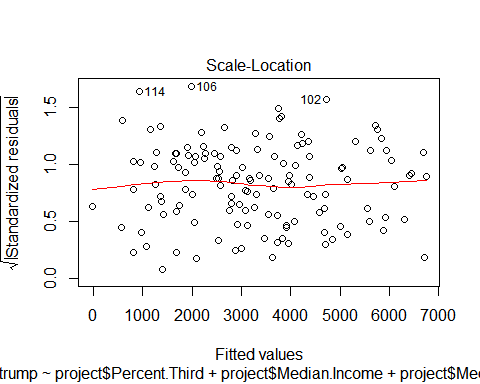
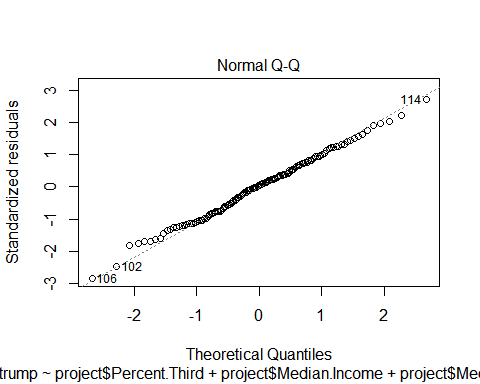
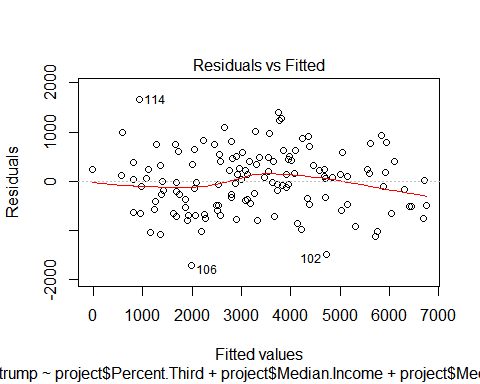
#Create new linear model with the transformed variables.   
NewTest5.lm<-lm(dist\_trump~dist\_third+dist\_income+dist\_age+dist\_white+dist\_employ+dist\_grad)  
  
  
  
  
  
  
#Note, however, that complicated models are rarely the best ones. We use the BoxCoxTrans above to look at the lambda values (the last value of the output) and see what is going on. Then I tried mixing and matching to find the model that used the least amount of transforming but still had constant variance and normality.   
  
  
#Create new linear model with fewer transformed variables  
NewTest6.lm<-lm(I(project$Percent.Trump^2) ~ project$Percent.Third + project$Median.Income + project$Median.Age +I(project$Percent.White^2) +project$Unemployment.Rate+ I(project$Graduate^-.5))  
#Run the BP test, which results in a BP stat of 9.2492, a df of 6, and a p-value of 0.16.  
bptest(NewTest6.lm, studentize = F)

##   
## Breusch-Pagan test  
##   
## data: NewTest6.lm  
## BP = 9.2492, df = 6, p-value = 0.16

#Run Lillie test, wich results in a D = 0.050511, p-value = 0.5563  
lillie.test(NewTest6.lm$residuals)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: NewTest6.lm$residuals  
## D = 0.050511, p-value = 0.5563

trump<-project$Percent.Trump^2  
white<-project$Percent.White^2  
grad<-project$Graduate^-.5  
NewTest7.lm<-lm(trump ~ project$Percent.Third + project$Median.Income + project$Median.Age +white +project$Unemployment.Rate+ grad)  
plot(NewTest7.lm)



summary(NewTest6.lm)

##   
## Call:  
## lm(formula = I(project$Percent.Trump^2) ~ project$Percent.Third +   
## project$Median.Income + project$Median.Age + I(project$Percent.White^2) +   
## project$Unemployment.Rate + I(project$Graduate^-0.5))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1700.48 -479.51 31.04 410.83 1659.91   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.018e+03 8.737e+02 -4.599 1.02e-05 \*\*\*  
## project$Percent.Third -1.465e+02 5.184e+01 -2.825 0.00549 \*\*   
## project$Median.Income 1.007e-02 4.052e-03 2.485 0.01428 \*   
## project$Median.Age 3.671e+01 1.196e+01 3.068 0.00264 \*\*   
## I(project$Percent.White^2) 5.472e-01 2.813e-02 19.454 < 2e-16 \*\*\*  
## project$Unemployment.Rate 6.299e+01 3.083e+01 2.044 0.04309 \*   
## I(project$Graduate^-0.5) 5.828e+03 9.182e+02 6.347 3.62e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 634.1 on 126 degrees of freedom  
## Multiple R-squared: 0.8686, Adjusted R-squared: 0.8624   
## F-statistic: 138.9 on 6 and 126 DF, p-value: < 2.2e-16

#Now that we found a good model, we check if there are any undue influencers to this model.   
influence.measures(NewTest6.lm)

