DEMO 5 - LINEAR REGRESSION

QUESTION 1 - REAL-LIFE USE-CASE WARRANTING THE USE OF LINEAR REGRESSION

the best example that I've applied a linear regression model to was a rainfall prediction system I built around five years ago. As you know climate can be a very difficult aspect to model. The model would take the past data filtered seasonally and use the following predictors to determine the estimated rainfall at a point in the future, updating the prediction in real-time -

1. Barometer readings for humidity (numerical predictor)
2. Anemometer readings for wind-speed (numerical predictor)
3. Build-Up fraction - this is a derived measure of how time has passed since the last rainfall over the avaerage time that usually exists between two rainfalls WITHIN the same season and NEAR the same barometer and anemometer readings (numerical predictor)
4. Temperature / Evaporation Rate (numerical predictor)
5. Total volume of water present within location (numerical predictor)
6. Total surface area of water present within location (numerical predictor)
7. Tree cover area (numerical predictor)

The model was enhanced for commercial use using the following two methods -

1. The locations to which this model was applied were decentralized and aggragated part by part for an overview report on weather for a large area.
2. All outliers were examined in terms of causal-inference relationships after the linear regression model was built, and the indicators were narrowed down for significance.

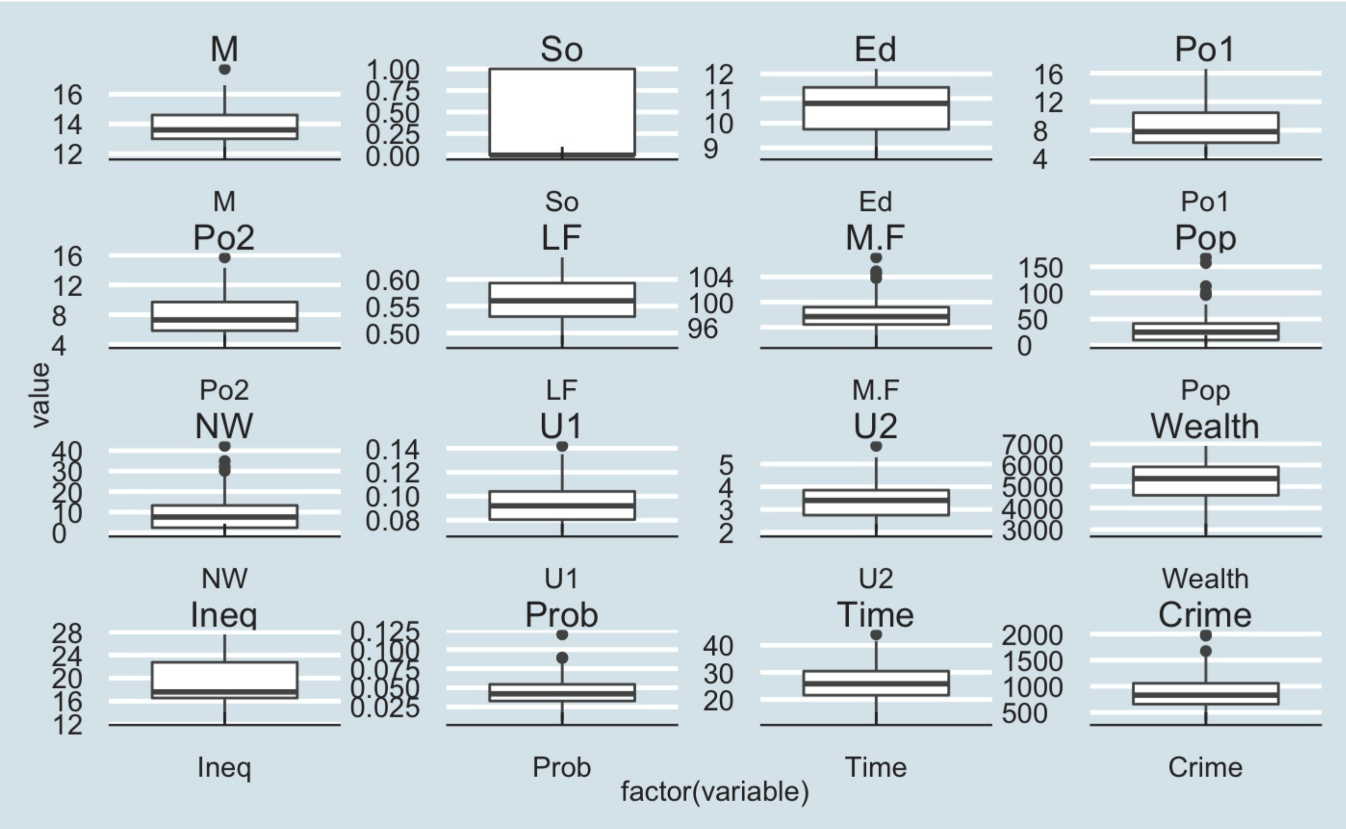
QUESTION 2 - PREDICTED CRIME RATE USING LINEAR REGRESSION

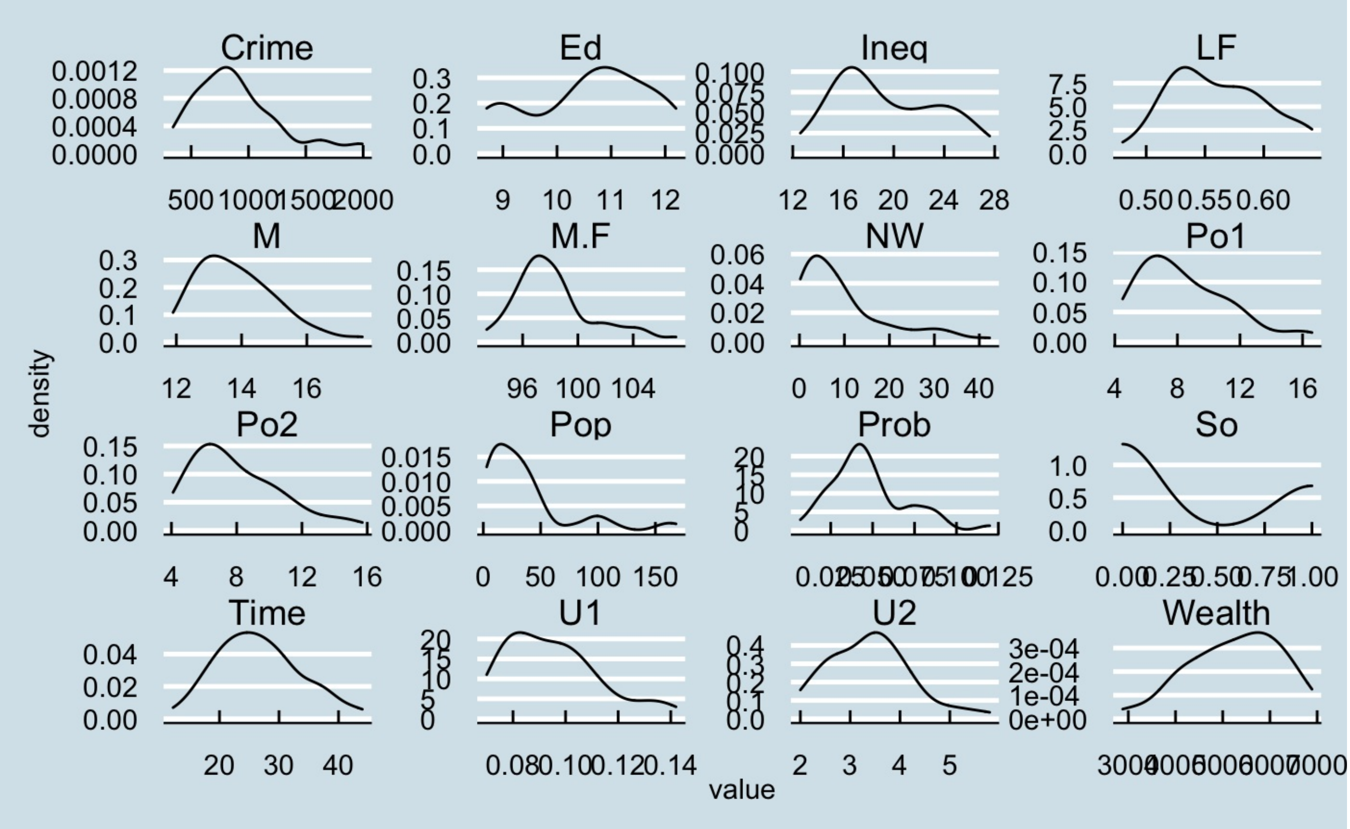
The answers is elaborated in the following manner -

