# HW5\_ISYE\_6501

## Question 8.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a linear regression model would be appropriate. List some (up to 5) predictors that you might use.

One of my recent task at work was to see if we could use Linear Regression to forecast/predict payroll. The drivers or predictor variables were head counts (also types of head counts), number of active work order activities, machine hours and labor hours, parts and repair orders.

**require**(ggthemes)  
library(tidyverse)  
library(magrittr)  
library(TTR)  
library(tidyr)  
library(dplyr)  
library(lubridate)  
library(ggplot2)  
library(plotly)  
library(fpp2)   
library(forecast)  
library(caTools)  
library(reshape2)  
library(psych)   
require(graphics)  
require(Matrix)  
library(corrplot)  
library(mltools)  
library(fBasics)  
library(kableExtra)  
library(DMwR)  
library(DAAG)  
library(caret)

## Question 8.2

* Linear Regression to Predict crime rates

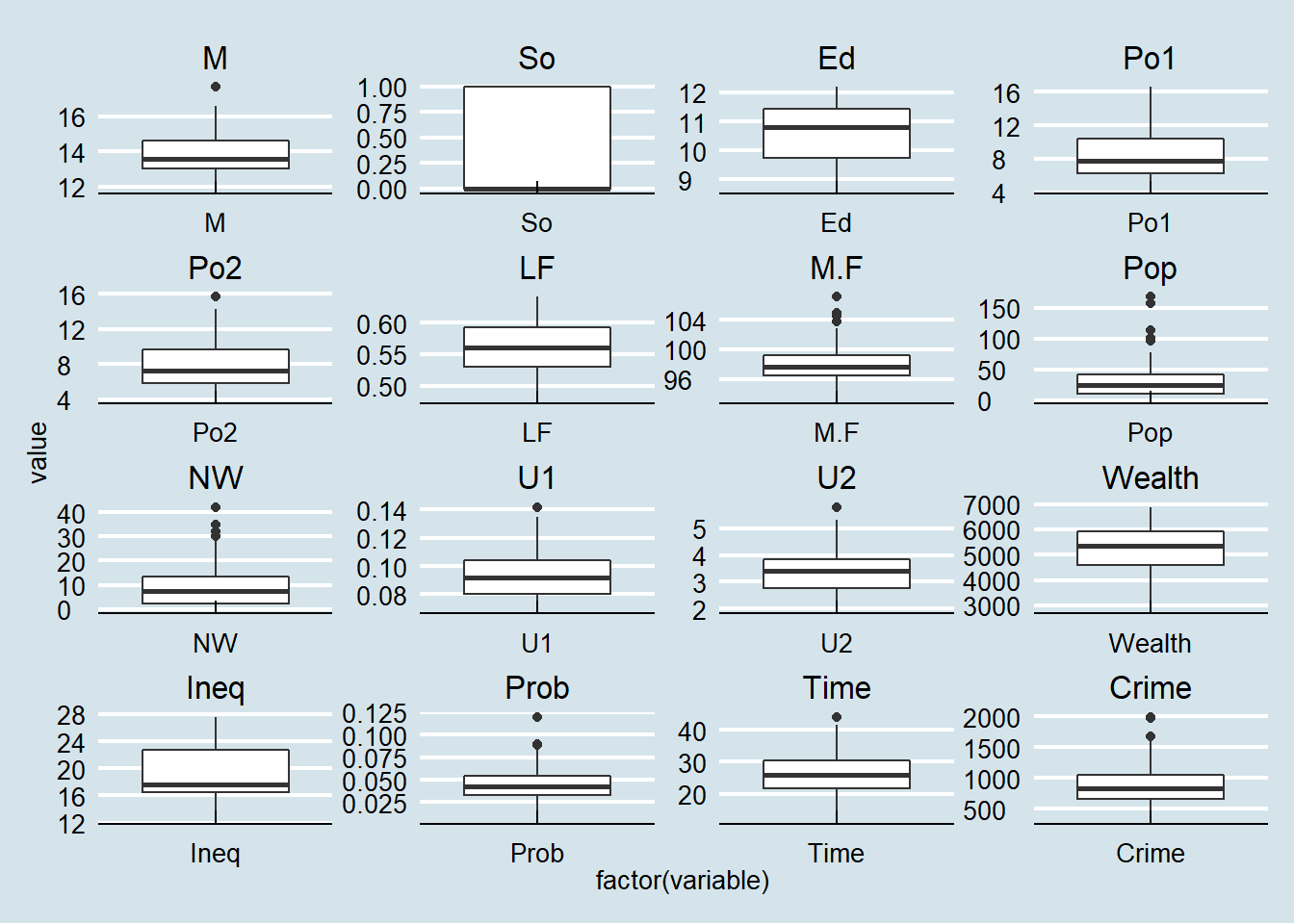
df <- read.csv(file="uscrime.csv",stringsAsFactors = F, header=T)   
head(df,2)## M So Ed Po1 Po2 LF M.F Pop NW U1 U2 Wealth Ineq Prob  
## 1 15.1 1 9.1 5.8 5.6 0.510 95.0 33 30.1 0.108 4.1 3940 26.1 0.084602  
## 2 14.3 0 11.3 10.3 9.5 0.583 101.2 13 10.2 0.096 3.6 5570 19.4 0.029599  
## Time Crime  
## 1 26.2011 791  
## 2 25.2999 1635