## Influence measures of  
## lm(formula = I(project$Percent.Trump^2) ~ project$Percent.Third + project$Median.Income + project$Median.Age + I(project$Percent.White^2) + project$Unemployment.Rate + I(project$Graduate^-0.5)) :  
##   
## dfb.1\_ dfb.p.P. dfb.p.M.I dfb.p.M.A dfb.I..P dfb.p.U. dfb.I..G  
## 1 0.043553 -3.53e-02 -4.41e-02 0.011574 -2.62e-02 -0.029074 -0.038857  
## 2 -0.103695 -1.03e-02 1.02e-01 -0.038544 1.13e-02 0.085339 0.176397  
## 3 -0.026778 4.09e-02 5.09e-02 -0.030812 -5.45e-02 0.034079 0.043951  
## 4 0.057920 -6.90e-02 4.87e-02 -0.076489 -8.12e-02 -0.152883 0.141327  
## 5 0.023104 -1.75e-02 -1.93e-02 -0.004104 -1.69e-02 -0.032103 0.010453  
## 6 0.203967 -2.35e-01 -4.49e-02 -0.119358 -2.68e-02 -0.164592 -0.004276  
## 7 -0.097283 1.02e-01 -2.49e-01 0.145172 2.08e-02 0.085570 0.032436  
## 8 -0.107394 1.74e-01 -1.96e-02 0.088527 1.57e-02 -0.038885 0.080788  
## 9 0.002750 -2.64e-04 -9.23e-03 0.008358 -1.57e-03 -0.010756 -0.003498  
## 10 -0.025012 -1.77e-02 -9.15e-03 0.075177 1.05e-01 0.053240 -0.097174  
## 11 -0.043952 -4.09e-02 5.82e-02 -0.052726 1.74e-01 0.043915 0.084227  
## 12 -0.004546 -3.81e-03 7.31e-03 0.005568 1.43e-02 -0.004543 0.000625  
## 13 -0.768477 1.25e+00 -9.64e-02 0.616282 -6.23e-01 0.110185 0.576925  
## 14 0.063641 1.39e-02 -5.26e-02 0.035824 -1.28e-01 -0.084074 -0.080736  
## 15 0.067685 -3.56e-02 -8.28e-02 0.039957 7.32e-02 -0.014466 -0.199748  
## 16 0.043636 9.39e-03 -6.56e-02 -0.009023 -1.63e-02 -0.108541 0.050429  
## 17 0.044094 -2.23e-02 -4.71e-02 -0.004686 -9.93e-03 -0.065997 -0.016488  
## 18 -0.001262 1.20e-02 3.08e-03 0.010330 -2.81e-02 0.005565 -0.015524  
## 19 0.096111 -8.90e-02 -2.09e-02 -0.147624 2.12e-01 0.006249 -0.087641  
## 20 0.200355 -1.18e-01 -1.91e-01 -0.006755 -1.62e-01 -0.225258 -0.047485  
## 21 -0.005085 7.06e-03 2.99e-02 0.001566 -9.75e-03 0.016682 -0.012142  
## 22 0.123591 -4.79e-02 -1.06e-01 -0.119510 -1.08e-01 -0.101748 0.060367  
## 23 0.079247 -5.10e-02 -7.03e-02 0.024454 -5.23e-02 0.038408 -0.175088  
## 24 -0.000603 -5.15e-03 4.29e-02 -0.034177 4.67e-03 -0.007359 0.038205  
## 25 -0.001179 2.76e-04 -7.56e-05 0.001969 -4.30e-03 0.001506 0.002138  
## 26 -0.015869 8.13e-02 -1.40e-02 0.071485 -1.44e-01 -0.040896 -0.020031  
## 27 -0.007266 3.07e-02 -2.67e-02 0.019858 6.70e-03 -0.009281 -0.016023  
## 28 -0.007046 3.91e-03 6.49e-03 -0.002025 1.46e-02 0.011082 -0.002319  
## 29 0.018541 2.31e-02 -1.44e-01 0.010874 2.28e-02 -0.013850 0.000620  
## 30 0.024884 -3.48e-03 -5.00e-02 -0.003842 -2.37e-02 -0.016903 -0.005068  
## 31 0.077028 -7.35e-02 -2.84e-03 -0.031456 -1.08e-01 0.009440 -0.076055  
## 32 0.011931 -1.99e-02 -7.61e-03 -0.021695 -1.09e-03 0.000288 0.010815  
## 33 0.012439 -1.71e-02 -1.60e-02 0.004108 1.86e-02 -0.005243 -0.017300  
## 34 -0.006970 1.67e-03 1.57e-02 -0.004973 2.40e-02 0.010131 -0.001511  
## 35 0.010575 -9.71e-03 -7.63e-03 -0.006520 1.21e-02 -0.010860 -0.002686  
## 36 -0.032904 3.98e-02 1.54e-02 0.026358 2.08e-02 -0.010141 0.018523  
## 37 -0.043230 -3.50e-04 6.25e-02 0.099638 -1.64e-02 0.004208 -0.054105  
## 38 0.191139 -1.52e-01 -5.65e-02 -0.034773 -9.29e-02 -0.038608 -0.250417  
## 39 0.010833 -1.36e-02 -5.98e-03 -0.002289 -3.94e-03 -0.002619 -0.009219  
## 40 -0.116353 9.01e-02 8.51e-03 0.092603 2.34e-01 0.195593 -0.173106  
## 41 0.057444 -5.49e-02 -5.73e-02 0.015868 -7.00e-02 -0.040273 -0.024228  
## 42 -0.022174 5.70e-03 5.34e-02 0.014790 3.21e-02 0.002377 -0.008861  
## 43 0.016415 4.48e-03 -1.36e-02 0.009615 -3.19e-02 -0.003592 -0.026802  
## 44 0.041866 -1.01e-02 -1.69e-02 -0.019821 -9.09e-03 -0.052693 -0.027646  
## 45 0.083504 -1.48e-01 2.59e-01 -0.420297 8.43e-02 0.289755 0.056436  
## 46 -0.036758 -7.80e-04 4.15e-02 0.059145 -1.11e-02 0.046419 -0.032926  
## 47 0.027248 -1.68e-02 -1.19e-02 -0.065906 -3.35e-03 -0.023287 0.055199  
## 48 0.023758 -1.45e-02 -1.52e-02 -0.001278 -3.05e-02 -0.044096 0.009691  
## 49 -0.052131 -2.14e-02 2.39e-01 -0.078107 8.54e-02 0.114793 0.011772  
## 50 0.064908 -5.20e-02 6.20e-02 -0.085763 -5.07e-02 -0.138406 0.099878  
## 51 0.002112 6.41e-04 6.77e-03 -0.017027 4.39e-03 -0.003538 0.014029  
## 52 -0.004690 -6.23e-02 2.54e-02 -0.052389 1.24e-01 0.084803 0.001510  
## 53 0.021077 6.28e-03 -1.05e-01 0.026225 1.38e-03 -0.005156 -0.032074  
## 54 0.002048 -1.46e-03 -1.28e-03 -0.001582 -4.18e-05 -0.001069 -0.000894  
## 55 0.023834 -1.05e-02 -1.80e-02 -0.005969 -5.10e-02 -0.053363 0.034227  
## 56 -0.002315 2.79e-03 -8.49e-03 0.014629 5.27e-03 0.000906 -0.012041  
## 57 0.016012 -3.14e-02 -4.55e-02 0.101320 -1.48e-03 -0.037980 -0.100695  
## 58 0.074592 -6.40e-02 -7.83e-02 0.035364 -8.30e-02 -0.066866 -0.053007  
## 59 0.000275 -5.56e-03 -2.42e-02 0.052750 -1.49e-02 -0.010179 -0.039153  
## 60 -0.004706 -2.59e-04 4.39e-03 0.004370 -2.17e-03 0.001420 0.002813  
## 61 0.015789 -5.17e-02 1.41e-01 -0.209141 1.97e-02 -0.000739 0.180982  
## 62 -0.047307 -2.15e-03 8.81e-02 0.004517 2.88e-02 0.030785 0.034687  
## 63 -0.022736 5.47e-05 1.17e-01 -0.125850 1.09e-01 0.019727 0.085323  
## 64 -0.065768 1.78e-02 -3.64e-02 0.196549 -5.41e-02 0.048886 -0.100874  
## 65 0.000215 -8.61e-04 1.68e-03 -0.004847 -2.05e-02 -0.002259 0.021112  
## 66 0.006466 3.44e-03 -9.27e-03 -0.003704 -1.16e-02 -0.016320 0.007949  
## 67 0.099690 1.67e-02 -9.08e-02 0.014956 -2.02e-01 -0.176920 -0.022009  
## 68 0.017408 -5.00e-03 -5.48e-03 -0.006452 -1.99e-02 -0.018484 -0.008249  
## 69 0.043086 -7.01e-02 -9.07e-03 -0.018900 -2.71e-02 -0.052697 0.034162  
## 70 -0.066961 4.89e-02 1.57e-01 0.015068 -8.08e-03 -0.117519 0.134807  
## 71 0.264923 -1.24e-01 -1.81e-01 -0.169073 -6.03e-02 -0.112440 -0.113241  
## 72 -0.009059 -4.91e-02 1.34e-01 -0.087675 -1.75e-02 0.077638 0.069730  
## 73 -0.000118 2.68e-04 -1.06e-03 0.000592 2.15e-04 0.000118 -0.000341  
## 74 0.036895 -1.95e-02 -7.94e-03 -0.014268 -5.50e-02 -0.016084 -0.028420  
## 75 -0.156457 1.67e-01 1.63e-01 -0.139934 -2.87e-02 0.169526 0.262671  
## 76 0.005858 1.59e-04 -7.83e-03 -0.000207 2.50e-03 -0.005066 -0.009052  
## 77 -0.001676 1.48e-03 3.76e-03 -0.007714 -9.55e-03 0.002072 0.012654  
## 78 -0.054549 1.56e-02 1.04e-01 -0.061705 -3.06e-04 0.099943 0.071799  
## 79 0.004066 4.45e-03 -1.12e-02 -0.005577 4.73e-02 -0.001527 -0.013380  
## 80 0.005497 -1.91e-02 -4.38e-03 -0.013664 4.01e-02 0.010108 -0.001871  
## 81 -0.000595 -1.87e-03 1.60e-03 -0.002501 4.48e-03 0.000138 0.003573  
## 82 0.002325 1.22e-03 -7.85e-03 0.001781 9.54e-03 -0.006569 -0.002212  
## 83 0.016179 -2.64e-02 -1.53e-02 -0.012985 4.26e-02 -0.001306 -0.010507  
## 84 0.017943 -9.80e-03 -2.55e-03 0.000111 -6.50e-02 -0.064124 0.045971  
## 85 -0.000134 -8.94e-03 9.41e-02 -0.054124 3.45e-03 -0.002332 0.037960  
## 86 -0.020163 -2.32e-02 2.09e-01 -0.091078 4.41e-03 0.012806 0.071340  
## 87 0.055252 -3.26e-02 -3.61e-02 -0.064829 9.58e-02 -0.015208 -0.051876  
## 88 -0.011728 6.13e-03 4.34e-03 0.008744 1.35e-02 0.016665 -0.008697  
## 89 0.037411 -1.13e-01 1.57e-02 -0.103789 1.65e-01 0.021299 0.041204  
## 90 0.035047 -1.37e-02 -4.70e-02 0.015938 -5.74e-02 -0.034495 -0.029200  
## 91 0.070270 -1.05e-01 -7.09e-02 -0.001030 1.33e-01 0.030787 -0.139581  
## 92 0.019614 2.76e-02 -5.58e-04 -0.067869 7.53e-03 -0.066295 0.062905  
## 93 0.056298 -1.79e-01 5.10e-02 -0.156895 2.45e-01 0.140822 -0.033655  
## 94 -0.011127 -6.36e-03 5.98e-03 -0.002006 3.97e-02 0.029896 -0.005382  
## 95 -0.084300 1.11e-01 7.95e-02 0.081786 -1.41e-02 0.036164 -0.025930  
## 96 -0.108852 8.44e-02 -3.77e-02 0.045678 6.28e-02 0.070052 0.086131  
## 97 -0.001982 -1.33e-02 -1.34e-02 -0.000276 5.23e-02 0.050363 -0.033699  
## 98 -0.277045 4.73e-01 -6.21e-02 0.155786 -2.31e-02 0.079855 0.179357  
## 99 -0.265011 8.36e-02 1.99e-01 0.140133 5.85e-02 0.162604 0.138459  
## 100 0.035652 -2.91e-02 5.20e-02 -0.044339 -4.48e-02 -0.008549 0.000958  
## 101 0.034956 -1.45e-02 -1.92e-03 -0.045148 1.49e-02 -0.040999 0.030173  
## 102 