1. The Inferences and Analyses WITH Graphs to Support
2. The Code
3. The Software Output

INFERENCES AND ANALYSES

1 - From the boxplots and density plots shown we can conclude that outliers are not a major issue for any of the predictors, and could be addressed (maybe!) for NW, MF, and Prob. And among distributions, Time is the only one normally distributed.





2 - Then we run the linear regression model as such without test-train partition and check the p-values and RMSE values of the predictors. The problem here is that even though these values appear to be good, the model is performing poorly with the predict result not even falling in the range of highest and lowest crime numbers. So what we can try is reducing the boundaries of significance to less that 0.01. We then get these results -

The old RMSE value -

209.06

The reduced set of predictors -

M, Ed, Po1, U2, Ineq, Prob

The new RMSE value -

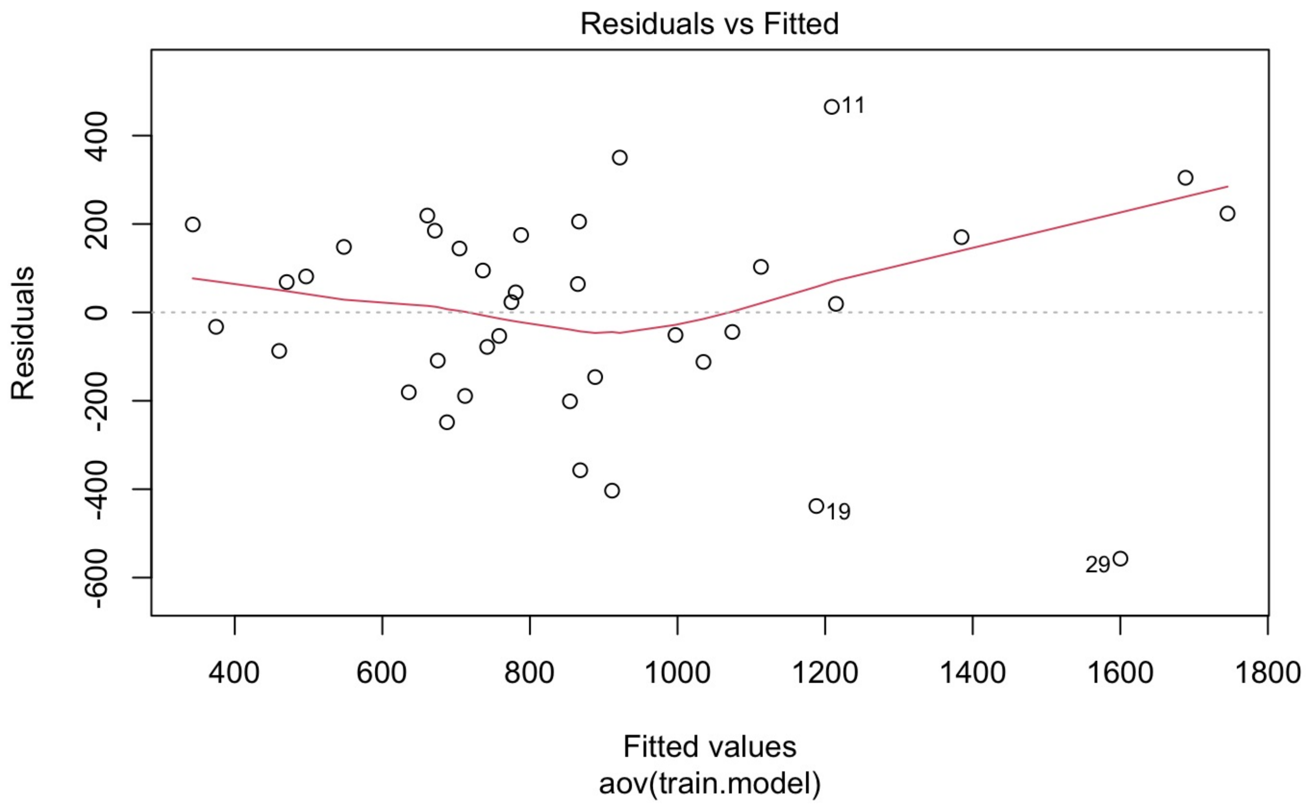
200.69

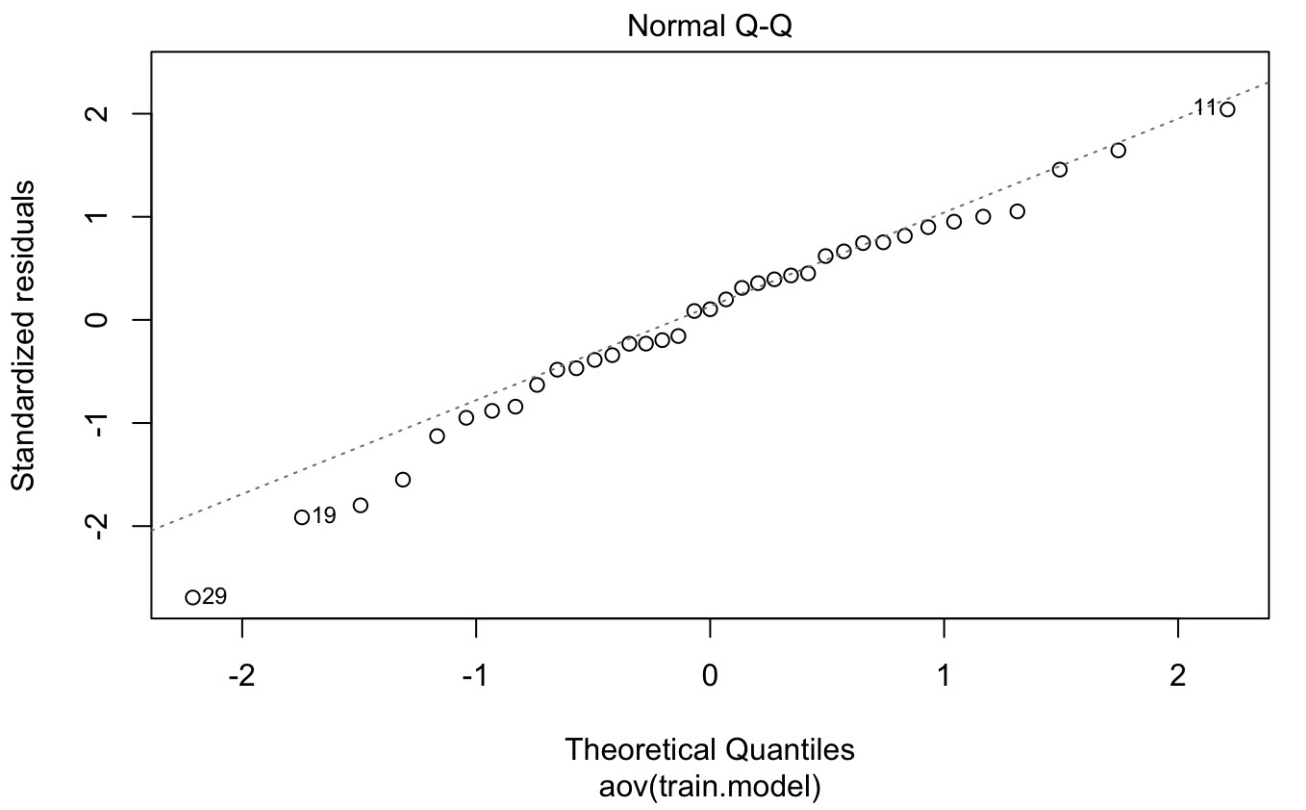
And then we use this to find the output value of the data instance given -