# Data Exploration and simple visualizations

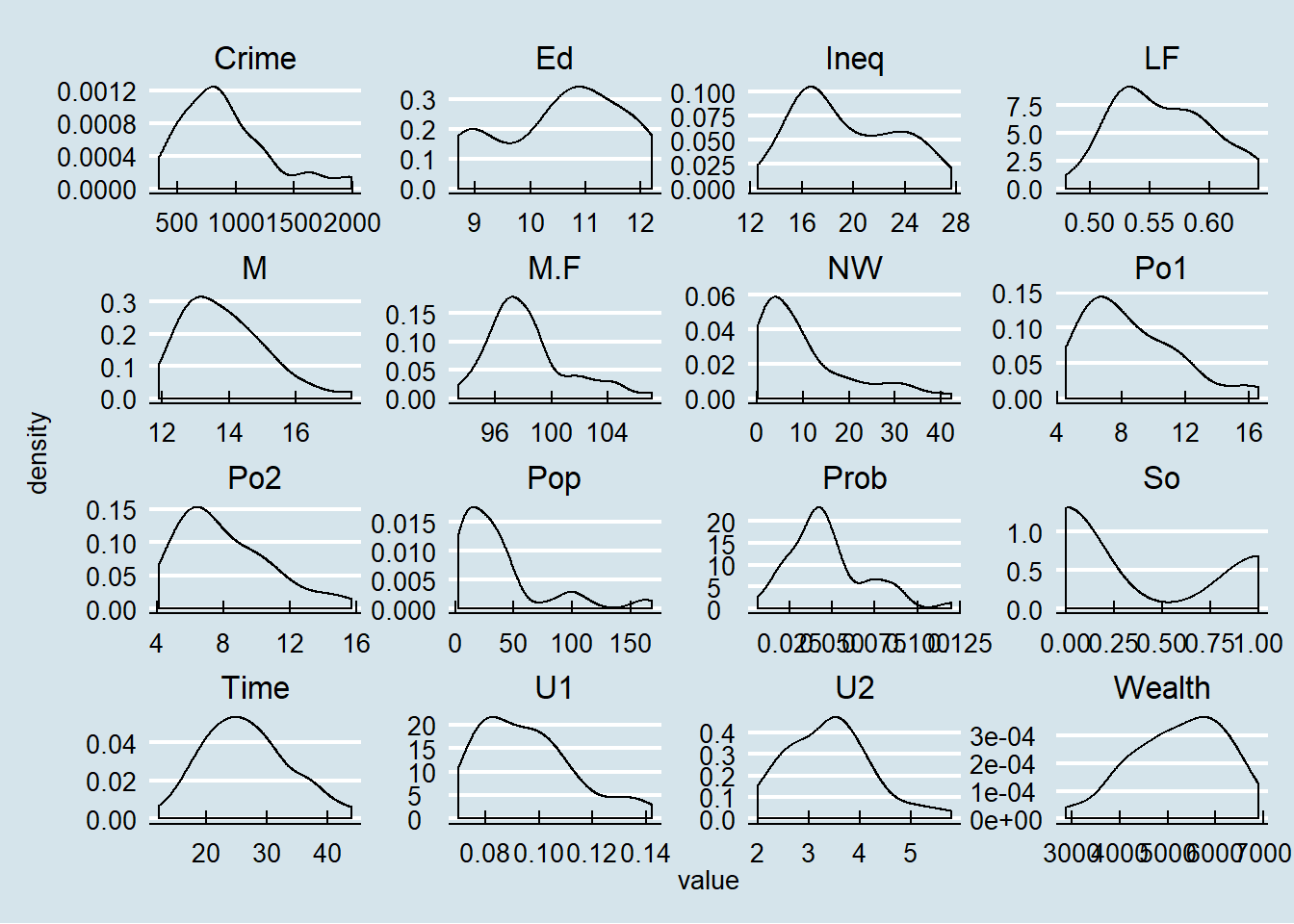
*#Checking if NA exist in Data set*  
is.null(df)## [1] FALSE*#Statistical Summary of original data*  
df\_unistats<- basicStats(df)[c("Minimum", "Maximum", "1. Quartile", "3. Quartile", "Mean", "Median",  
 "Variance", "Stdev", "Skewness", "Kurtosis"), ] %>% t() %>% as.data.frame()  
  
kable(df\_unistats)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Minimum** | **Maximum** | 1. **Quartile** | 1. **Quartile** | **Mean** | **Median** | **Variance** | **Stdev** | **Skewness** | **Kurtosis** |
| M | 11.9000 | 17.700000 | 13.000000 | 14.60000 | 13.857447 | 13.6000 | 1.579454e+00 | 1.256763 | 0.821917 | 0.377753 |
| So | 0.0000 | 1.000000 | 0.000000 | 1.00000 | 0.340426 | 0.0000 | 2.294170e-01 | 0.478975 | 0.652139 | -1.607569 |
| Ed | 8.7000 | 12.200000 | 9.750000 | 11.45000 | 10.563830 | 10.8000 | 1.251489e+00 | 1.118700 | -0.318987 | -1.149253 |
| Po1 | 4.5000 | 16.600000 | 6.250000 | 10.45000 | 8.500000 | 7.8000 | 8.832174e+00 | 2.971897 | 0.890124 | 0.162309 |
| Po2 | 4.1000 | 15.700000 | 5.850000 | 9.70000 | 8.023404 | 7.3000 | 7.818353e+00 | 2.796132 | 0.844367 | 0.008590 |
| LF | 0.4800 | 0.641000 | 0.530500 | 0.59300 | 0.561191 | 0.5600 | 1.633000e-03 | 0.040412 | 0.270675 | -0.888732 |
| M.F | 93.4000 | 107.100000 | 96.450000 | 99.20000 | 98.302128 | 97.7000 | 8.683256e+00 | 2.946737 | 0.993223 | 0.652010 |
| Pop | 3.0000 | 168.000000 | 10.000000 | 41.50000 | 36.617021 | 25.0000 | 1.449415e+03 | 38.071188 | 1.854230 | 3.078936 |
| NW | 0.2000 | 42.300000 | 2.400000 | 13.25000 | 10.112766 | 7.6000 | 1.057377e+02 | 10.282882 | 1.379966 | 1.077648 |
| U1 | 0.0700 | 0.142000 | 0.080500 | 0.10400 | 0.095468 | 0.0920 | 3.250000e-04 | 0.018029 | 0.774876 | -0.131208 |
| U2 | 2.0000 | 5.800000 | 2.750000 | 3.85000 | 3.397872 | 3.4000 | 7.132560e-01 | 0.844545 | 0.542264 | 0.173008 |
| Wealth | 2880.0000 | 6890.000000 | 4595.000000 | 5915.00000 | 5253.829787 | 5370.0000 | 9.310502e+05 | 964.909442 | -0.381952 | -0.613169 |
| Ineq | 12.6000 | 27.600000 | 16.550000 | 22.75000 | 19.400000 | 17.6000 | 1.591696e+01 | 3.989606 | 0.367063 | -1.138824 |
| Prob | 0.0069 | 0.119804 | 0.032701 | 0.05445 | 0.047091 | 0.0421 | 5.170000e-04 | 0.022737 | 0.883336 | 0.749579 |
| Time | 12.1996 | 44.000400 | 21.600350 | 30.45075 | 26.597921 | 25.8006 | 5.022408e+01 | 7.086895 | 0.371275 | -0.413474 |
| Crime | 342.0000 | 1993.000000 | 658.500000 | 1057.50000 | 905.085106 | 831.0000 | 1.495854e+05 | 386.762697 | 1.053927 | 0.777628 |

*# Visualizations via Box plots*   
meltData <- melt(df)  
  
p <- ggplot(meltData, aes(factor(variable), value))   
p + geom\_boxplot() + facet\_wrap(~variable, scale="free")+theme\_economist()+scale\_colour\_economist()



*#density plots of original data*  
density <- df %>%  
 gather() %>%   
 ggplot(aes(value)) +   
 facet\_wrap(~ key, scales = "free") +  
 geom\_density()+theme\_economist()+scale\_colour\_economist()  
density



Observation1:

* No Null predictors
* None of the predictors were normally distributed except for time. Predictors such as Ed, Ineq, Pop, So had bi-modalities issues while the rest were right-skewed (especially Wealth) and the response variable (Crime) as well

# Linear Regression Model - Base Model

* All predictors considered; no filtering of the predictors

set.seed(123)  
#Preparing data for running a better model later  
df.scaled <- scale(df[,1:15])  
df2.scaled<- data.frame(df.scaled)  
  
df3<- df2.scaled%>% mutate(Crime=df[,16])  
  