0.112556 -3.29e-01 1.31e-02 0.075020 2.83e-01 0.509744 -0.730817  
## 103 0.013122 1.29e-02 2.21e-02 -0.046840 1.94e-02 -0.112659 0.073270  
## 104 0.012289 3.08e-02 1.56e-02 -0.053752 1.00e-02 -0.132409 0.098229  
## 105 0.106250 -8.48e-02 -1.73e-01 -0.057280 3.48e-02 -0.050840 -0.021072  
## 106 0.357305 -1.23e-01 -7.95e-01 -0.025565 -1.74e-01 -0.323146 -0.025157  
## 107 -0.014569 1.84e-02 8.44e-03 0.008375 1.91e-02 -0.034476 0.018175  
## 108 0.004294 2.03e-03 -3.41e-03 -0.006274 2.24e-03 0.007783 -0.006694  
## 109 0.036323 -3.80e-02 1.42e-02 0.095610 -1.06e-01 0.066144 -0.237628  
## 110 0.002229 -7.67e-05 -8.39e-04 -0.001267 -5.10e-03 0.001159 -0.000903  
## 111 -0.198102 -2.09e-02 1.60e-01 0.249569 -7.75e-02 0.067513 0.034883  
## 112 0.126413 -1.14e-01 -1.07e-01 0.043870 4.10e-02 -0.153954 -0.173689  
## 113 -0.380547 1.15e-01 2.54e-01 0.348638 -6.22e-02 0.199459 0.108006  
## 114 0.091749 3.25e-01 -2.73e-01 -0.143966 -9.50e-02 0.105230 -0.059550  
## 115 0.005523 -2.14e-02 -1.12e-01 0.111307 -3.65e-02 -0.053493 -0.051449  
## 116 -0.175739 7.31e-02 -5.38e-02 0.221437 1.28e-01 0.265291 -0.155050  
## 117 0.037061 -2.66e-02 6.46e-02 -0.104648 4.55e-02 -0.148344 0.103674  
## 118 0.018723 3.15e-02 -1.09e-02 -0.029372 -5.58e-02 0.004546 0.013452  
## 119 0.014851 -1.13e-03 -7.76e-03 -0.017317 -9.19e-03 0.013732 -0.009828  
## 120 -0.000420 -5.66e-03 4.25e-04 -0.006018 1.88e-02 0.016463 -0.006095  
## 121 -0.047351 1.18e-01 9.18e-03 -0.003350 1.58e-01 -0.233555 0.104162  
## 122 -0.124385 6.32e-02 1.38e-01 0.087420 1.34e-01 0.067203 -0.030233  
## 123 -0.008602 6.17e-04 2.18e-03 0.008555 1.87e-02 0.000990 -0.004333  
## 124 -0.028716 -1.47e-02 3.40e-02 0.047649 -3.03e-02 -0.014002 0.014610  
## 125 0.019870 2.74e-02 -4.40e-02 0.004812 -4.20e-02 0.020469 -0.036638  
## 126 -0.018178 -2.48e-02 4.68e-02 -0.010732 3.73e-02 0.017756 0.012405  
## 127 0.003468 3.41e-03 -9.04e-03 0.002570 3.65e-03 -0.003286 -0.007015  
## 128 -0.023051 -1.04e-01 1.92e-01 -0.114482 2.15e-02 0.031915 0.122743  
## 129 0.014255 -1.87e-02 2.48e-02 -0.006111 -3.90e-02 0.008948 -0.010809  
## 130 0.026273 6.07e-02 1.87e-02 -0.047975 -7.45e-02 -0.059629 0.046115  
## 131 0.023601 -6.59e-02 1.85e-02 -0.004909 2.20e-03 -0.003897 -0.028112  
## 132 -0.291747 1.60e-01 1.42e-01 0.235829 -2.85e-02 0.110395 0.135130  
## 133 -0.052797 -1.52e-02 7.72e-02 0.019474 -9.82e-03 0.013039 0.054006  
## dffit cov.r cook.d hat inf  
## 1 0.07199 1.094 7.46e-04 0.04005   
## 2 -0.31940 0.942 1.44e-02 0.03585   
## 3 -0.14902 1.033 3.18e-03 0.02497   
## 4 0.24236 1.056 8.39e-03 0.05409   
## 5 0.07124 1.048 7.29e-04 0.01275   
## 6 0.34345 0.817 1.63e-02 0.02258 \*  
## 7 -0.48964 0.953 3.37e-02 0.06981   
## 8 0.24321 1.029 8.43e-03 0.04364   
## 9 0.01968 1.108 5.58e-05 0.04590   
## 10 0.20653 0.961 6.04e-03 0.02006   
## 11 0.26761 0.962 1.01e-02 0.03059   
## 12 0.03238 1.081 1.51e-04 0.02380   
## 13 1.34385 1.273 2.52e-01 0.32482 \*  
## 14 -0.21216 1.093 6.45e-03 0.06649   
## 15 -0.27047 1.039 1.04e-02 0.05319   
## 16 0.22685 0.847 7.17e-03 0.01200   
## 17 -0.08571 1.089 1.06e-03 0.03858   
## 18 -0.05418 1.097 4.22e-04 0.03994   
## 19 -0.28786 1.046 1.18e-02 0.05969   
## 20 0.31006 1.011 1.37e-02 0.05168   
## 21 0.07502 1.059 8.09e-04 0.01847   
## 22 -0.25895 0.988 9.52e-03 0.03487   
## 23 -0.22039 1.091 6.96e-03 0.06704   
## 24 0.07378 1.059 7.82e-04 0.01816   
## 25 0.01014 1.079 1.48e-05 0.02002   
## 26 -0.20150 1.075 5.81e-03 0.05437   
## 27 -0.06671 1.081 6.40e-04 0.02963   
## 28 -0.02024 1.093 5.90e-05 0.03336   
## 29 -0.20324 1.123 5.93e-03 0.08245   
## 30 -0.06635 1.103 6.33e-04 0.04597   
## 31 -0.23391 0.963 7.75e-03 0.02498   
## 32 -0.06925 1.050 6.89e-04 0.01323   
## 33 0.04611 1.070 3.06e-04 0.01832   
## 34 0.03729 1.078 2.00e-04 0.02254   
## 35 0.02637 1.089 1.00e-04 0.03065   
## 36 0.10207 1.035 1.49e-03 0.01559   
## 37 0.18890 1.046 5.10e-03 0.03855   
## 38 -0.35682 1.053 1.81e-02 0.07734   
## 39 -0.01855 1.100 4.95e-05 0.03933   
## 40 -0.36418 1.091 1.89e-02 0.09612   
## 41 0.13604 1.062 2.65e-03 0.03353   
## 42 0.11695 1.048 1.96e-03 0.02298   
## 43 0.06206 1.075 5.54e-04 0.02477   
## 44 -0.10051 1.076 1.45e-03 0.03351   
## 45 -0.65778 1.030 6.09e-02 0.12801   
## 46 0.11731 1.041 1.97e-03 0.02055   
## 47 -0.10807 1.087 1.68e-03 0.04154   
## 48 0.05297 1.117 4.04e-04 0.05551   
## 49 0.32726 0.919 1.50e-02 0.03294   
## 50 0.23710 0.930 7.92e-03 0.02049   
## 51 -0.02155 1.193 6.69e-05 0.11352 \*  
## 52 0.17364 1.104 4.33e-03 0.06542   
## 53 -0.12707 1.197 2.32e-03 0.12228 \*  
## 54 -0.00370 1.070 1.98e-06 0.01219   
## 55 0.08082 1.106 9.40e-04 0.05043   
## 56 0.02803 1.073 1.13e-04 0.01700   
## 57 0.17668 1.088 4.48e-03 0.05637   
## 58 0.15219 1.086 3.32e-03 0.05003   
## 59 0.07348 1.108 7.77e-04 0.05079   
## 60 -0.00914 1.151 1.20e-05 0.08168   
## 61 -0.29568 1.017 1.24e-02 0.05079   
## 62 0.10957 1.057 1.72e-03 0.02511   
## 63 -0.21038 1.051 6.33e-03 0.04513   
## 64 0.22825 1.101 7.46e-03 0.07386   
## 65 0.04355 1.094 2.73e-04 0.03603   
## 66 -0.02200 1.092 6.97e-05 0.03245   
## 67 -0.26362 1.051 9.92e-03 0.05680   
## 68 -0.03606 1.099 1.87e-04 0.03942   
## 69 0.15141 1.025 3.28e-03 0.02298   
## 70 0.33249 0.874 1.54e-02 0.02714   
## 71 0.30438 1.034 1.32e-02 0.05849   
## 72 0.23064 1.010 7.57e-03 0.03494   
## 73 -0.00153 1.131 3.36e-07 0.06531   
## 74 -0.10611 1.044 1.61e-03 0.01912   
## 75 -0.43678 0.984 2.69e-02 0.06873   
## 76 -0.01779 1.091 4.56e-05 0.03125   
## 77 -0.02824 1.076 1.15e-04 0.01961   
## 78 -0.18769 1.082 5.05e-03 0.05531   
## 79 0.07750 1.064 8.63e-04 0.02163   
## 80 0.05518 1.093 4.38e-04 0.03729   
## 81 0.00812 1.118 9.49e-06 0.05462   
## 82 0.02392 1.079 8.24e-05 0.02132   
## 83 0.06570 1.082 6.21e-04 0.03049   
## 84 0.09804 1.103 1.38e-03 0.05126   
## 85 0.14299 1.038 2.93e-03 0.02530   
## 86 0.26572 1.061 1.01e-02 0.06144   
## 87 -0.15601 1.065 3.49e-03 0.03941   
## 88 -0.02180 1.244 6.84e-05 0.15023 \*  
## 89 0.26118 1.010 9.70e-03 0.04124   
## 90 -0.08692 1.068 1.09e-03 0.02560   
## 91 0.21293 1.056 6.48e-03 0.04769   
## 92 -0.13538 1.061 2.63e-03 0.03308   
## 93 0.34166 1.036 1.66e-02 0.06732   
## 94 0.05021 1.097 3.63e-04 0.03939   
## 95 0.24614 0.990 8.60e-03 0.03286   
## 96 -0.23840 1.047 8.11e-03 0.04970   
## 97 0.07972 1.097 9.14e-04 0.04360   
## 98 0.49862 1.461 3.56e-02 0.29954 \*  
## 99 -0.32029 1.088 1.46e-02 0.08564   
## 100 0.13941 1.026 2.78e-03 0.02083   
## 101 0.15030 0.933 3.19e-03 0.00923   
## 102 -0.89726 0.839 1.10e-01 0.11178 \*  
## 103 -0.17492 1.123 4.39e-03 0.07761   
## 104 -0.19509 1.236 5.47e-03 0.15417 \*  
## 105 -0.26701 1.058 1.02e-02 0.06066   
## 106 -1.02397 0.747 1.41e-01 0.10917 \*  
## 107 -0.07997 1.120 9.20e-04 0.06079   
## 108 0.01907 1.119 5.24e-05 0.05511   
## 109 -0.38603 0.918 2.09e-02 0.04261   
## 110 0.00913 1.093 1.20e-05 0.03282   
## 111 -0.39477 1.095 2.22e-02 0.10414   
## 112 -0.32032 1.059 1.46e-02 0.07212   
## 113 -0.50251 1.142 3.59e-02 0.14682   
## 114 0.71802 0.742 6.99e-02 0.06276 \*  
## 115 -0.21376 1.006 6.50e-03 0.03059   
## 116 -0.37202 1.037 1.97e-02 0.07411   
## 117 -0.24927 1.100 8.89e-03 0.07754   
## 118 0.11314 1.072 1.84e-03 0.03368   
## 119 0.04611 1.112 3.06e-04 0.05098   
## 120 0.02584 1.109 9.61e-05 0.04757   
## 121 -0.52125 0.979 3.82e-02 0.08392   
## 122 0.30038 0.989 1.28e-02 0.04329   
## 123 -0.03075 1.095 1.36e-04 0.03609   
## 124 -0.09344 1.210 1.26e-03 0.12900 \*  
## 125 0.10259 1.116 1.51e-03 0.06100   
## 126 -0.07127 1.079 7.30e-04 0.02952   
## 127 0.01818 1.080 4.76e-05 0.02198   
## 128 -0.27628 1.014 1.09e-02 0.04567   
## 129 0.08445 1.070 1.02e-03 0.02655   
## 130 0.18949 1.000 5.11e-03 0.02437   
## 131 -0.09231 1.063 1.22e-03 0.02433   
## 132 -0.33460 1.123 1.60e-02 0.10665   
## 133 -0.12726 1.051 2.32e-03 0.02645