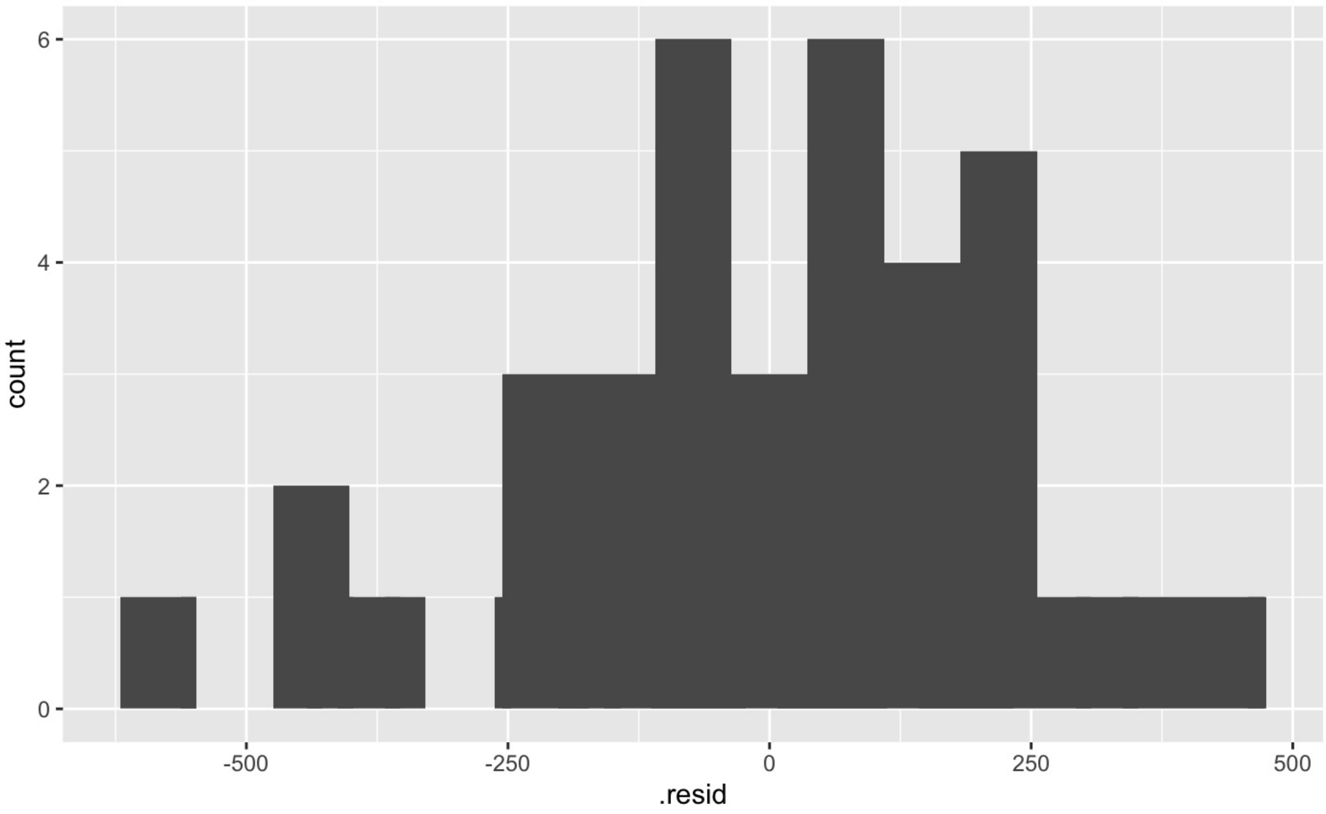
1304.245

The quality report -

From the residuals, QQplot, and histogram, we derive that the graph is homoskedastic and the variance is constant, the residuals are normal except at the extremes, and the outliers are not very alarming. Note that predictor M failed the F-test and will be weeded out for the final model. The graphs are as follows -