#Generate a random sample of 80% of the rows  
random\_row<- sample(1:nrow(df3),as.integer(0.8\*nrow(df3)))  
trainData = df3[random\_row,]  
#Assign the testData set to the remaining 20% of the original set  
testData = df3[-random\_row,]

* Run the model as-is; no training and no testing per HW question
* No scaling for now

*#fit the base regression model*  
base.model <- lm(Crime ~. ,data = df)  
#view the output of the base regression model  
summary(base.model)##   
## Call:  
## lm(formula = Crime ~ ., data = df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -395.74 -98.09 -6.69 112.99 512.67   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.984e+03 1.628e+03 -3.675 0.000893 \*\*\*  
## M 8.783e+01 4.171e+01 2.106 0.043443 \*   
## So -3.803e+00 1.488e+02 -0.026 0.979765   
## Ed 1.883e+02 6.209e+01 3.033 0.004861 \*\*   
## Po1 1.928e+02 1.061e+02 1.817 0.078892 .   
## Po2 -1.094e+02 1.175e+02 -0.931 0.358830   
## LF -6.638e+02 1.470e+03 -0.452 0.654654   
## M.F 1.741e+01 2.035e+01 0.855 0.398995   
## Pop -7.330e-01 1.290e+00 -0.568 0.573845   
## NW 4.204e+00 6.481e+00 0.649 0.521279   
## U1 -5.827e+03 4.210e+03 -1.384 0.176238   
## U2 1.678e+02 8.234e+01 2.038 0.050161 .   
## Wealth 9.617e-02 1.037e-01 0.928 0.360754   
## Ineq 7.067e+01 2.272e+01 3.111 0.003983 \*\*   
## Prob -4.855e+03 2.272e+03 -2.137 0.040627 \*   
## Time -3.479e+00 7.165e+00 -0.486 0.630708   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 209.1 on 31 degrees of freedom  
## Multiple R-squared: 0.8031, Adjusted R-squared: 0.7078   
## F-statistic: 8.429 on 15 and 31 DF, p-value: 3.539e-07*#(RMSE) of the model*  
print(sprintf("RMSE of Base Model = %0.2f", sigma(base.model) ))## [1] "RMSE of Base Model = 209.06"

* Using the entire data and running a Linear regression, apply the given test data points again per HW question
* Given Unscaled Test Data:

M = 14.0,So = 0,Ed = 10.0,Po1 = 12.0,Po2 = 15.5,LF = 0.640,M.F = 94.0,Pop = 150,NW = 1.1,U1 = 0.120,U2 = 3.6

Wealth = 3200,Ineq = 20.1,Prob = 0.04 & Time = 39.0

testpts <-data.frame(M = 14.0,So = 0, Ed = 10.0, Po1 = 12.0, Po2 = 15.5,LF = 0.640, M.F = 94.0, Pop = 150, NW = 1.1, U1 = 0.120, U2 = 3.6, Wealth = 3200, Ineq = 20.1, Prob = 0.040,Time = 39.0)  
  
#Predict the crime rate for test data point  
pred.basemodel <- predict(base.model, testpts)  
pred.basemodel## 1   
## 155.4349

Observation1:

* The prediction is really low given the test data predictors were well within all the other predictors. So one would expect that the predicted response value to be somewhere in between the lowest and the highest crime numbers. Its lower than the minimum of the original Crime data which is 342
* Probable cause: model was not specified properly in that all the predictors were utilized regardless of its significance to the model. Although the p-value, F-stat and R-square of the entire model looked deceptively good, the model was using predictors that were not statistically significant. Could result in over fitting issue

* Let’s be more judicious (parsimonious) in using only the significant predictors
* Using summary output as a guide, took predictors with ., \* & \*\* only and fed that to the lm function again

better.model <- lm( Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = df)  
summary(better.model)##   
## Call:  
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -470.68 -78.41 -19.68 133.12 556.23   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5040.50 899.84 -5.602 1.72e-06 \*\*\*  
## M 105.02 33.30 3.154 0.00305 \*\*   
## Ed 196.47 44.75 4.390 8.07e-05 \*\*\*  
## Po1 115.02 13.75 8.363 2.56e-10 \*\*\*  
## U2 89.37 40.91 2.185 0.03483 \*   
## Ineq 67.65 13.94 4.855 1.88e-05 \*\*\*  
## Prob -3801.84 1528.10 -2.488 0.01711 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 200.7 on 40 degrees of freedom  
## Multiple R-squared: 0.7659, Adjusted R-squared: 0.7307   
## F-statistic: 21.81 on 6 and 40 DF, p-value: 3.418e-11*#(RMSE) of the model*  
print(sprintf("RMSE of Better Model = %0.2f", sigma(better.model) ))## [1] "RMSE of Better Model = 200.69"pred.better.model <- predict(better.model, testpts)  
pred.better.model## 1   
## 1304.245

Observation2:

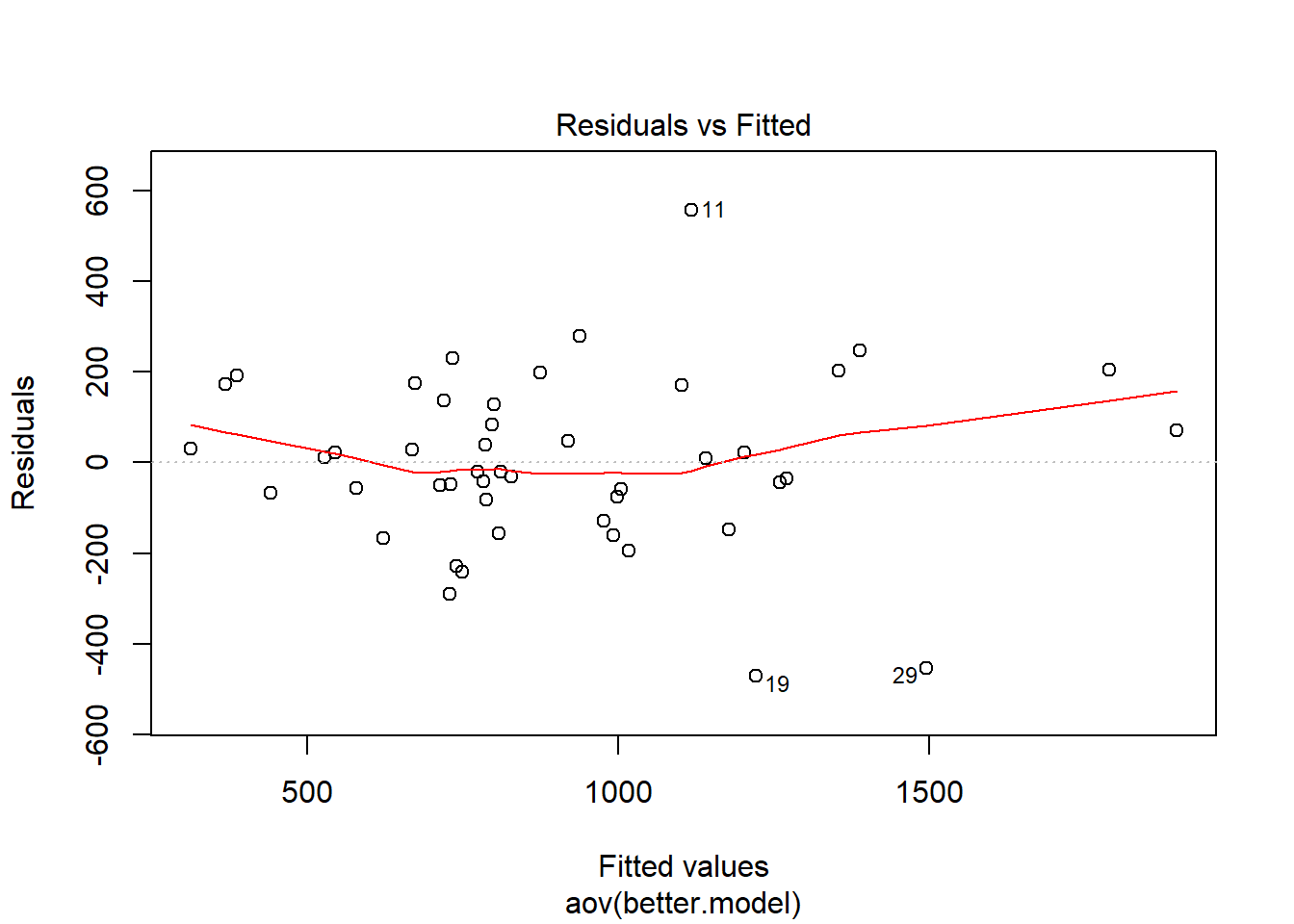
* This looks more reasonable as it’s somewhere between the highest and the lowest. It’s better than the base model for sure. But how much better? The RSME improved a little from 209 to 200
* The p-value, F-stat and the R-square still looked good now that all the predictors are statistically significant! So we can be more confident about this than the previous one
* The main question is quality; is this a “better” model? One of the ways to determine that is to perform a statistical (residual) analysis via ANOVA as follows:

# Statistical Diagnostics

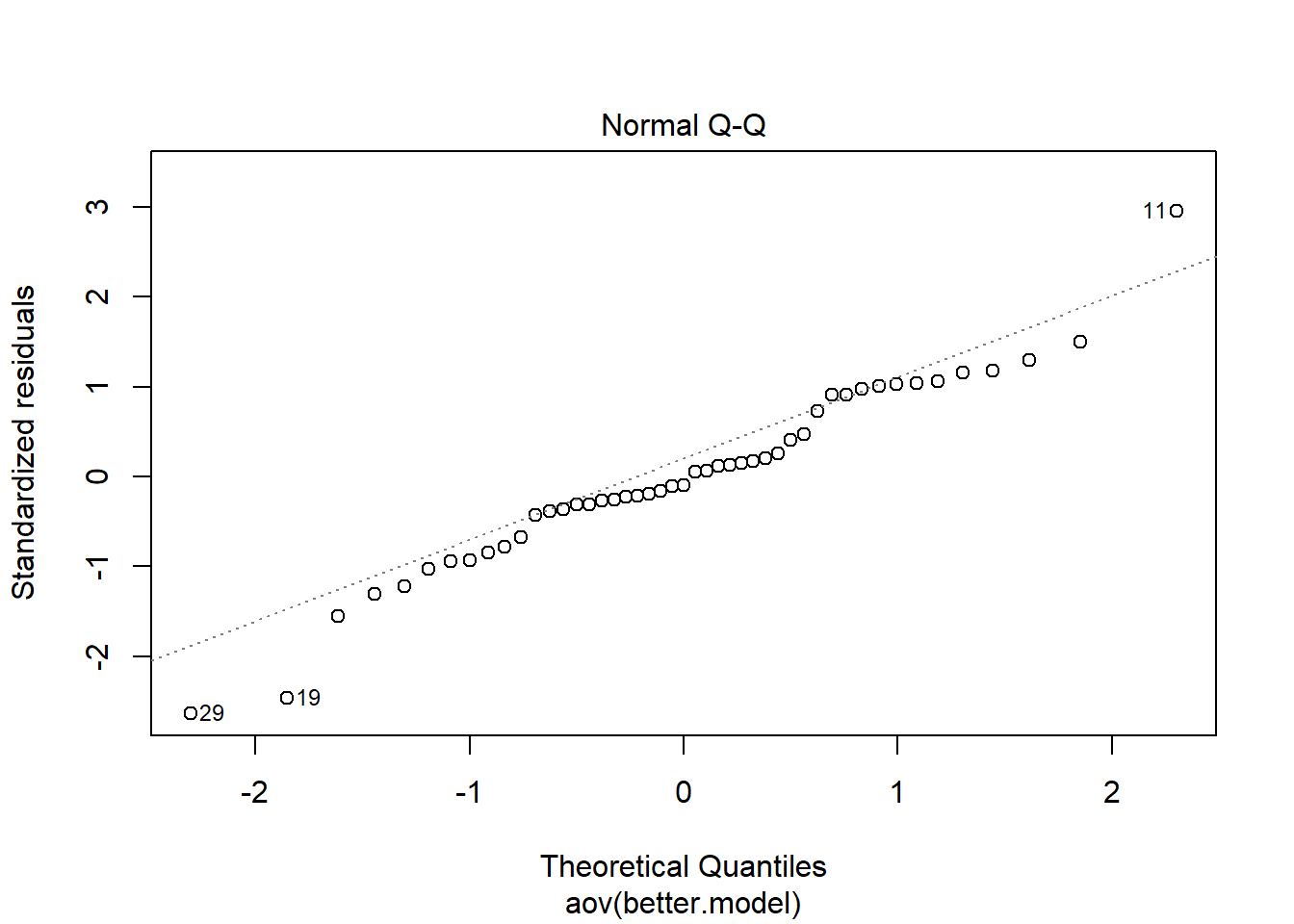
* Statistical Analysis on the “better” model