#influencers FOR NEW TEST 6 (numbers 13, 45, 51, 88, 98, 102, 104, 106, 113, 114, 117, 121, 124)- Appomattox County, Brunswick County, Lancaster County, Loudon County, Sussex County, Buena Vista City, Covington City, Emporia City, Falls Church City, Lynchburg City, Radford City, Interesting to note, fairly evenly split.   
project[c(6, 13, 51, 53, 88, 98, 102, 104, 106, 114, 124), ]

## County Percent.Trump Median.Income Percent.Third Median.Age  
## 6 Appomattox County 71.8 47927 2.7 42.5  
## 13 Brunswick County 39.8 37028 9.7 41.6  
## 51 Lancaster County 53.3 49456 3.2 55.3  
## 53 Loudoun County 38.6 123966 6.0 35.1  
## 88 Sussex County 40.8 36972 2.1 37.8  
## 98 Buena Vista city 59.8 32789 11.3 37.2  
## 102 Covington city 56.9 38176 4.5 44.1  
## 104 Emporia city 33.5 30240 1.8 43.2  
## 106 Falls Church city 17.3 120500 7.0 36.7  
## 114 Lynchburg city 50.9 39391 7.3 29.0  
## 124 Radford city 43.7 30284 7.9 21.9  
## Percent.White Unemployment.Rate Bachelors Graduate Bachelors.Higher  
## 6 77.1 7.3 9.4 5.8 15.2  
## 13 41.7 9.8 8.1 5.2 13.3  
## 51 69.7 8.3 16.2 12.4 28.6  
## 53 68.5 4.3 35.0 23.0 58.0  
## 88 39.2 6.2 5.2 3.7 8.9  
## 98 89.9 7.8 7.2 6.8 14.0  
## 102 82.3 5.1 6.0 3.0 9.0  
## 104 31.0 17.2 7.8 7.2 15.0  
## 106 77.5 5.0 31.4 43.7 75.1  
## 114 64.8 9.6 19.9 12.4 32.3  
## 124 85.7 9.4 20.0 14.9 34.9  
## Trump.Won  
## 6 1  
## 13 0  
## 51 1  
## 53 0  
## 88 0  
## 98 1  
## 102 1  
## 104 0  
## 106 0  
## 114 1  
## 124 0

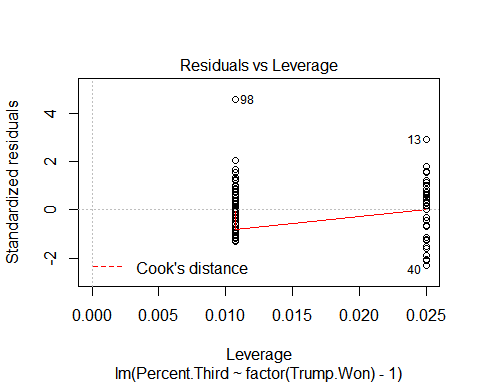
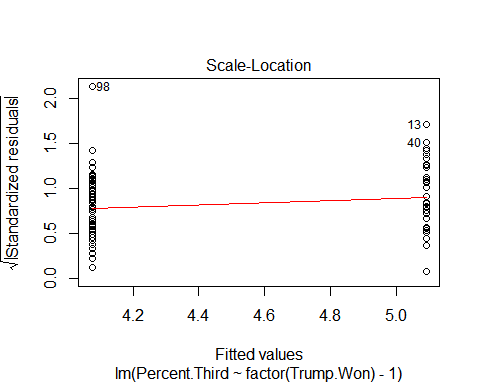
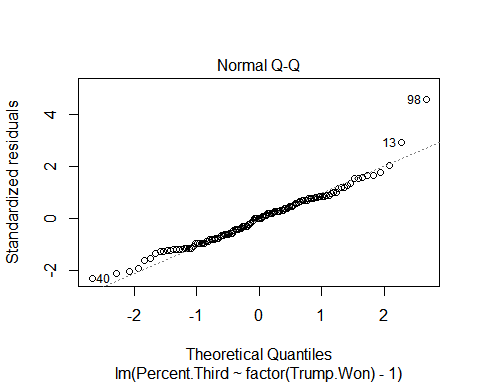
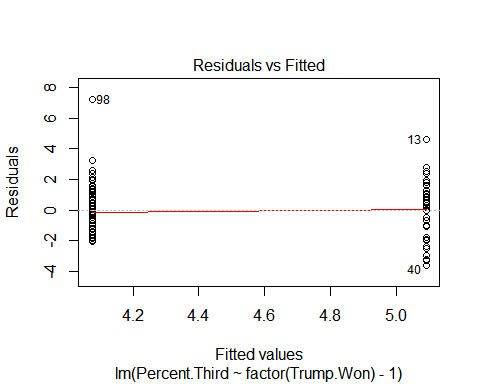
###########################################################################  
  
  
  
  
#Here, we compare the factor means of each variable, with the factors being 1= Trump won or 0=Clinton won.   
library(car)  
  
  
####### Third.Percent  
  
  
Third.lm<-lm(Percent.Third ~ factor(Trump.Won)-1, data=project)  
#meets the constant variance assumption!  
leveneTest(Percent.Third ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 2.7709 0.09838 .  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#calculate the F value for alpha = 0.05  
qf(0.95,1,131)

## [1] 3.913428

#Third.lm actually meets the constant variance assumption with the Brown-Forsythe test  
#mostly normal, good!  
plot(Third.lm)



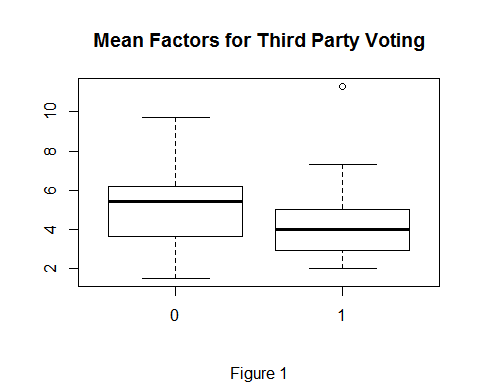
summary(Third.lm)

##   
## Call:  
## lm(formula = Percent.Third ~ factor(Trump.Won) - 1, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.5908 -1.1753 0.0247 1.0247 7.2247   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## factor(Trump.Won)0 5.0907 0.2516 20.23 <2e-16 \*\*\*  
## factor(Trump.Won)1 4.0753 0.1650 24.69 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.592 on 131 degrees of freedom  
## Multiple R-squared: 0.8861, Adjusted R-squared: 0.8843   
## F-statistic: 509.5 on 2 and 131 DF, p-value: < 2.2e-16

anova(Third.lm)

## Analysis of Variance Table  
##   
## Response: Percent.Third  
## Df Sum Sq Mean Sq F value Pr(>F)   
## factor(Trump.Won) 2 2581.16 1290.58 509.49 < 2.2e-16 \*\*\*  
## Residuals 131 331.83 2.53   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

boxplot(Percent.Third ~ factor(Trump.Won), data=project, sub="Figure 1", main="Mean Factors for Third Party Voting")



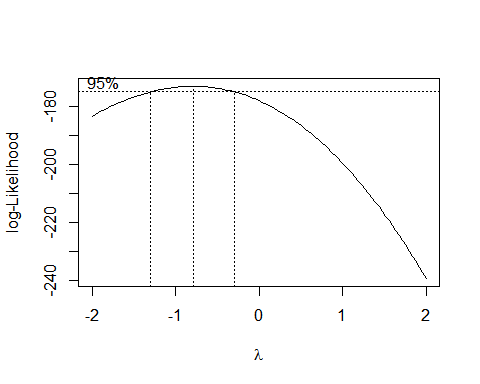
Third.aov<-aov(Percent.Third ~ factor(Trump.Won), data=project)  
#Different  
TukeyHSD(Third.aov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percent.Third ~ factor(Trump.Won), data = project)  
##   
## $`factor(Trump.Won)`  
## diff lwr upr p adj  
## 1-0 -1.015481 -1.610812 -0.4201506 0.0009736

####### Median.Income  
  
  
Income.lm <- lm(Median.Income ~ factor(Trump.Won)-1,data=project)  
#median income does not meet the constant variance assumption from Brown Forsythe test  
leveneTest(Median.Income ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 4.8988 0.02861 \*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

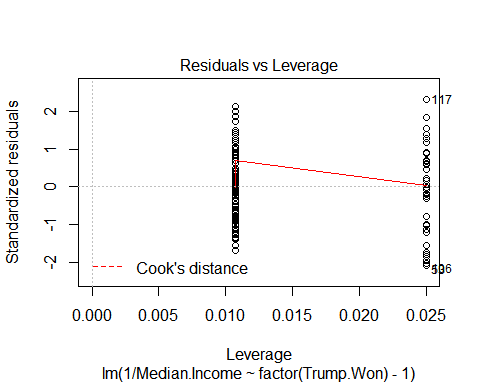
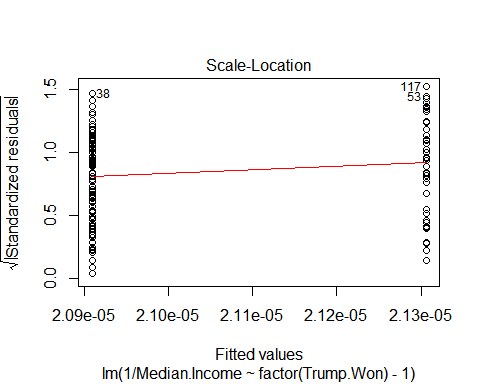
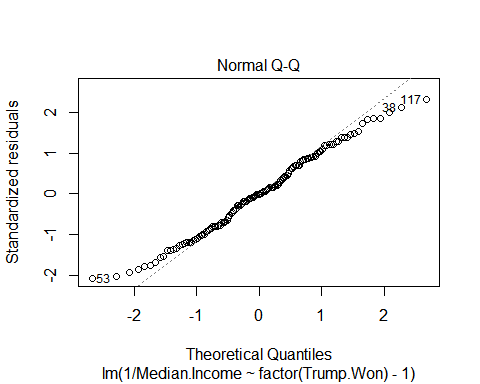
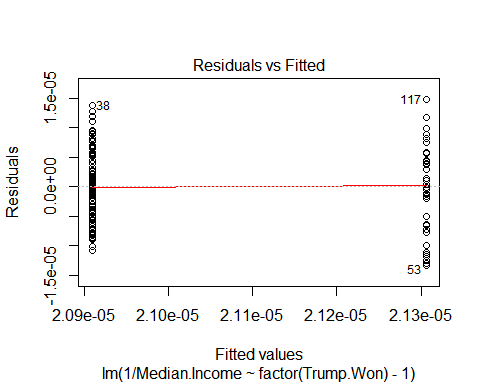
boxcox(Income.lm)



Income2.lm <- lm(1/Median.Income ~ factor(Trump.Won)-1,data=project)  
#New model meets constant variance assumption!  
leveneTest(1/Median.Income ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 3.4563 0.06525 .  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#mostly normal, good!  
plot(Income2.lm)



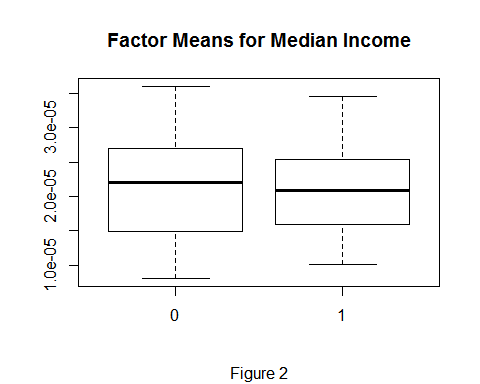
summary(Income2.lm)

##   
## Call:  
## lm(formula = 1/Median.Income ~ factor(Trump.Won) - 1, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.324e-05 -5.076e-06 -4.440e-08 5.075e-06 1.474e-05   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## factor(Trump.Won)0 2.131e-05 1.018e-06 20.93 <2e-16 \*\*\*  
## factor(Trump.Won)1 2.091e-05 6.676e-07 31.32 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.438e-06 on 131 degrees of freedom  
## Multiple R-squared: 0.9155, Adjusted R-squared: 0.9142   
## F-statistic: 709.5 on 2 and 131 DF, p-value: < 2.2e-16

anova(Income2.lm)

## Analysis of Variance Table  
##   
## Response: 1/Median.Income  
## Df Sum Sq Mean Sq F value Pr(>F)   
## factor(Trump.Won) 2 5.8818e-08 2.9409e-08 709.5 < 2.2e-16 \*\*\*  
## Residuals 131 5.4300e-09 4.1500e-11   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

boxplot(1/Median.Income ~ factor(Trump.Won),data=project, sub="Figure 2", main="Factor Means for Median Income")