1. Homogeneity of variances
2. Normality of residual via QQ plot
3. histogram of residuals via Histogram

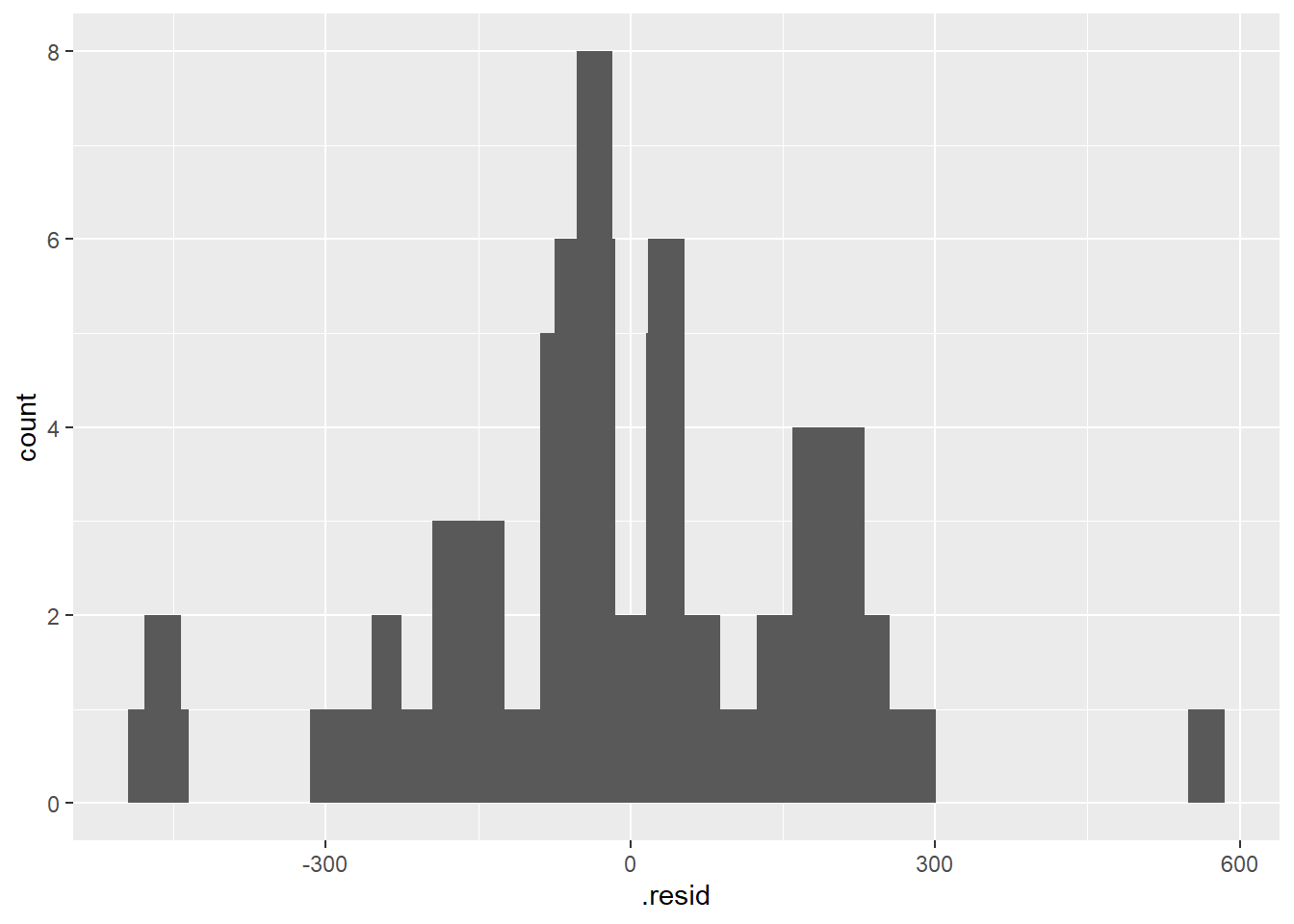
*# Compute the analysis of variance*  
res.aov <- aov(better.model, data = df)  
# Summary of the analysis  
summary(res.aov)## Df Sum Sq Mean Sq F value Pr(>F)   
## M 1 55084 55084 1.368 0.249138   
## Ed 1 725967 725967 18.025 0.000126 \*\*\*  
## Po1 1 3173852 3173852 78.802 5.33e-11 \*\*\*  
## U2 1 217386 217386 5.397 0.025336 \*   
## Ineq 1 848273 848273 21.061 4.34e-05 \*\*\*  
## Prob 1 249308 249308 6.190 0.017114 \*   
## Residuals 40 1611057 40276   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1*# 1. Homogeneity of variances check*  
plot(res.aov, 1)



*# 2. Normality of residual check*  
plot(res.aov, 2)



*#histogram of residuals*  
ggplot(better.model, aes(x=.resid))+geom\_histogram(binwidth = 30)+ geom\_histogram(bins=30)



Observation3:

ANOVA showed the following:

1. The residuals do Not exhibit heterodasticity; this means we have constant variance of residuals and that is good
2. QQ plot showed residuals appeared to be normal at the core but not at the outer fringes of the data
3. Histogram of residuals confirmed what QQ plot had revealed; a few outliers (May or may not need corrective actions)

* M failed the F-test and could be in danger of being tossed out
* The other thing we have not touched on is whether there are any multicollinearity between the predictors. For that we can use correlation matrix between predictors first and confirm it with Variance Inflation Factors (VIF) analysis

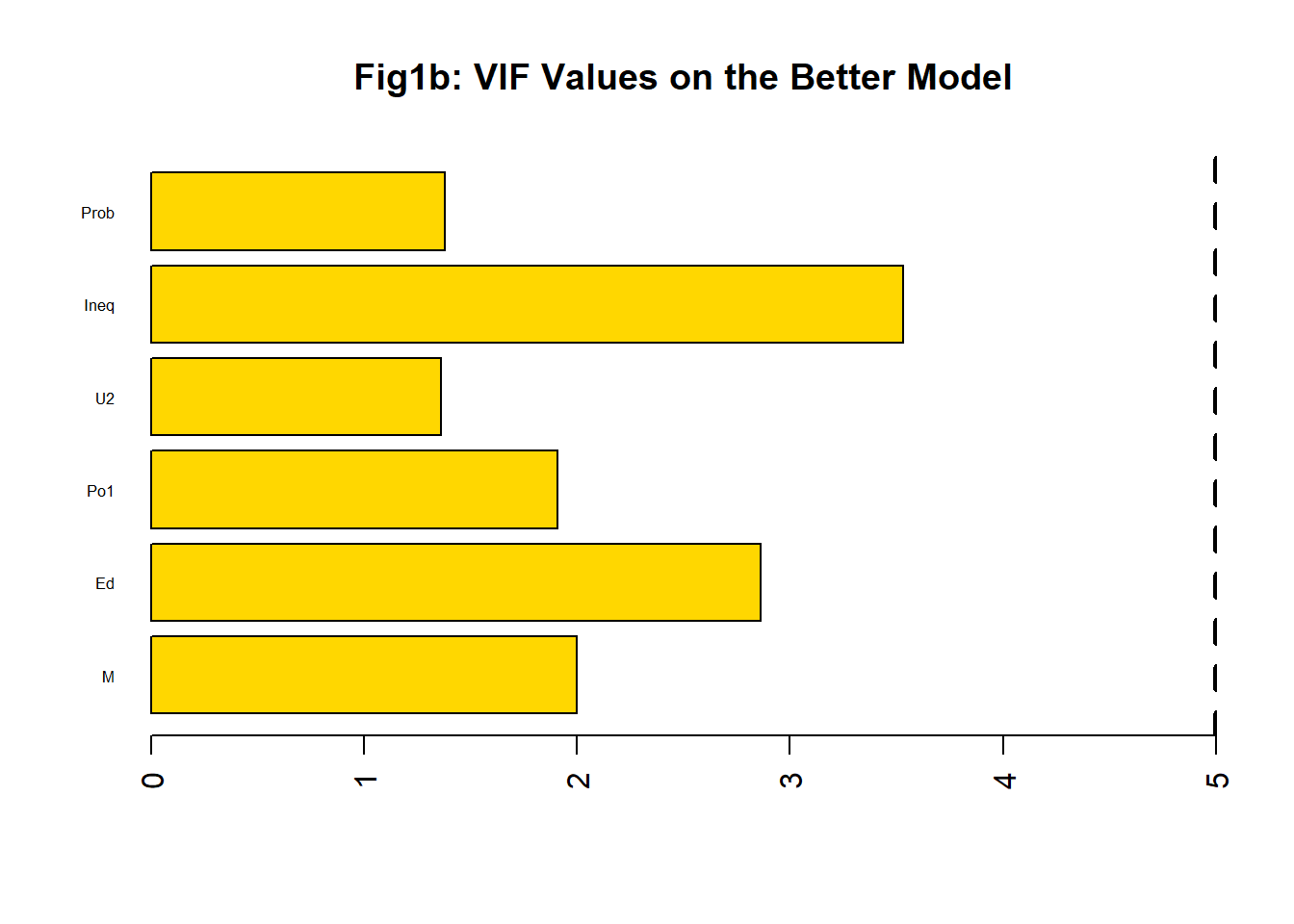
headers = c("M","Ed","Po1","U2","Ineq","Prob")  
  