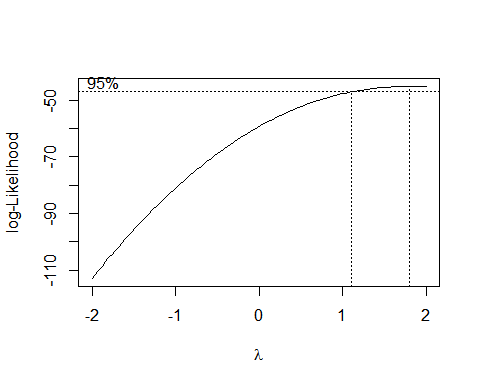
Income2.aov<-aov(1/Median.Income ~ factor(Trump.Won),data=project)  
#The same!  
TukeyHSD(Income2.aov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = 1/Median.Income ~ factor(Trump.Won), data = project)  
##   
## $`factor(Trump.Won)`  
## diff lwr upr p adj  
## 1-0 -3.966333e-07 -2.80486e-06 2.011593e-06 0.7450845

####### Median.Age  
  
  
Age.lm <- lm(Median.Age~ factor(Trump.Won)-1,data=project)  
#median age does not meet the constant variance assumption from Brown Forsythe test  
leveneTest(Median.Age ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 10.025 0.001921 \*\*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

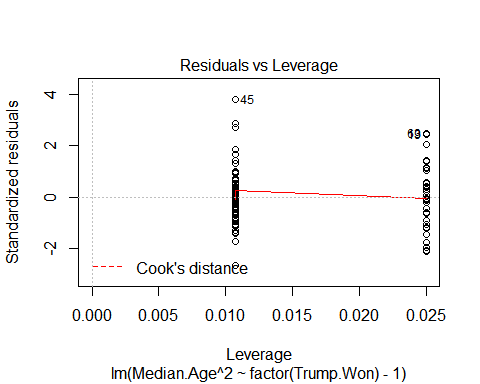
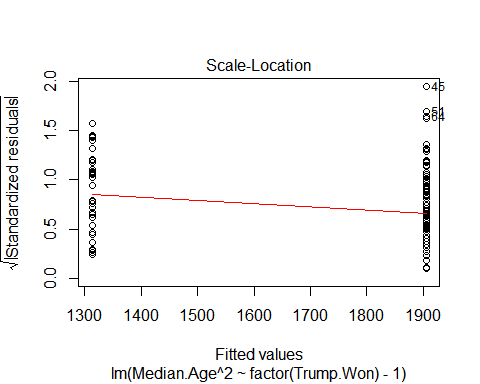
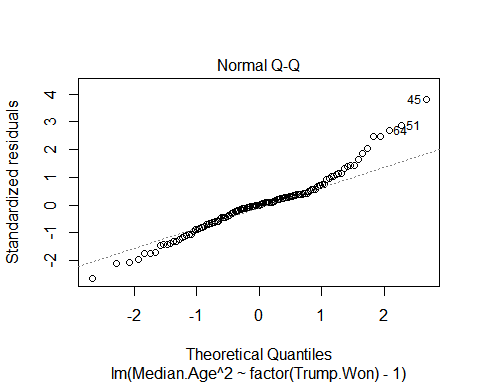
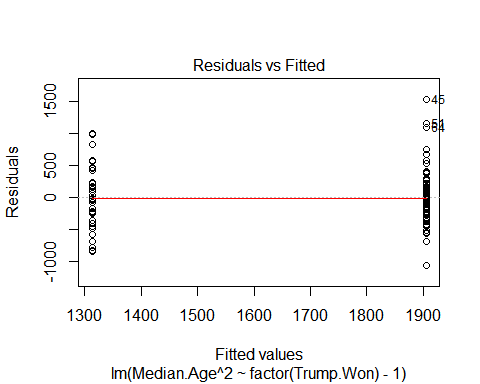
boxcox(Age.lm)



Age2.lm <- lm(Median.Age^2 ~ factor(Trump.Won)-1,data=project)  
#New model meets the constant variance assumption!  
leveneTest(Median.Age^2 ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 3.3534 0.06934 .  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#mostly normal??  
plot(Age2.lm)



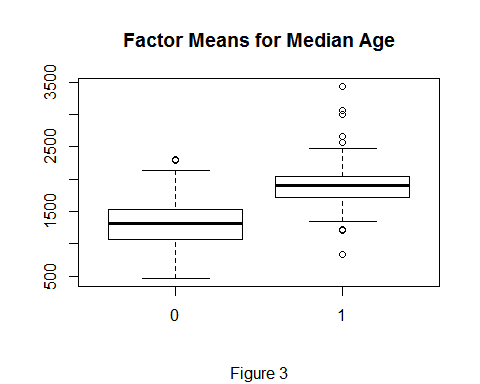
summary(Age2.lm)

##   
## Call:  
## lm(formula = Median.Age^2 ~ factor(Trump.Won) - 1, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1064.49 -240.85 -4.53 153.96 1528.47   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## factor(Trump.Won)0 1312.92 63.93 20.54 <2e-16 \*\*\*  
## factor(Trump.Won)1 1905.49 41.92 45.45 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 404.3 on 131 degrees of freedom  
## Multiple R-squared: 0.95, Adjusted R-squared: 0.9492   
## F-statistic: 1244 on 2 and 131 DF, p-value: < 2.2e-16

anova(Age2.lm)

## Analysis of Variance Table  
##   
## Response: Median.Age^2  
## Df Sum Sq Mean Sq F value Pr(>F)   
## factor(Trump.Won) 2 406624652 203312326 1243.8 < 2.2e-16 \*\*\*  
## Residuals 131 21413323 163460   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

boxplot(Median.Age^2 ~ factor(Trump.Won),data=project, sub="Figure 3", main="Factor Means for Median Age")



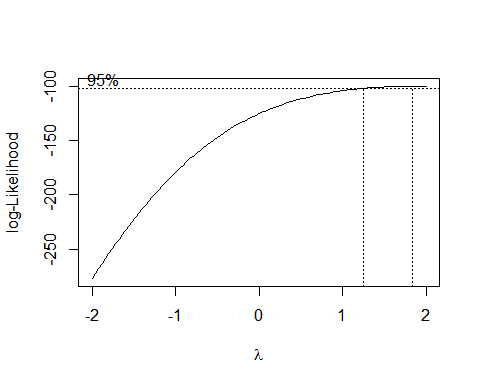
Age2.aov<-aov(Median.Age^2 ~ factor(Trump.Won),data=project)  
#Different  
TukeyHSD(Age2.aov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Median.Age^2 ~ factor(Trump.Won), data = project)  
##   
## $`factor(Trump.Won)`  
## diff lwr upr p adj  
## 1-0 592.5673 441.3368 743.7978 0

####### Percent.White  
  
  
Race.lm <- lm(Percent.White~ factor(Trump.Won)-1,data=project)  
#percent white does not meet the constant variance assumption from Brown Forsythe test  
leveneTest(Percent.White ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 6.4886 0.01201 \*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

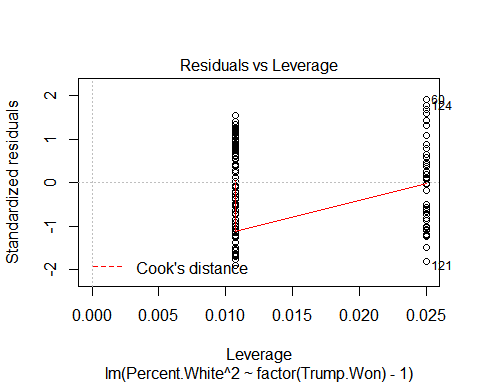
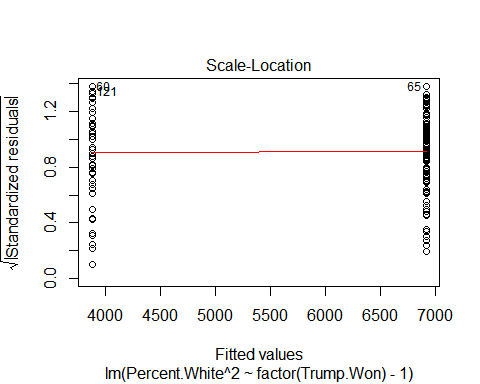
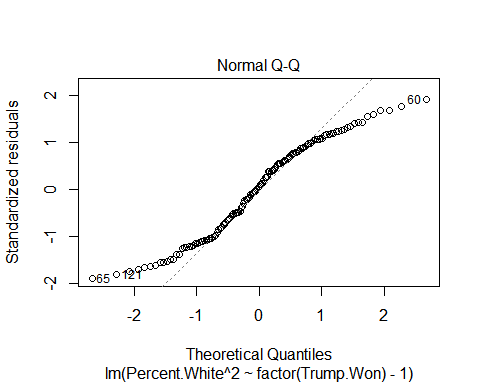
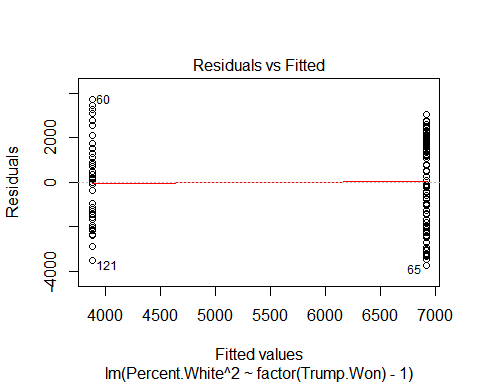
boxcox(Race.lm)



Race2.lm <- lm(Percent.White^2 ~ factor(Trump.Won)-1,data=project)  
#new model meets constant variance assumption!  
leveneTest(Percent.White^2 ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)  
## group 1 0.0268 0.8703  
## 131

#pretty bad normality.......  
plot(Race2.lm)



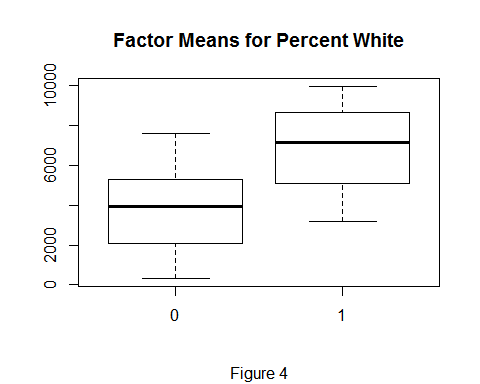
summary(Race2.lm)

##   
## Call:  
## lm(formula = Percent.White^2 ~ factor(Trump.Won) - 1, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3735.4 -1818.4 119.4 1695.4 3729.0   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## factor(Trump.Won)0 3874.9 312.6 12.40 <2e-16 \*\*\*  
## factor(Trump.Won)1 6916.4 205.0 33.74 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1977 on 131 degrees of freedom  
## Multiple R-squared: 0.9079, Adjusted R-squared: 0.9065   
## F-statistic: 646 on 2 and 131 DF, p-value: < 2.2e-16

anova(Race2.lm)

## Analysis of Variance Table  
##   
## Response: Percent.White^2  
## Df Sum Sq Mean Sq F value Pr(>F)   
## factor(Trump.Won) 2 5049401718 2524700859 646.03 < 2.2e-16 \*\*\*  
## Residuals 131 511954226 3908048   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

boxplot(Percent.White^2 ~ factor(Trump.Won),data=project, sub="Figure 4", main="Factor Means for Percent White")



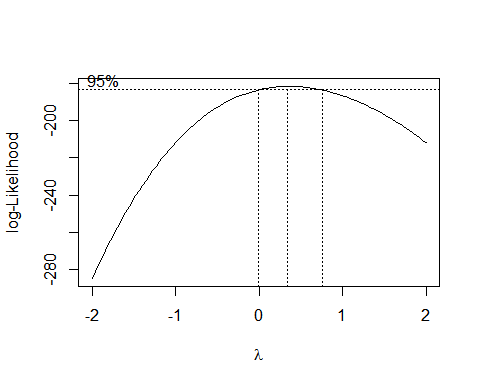
Race2.aov<-aov(Percent.White^2 ~ factor(Trump.Won),data=project)  
#Different  
TukeyHSD(Race2.aov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Percent.White^2 ~ factor(Trump.Won), data = project)  
##   
## $`factor(Trump.Won)`  
## diff lwr upr p adj  
## 1-0 3041.521 2302.064 3780.978 0

####### Unemployment.Rate  
  
  
Unemployment.lm <- lm(Unemployment.Rate ~ factor(Trump.Won)-1, data=project)  
#unemployment rate does not meet the constant variance assumption from Brown Forsythe test  
leveneTest(Unemployment.Rate ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 13.085 0.0004236 \*\*\*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

boxcox(Unemployment.lm)



Unemployment2.lm <- lm(Unemployment.Rate^(0.4) ~ factor(Trump.Won)-1, data=project)  
#This still does not meet the constant variance assumption even after performing the Box-Cox Tramsformation. CAN'T CONDUCT TEST.  
leveneTest(Unemployment.Rate^0.4 ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 9.5846 0.0024 \*\*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

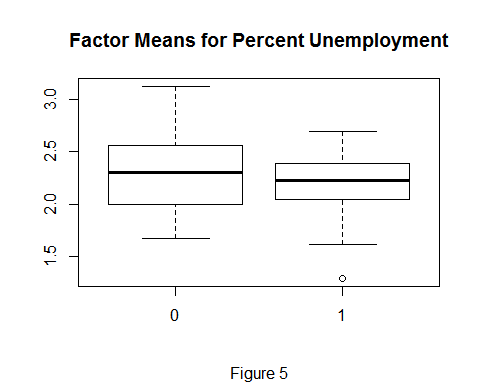
summary(Unemployment2.lm)

##   
## Call:  
## lm(formula = Unemployment.Rate^(0.4) ~ factor(Trump.Won) - 1,   
## data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.92191 -0.19452 0.00016 0.21063 0.80914   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## factor(Trump.Won)0 2.31127 0.04816 47.99 <2e-16 \*\*\*  
## factor(Trump.Won)1 2.21462 0.03158 70.12 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3046 on 131 degrees of freedom  
## Multiple R-squared: 0.9822, Adjusted R-squared: 0.9819   
## F-statistic: 3610 on 2 and 131 DF, p-value: < 2.2e-16

anova(Unemployment2.lm)

## Analysis of Variance Table  
##   
## Response: Unemployment.Rate^(0.4)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## factor(Trump.Won) 2 669.80 334.90 3610.2 < 2.2e-16 \*\*\*  
## Residuals 131 12.15 0.09   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

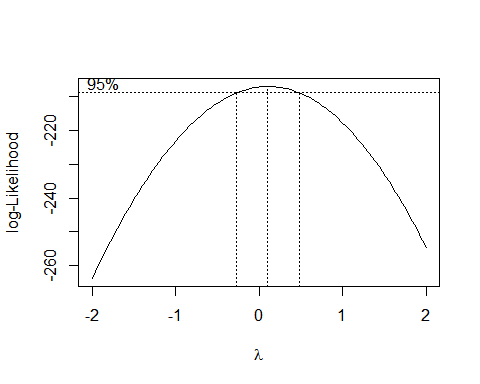
#plot(Unemployment2.lm)  
boxplot(Unemployment.Rate^(0.4) ~ factor(Trump.Won), data=project, sub="Figure 5", main="Factor Means for Percent Unemployment")



Unemployment2.aov<-aov(Unemployment.Rate^(0.4) ~ factor(Trump.Won), data=project)  
#Different  
#TukeyHSD(Unemployment2.aov)  
  
  
####### Bachelors  
  
  
Bachelors.lm <- lm(Bachelors ~ factor(Trump.Won)-1,data=project)  
#Bachelors degree does not meet the constant variance assumption from Brown Forsythe test  
leveneTest(Bachelors ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 17.038 6.479e-05 \*\*\*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

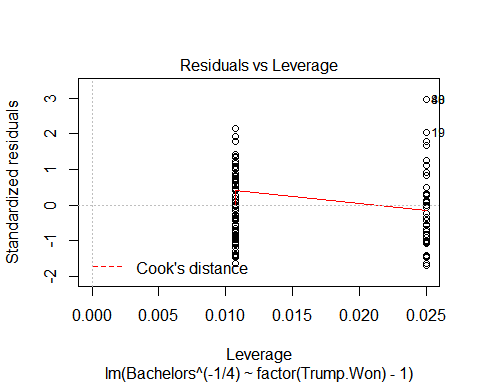
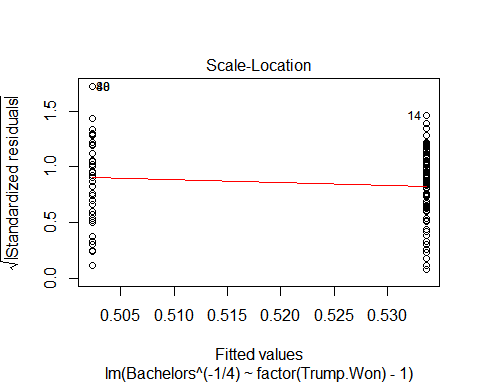
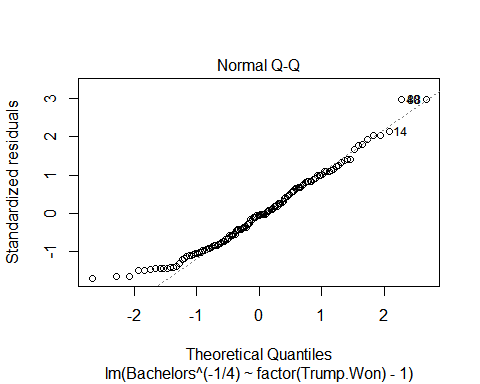
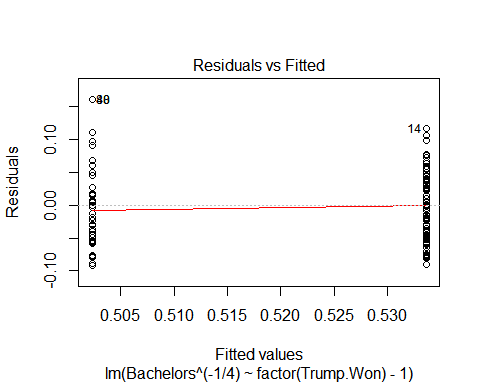
library(MASS)  
boxcox(Bachelors.lm)