newdata <- df2.scaled [headers]  
correlation = cor(newdata )  
corrplot(correlation, method = 'number',type='upper',bg="lightblue")  
mtext("Fig1a: Correlation Matrix", at=.95, line=-15, cex=1.2)



*#calculate the VIF for each predictor variable in the model*  
 VIF <- **function**(linear.model, no.intercept=FALSE, all.diagnostics=FALSE, plot=FALSE) {  
 **require**(mctest)  
 **if**(no.intercept==FALSE) design.matrix <- model.matrix(linear.model)[,-1]  
 **if**(no.intercept==TRUE) design.matrix <- model.matrix(linear.model)  
 **if**(plot==TRUE) mc.plot(design.matrix,linear.model$model[1])  
 **if**(all.diagnostics==FALSE) output <- imcdiag(design.matrix,linear.model$model[1], method='VIF')$idiags[,1]  
 **if**(all.diagnostics==TRUE) output <- imcdiag(design.matrix,linear.model$model[1])  
 output  
 }  
  
VIF(better.model)## M Ed Po1 U2 Ineq Prob   
## 2.000244 2.862893 1.908124 1.363074 3.530429 1.378714*#create vector of VIF values*  
vif\_values <- VIF(better.model)

* VIF plots

*#create horizontal bar chart to display each VIF value*  
barplot(vif\_values, main = "Fig1b: VIF Values on the Better Model", horiz = TRUE, col = "gold",las=2,cex.names=.53,xlim=c(0,5))  
#add vertical line at 5  
abline(v = 5, lwd = 3, lty = 2)



observation4:

* In general correlation matrix showed inter-correlation relationships were low. However some were greater than 0.5. But the VIF plot indicated no multicollinearity issues in using these set of predictors. Note: Anything lower than 5 is good
* Let’s move forward with these predictors but dropping M and go with only Pro, Ineq, U2, P01 and Ed. This time let’s do a simple 80/20 split on training data and check for model quality. And repeating that for test data

# Linear Regression Model on training set

set.seed(123)  
#fit the regression model on train data  
train.model <-lm( Crime ~ Ed + Po1 + U2 + Ineq + Prob, data = trainData)  
#view the output of the regression model  
summary(train.model)##   
## Call:  
## lm(formula = Crime ~ Ed + Po1 + U2 + Ineq + Prob, data = trainData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -557.11 -112.13 23.15 170.20 465.00   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 897.02 40.34 22.236 < 2e-16 \*\*\*  
## Ed 164.92 67.58 2.440 0.020584 \*   
## Po1 325.69 51.38 6.338 4.7e-07 \*\*\*  
## U2 28.67 49.99 0.574 0.570421   
## Ineq 282.25 72.09 3.915 0.000462 \*\*\*  
## Prob -68.59 43.96 -1.560 0.128846   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 242.1 on 31 degrees of freedom  
## Multiple R-squared: 0.6951, Adjusted R-squared: 0.6459   
## F-statistic: 14.13 on 5 and 31 DF, p-value: 3.146e-07*#(RMSE) of the model*  
print(sprintf("RMSE of Train Model = %0.2f", sigma(train.model) ))## [1] "RMSE of Train Model = 242.13"

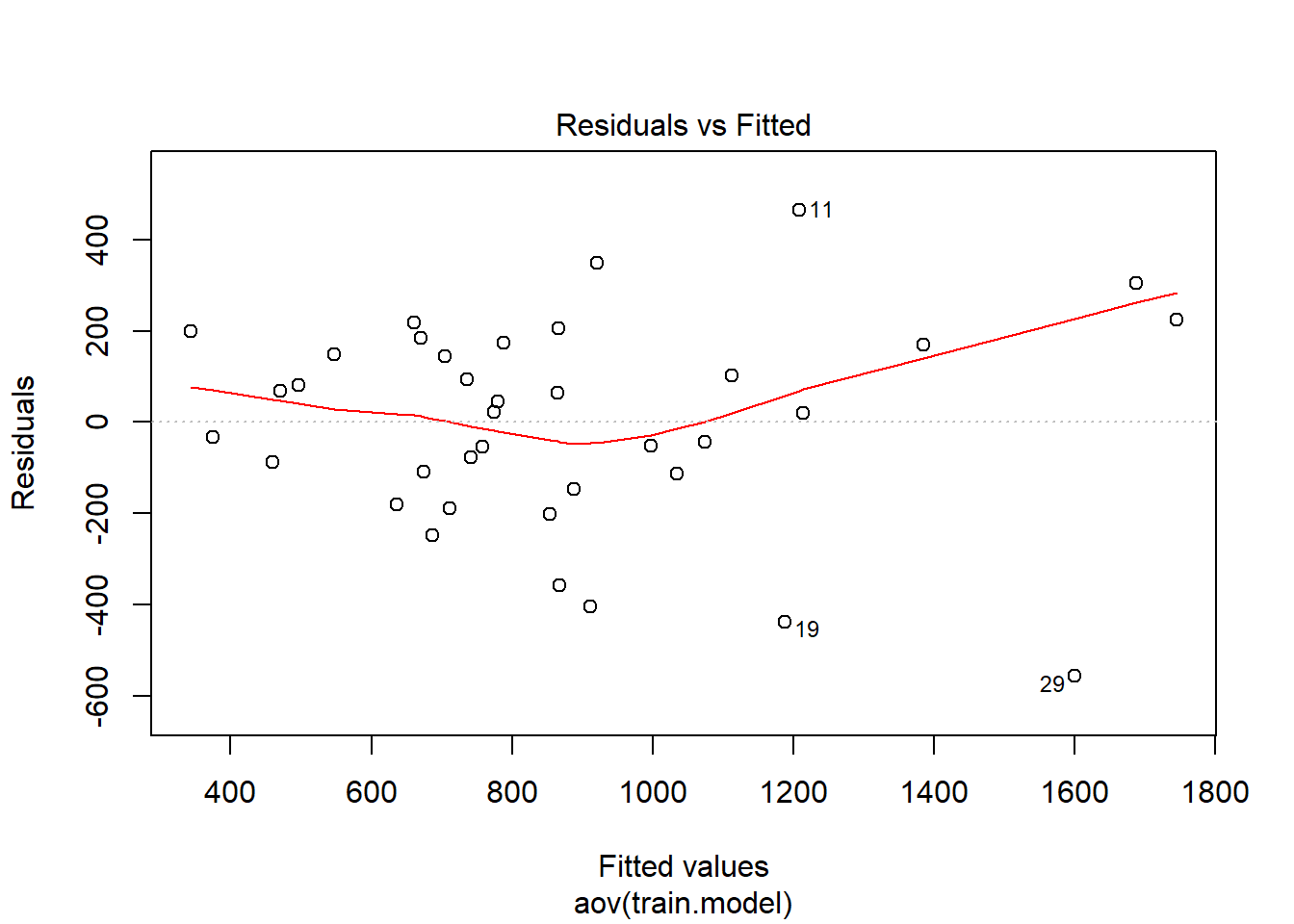
observation5:

* Let’s perform a statistical diagnostic check on the quality of the train model
* The RSME of the train model saw a jump to 242 from 200 while R-square, F-Stat and p-values still looked good

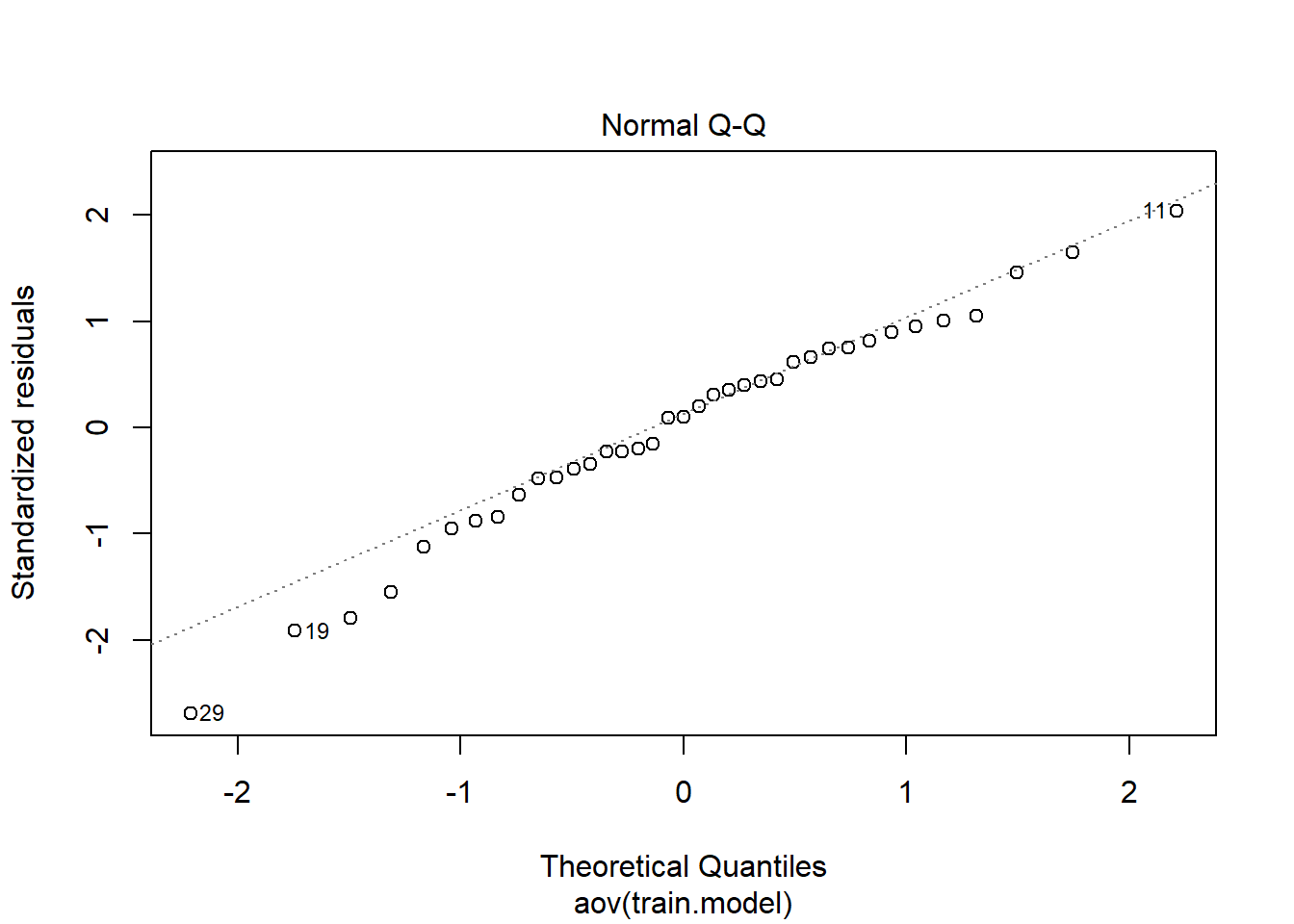
# Statistical Diagnostics

* On Train Data only

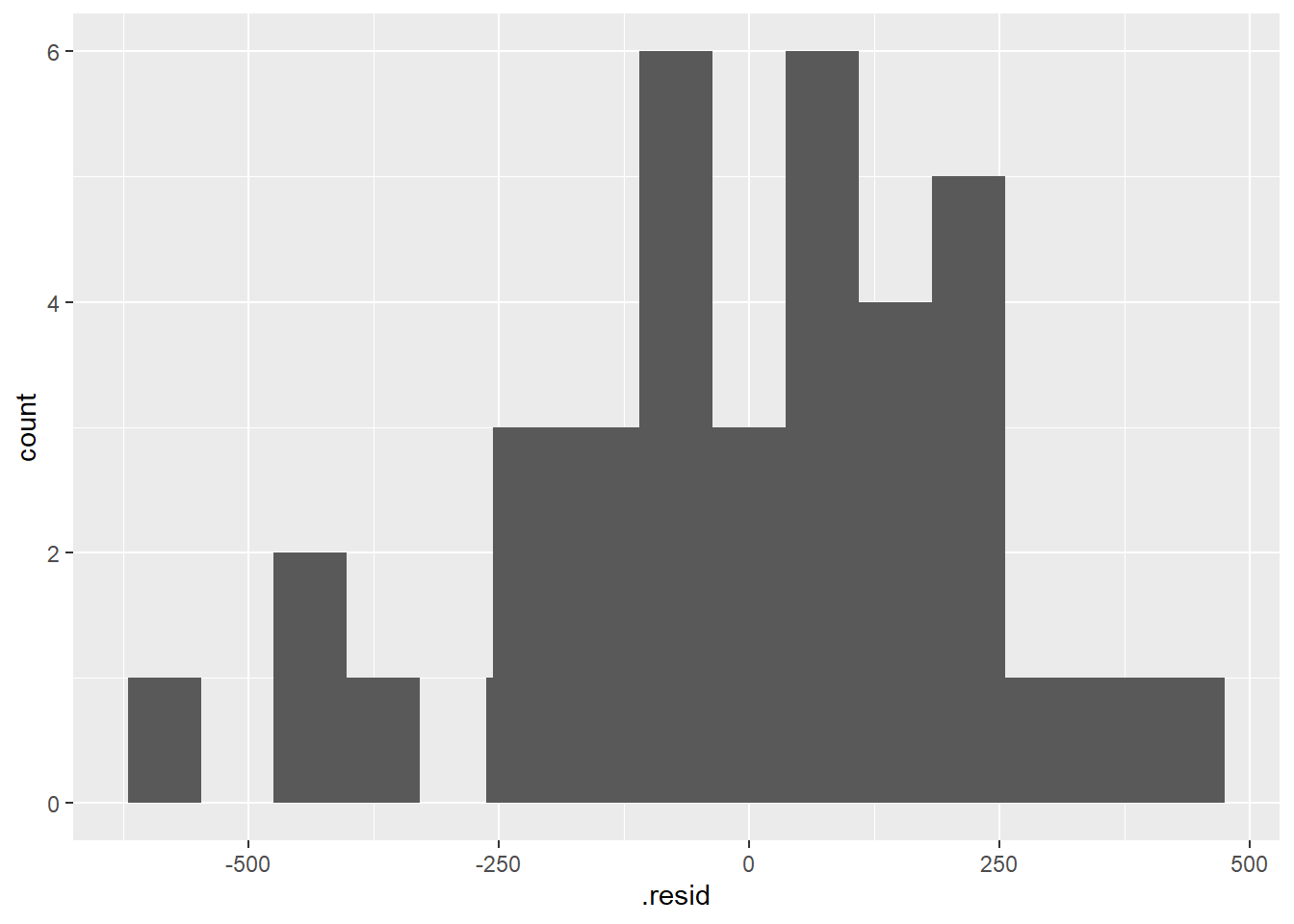
*# Compute the analysis of variance*  
res.aov.train <- aov(train.model, data = trainData)  
# Summary of the analysis  
summary(res.aov.train)## Df Sum Sq Mean Sq F value Pr(>F)   
## Ed 1 741889 741889 12.655 0.001228 \*\*   
## Po1 1 2442479 2442479 41.663 3.39e-07 \*\*\*  
## U2 1 5150 5150 0.088 0.768907   
## Ineq 1 810201 810201 13.820 0.000796 \*\*\*  
## Prob 1 142720 142720 2.434 0.128846   
## Residuals 31 1817365 58625   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1*# 1. Homogeneity of variances check*  
plot(res.aov.train, 1)



*# 2. Normality of residual check*  
plot(res.aov.train, 2)



*#histogram of residuals*  
ggplot(train.model, aes(x=.resid))+geom\_histogram(binwidth = 15)+ geom\_histogram(bins=15)



# Linear Regression Model on test set

* Since training model indicated U2 and Prob are insignificant, let’s tossed those as well on the test set
* We are down to only 3 predictors; Ed + Po1 + Ineq;

set.seed(123)  
#fit the regression model on test data  
test.model <-lm( Crime ~ Ed + Po1 + Ineq , data = testData)  
#view the output of the regression model  
summary(test.model)##   
## Call:  
## lm(formula = Crime ~ Ed + Po1 + Ineq, data = testData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -155.608 -89.161 8.519 93.773 158.841   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 897.96 46.88 19.155 1.31e-06 \*\*\*  
## Ed 363.09 91.11 3.985 0.00724 \*\*   
## Po1 670.37 125.05 5.361 0.00173 \*\*   
## Ineq 592.25 101.69 5.824 0.00113 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 135.1 on 6 degrees of freedom  
## Multiple R-squared: 0.8602, Adjusted R-squared: 0.7903   
## F-statistic: 12.31 on 3 and 6 DF, p-value: 0.005655print(sprintf("RMSE of Test Model = %0.2f", sigma(test.model) ))## [1] "RMSE of Test Model = 135.07"

observation6:

* ANOVA results on training data agreed with testing data in that all the predictors “recommended” from the training set were also statistically significant in the testing set
* Now we see further reduction in RSME to 135; another good sign
* From ANOVA, homogeneity of variances checked out while normality of the residuals were still bit deviated but only at the lower end. All indication of a parsimonious model; Crime ~ Ed + Po1 + Ineq

Final Linear Equation:   
  
Crime = 897.96+363.09Ed+670.37P01+592.25Ineq  
  
Note: Ed, P01 & INeq are scaled independent variables

# Predict the observed crime rate

* Using the same HW given observed data again on the test model
* Ed = 10.0 => scaled: -0.504 (avg=10.563, Std Dev.=1.118)
* P01 = 12.0 => scaled: 1.177 (avg=8.5, Std Dev.=2.971)
* Ineq = 20.1 => scaled: 0.175 (avg=19.4, Std Dev.=3.989)

testpts.filtered <-data.frame(Ed = -0.504, Po1 = 1.177, Ineq = 0.175)*#these test points had to be scaled*  
*#Predict the crime rate for test data point*  
pred.prodmodel <- predict(test.model , testpts.filtered)  
cat("The predicted Crime using the test model = ",pred.prodmodel,sep="",fill=TRUE)## The predicted Crime using the test model = 1607.626

# Summary

- As hinted in the HW question, there were excess parameters or factors for such a small sample size, leading to overfitting. We tossed out alot of insignifcant predictors, ran both a correlation and a VIF analyses to ensure we do not have multicollinearity issues for predictors that were deemed signifcant   
  
- To produce a less-overfitting and more robust linear model, we used the following items as our quality guide:  
  
(1) Overall RSME, F-stat, p-value and R-Sqaured   
  
(2) Further confirmation with ANOVA analysis to ensure we maintained Homogeneity of variances and Normality of residuals  
  
- Statistical diagnostics showed that we only needed a handful of predictors to form a robust, yet parsimonious linear predictive model: Crime ~ function(Ed, Po1, Ineq)  
  
- Final linear regression equation: Crime = 897.96+363.09Ed+670.37P01+592.25Ineq