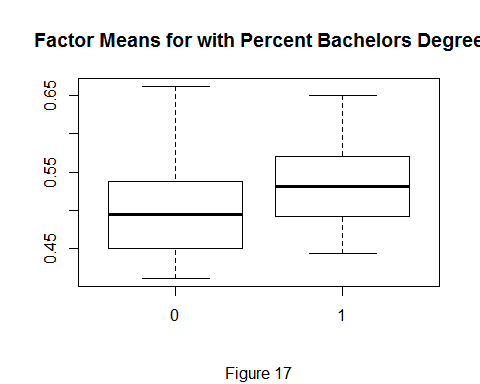
Bachelors2.lm <- lm(Bachelors^(-1/4) ~ factor(Trump.Won)-1,data=project)  
#new model meets the constant variance requirements!  
leveneTest(Bachelors^(-1/4) ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 3.6123 0.05955 .  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

plot(Bachelors2.lm)



boxplot(Bachelors^(-1/4) ~ factor(Trump.Won),data=project, sub="Figure 17", main="Factor Means for with Percent Bachelors Degree")



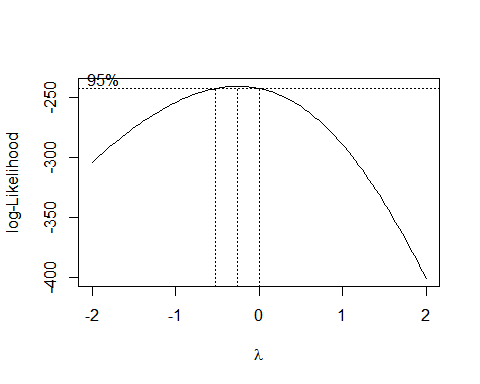
Bachelors2.aov<-aov(Bachelors^(-1/4) ~ factor(Trump.Won),data=project)  
#Different  
TukeyHSD(Bachelors2.aov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Bachelors^(-1/4) ~ factor(Trump.Won), data = project)  
##   
## $`factor(Trump.Won)`  
## diff lwr upr p adj  
## 1-0 0.03127571 0.01084417 0.05170725 0.0029636

####### Graduate  
  
  
Graduate.lm <- lm(Graduate ~ factor(Trump.Won)-1,data=project)  
#Graduate degree does not meet the constant variance assumption from Brown Forsythe test  
leveneTest(Graduate ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 1 29.326 2.827e-07 \*\*\*  
## 131   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

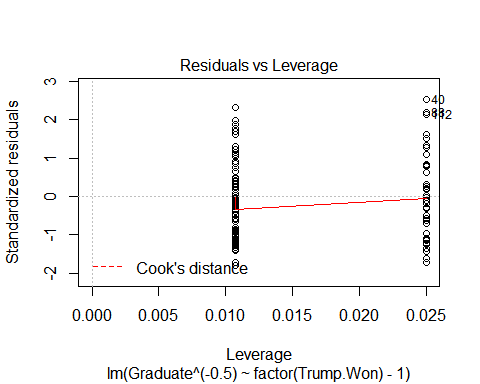
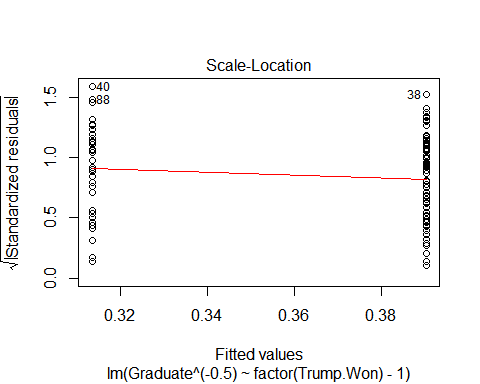
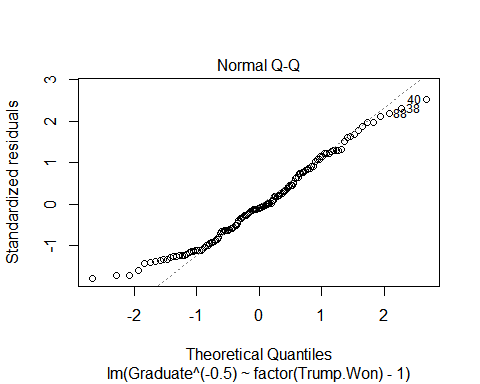
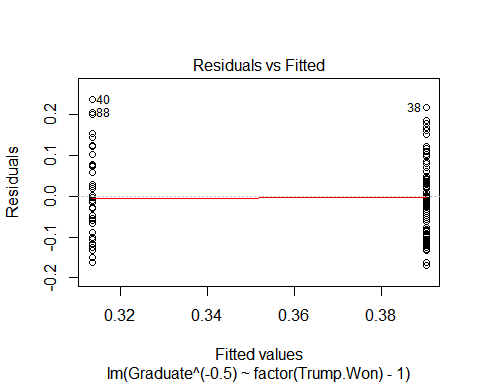
boxcox(Graduate.lm)



Graduate2.lm <- lm(Graduate^(-.5) ~ factor(Trump.Won)-1,data=project)  
#new model satisfies constant variance assumptions  
leveneTest(Graduate^(-.5) ~ factor(Trump.Won), data=project, center=median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)  
## group 1 1.6756 0.1978  
## 131

#check normality, looks fairly good  
plot(Graduate2.lm)



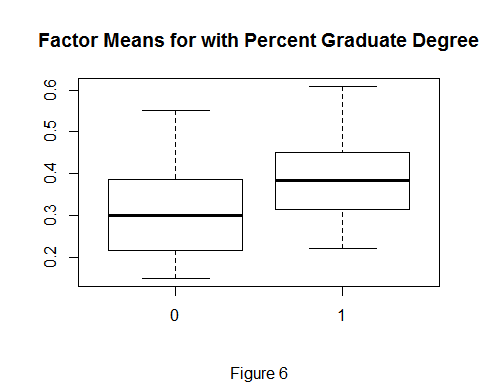
summary(Graduate2.lm)

##   
## Call:  
## lm(formula = Graduate^(-0.5) ~ factor(Trump.Won) - 1, data = project)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.168999 -0.080316 -0.009215 0.070862 0.236976   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## factor(Trump.Won)0 0.313506 0.015054 20.82 <2e-16 \*\*\*  
## factor(Trump.Won)1 0.390403 0.009873 39.54 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.09521 on 131 degrees of freedom  
## Multiple R-squared: 0.9384, Adjusted R-squared: 0.9375   
## F-statistic: 998.6 on 2 and 131 DF, p-value: < 2.2e-16

anova(Graduate2.lm)

## Analysis of Variance Table  
##   
## Response: Graduate^(-0.5)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## factor(Trump.Won) 2 18.1060 9.0530 998.64 < 2.2e-16 \*\*\*  
## Residuals 131 1.1876 0.0091   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

boxplot(Graduate^(-.5) ~ factor(Trump.Won),data=project, sub="Figure 6", main="Factor Means for with Percent Graduate Degree")



Graduate2.aov<-aov(Graduate^(-.5) ~ factor(Trump.Won),data=project)  
#Different  
TukeyHSD(Graduate2.aov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = Graduate^(-0.5) ~ factor(Trump.Won), data = project)  
##   
## $`factor(Trump.Won)`  
## diff lwr upr p adj  
## 1-0 0.07689737 0.04128308 0.1125117 3.71e-05