Residuals vs Fitted
Residuals
400
200
0
-200
-400
-600
400
600
800
1000
1200
1400
1600
1800
Fitted values
aov(train.model)
11
19
29


Normal Q-Q
Standardized residuals
2
1
0
-1
-2
-2
-1
0
1
2
Theoretical Quantiles
aov(train.model)
11
19
29



3 - Now we run the same after we drop M, and divide data into test-train 80-20 ratio to get better results. And we get the following results after scaling while still maintaining good p-values and r-squared values-

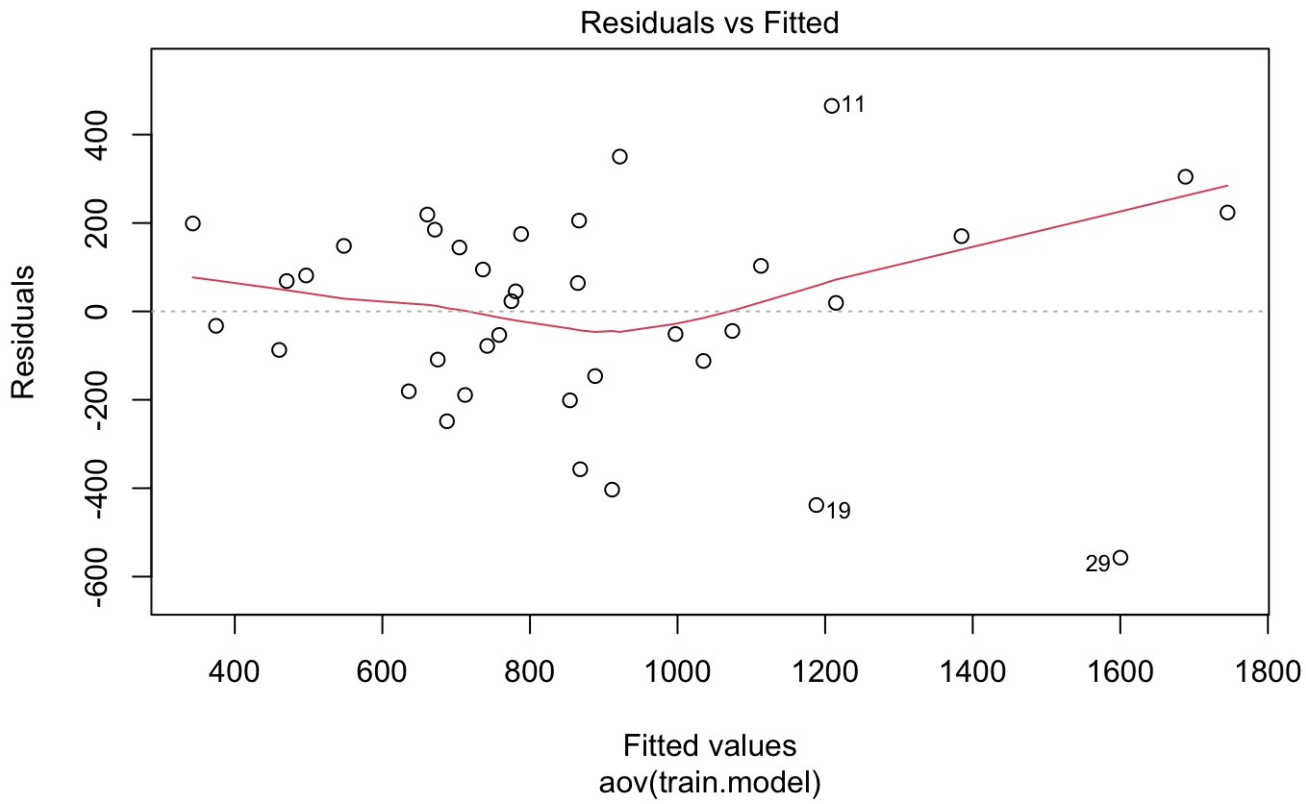
The new RMSE value on train set -

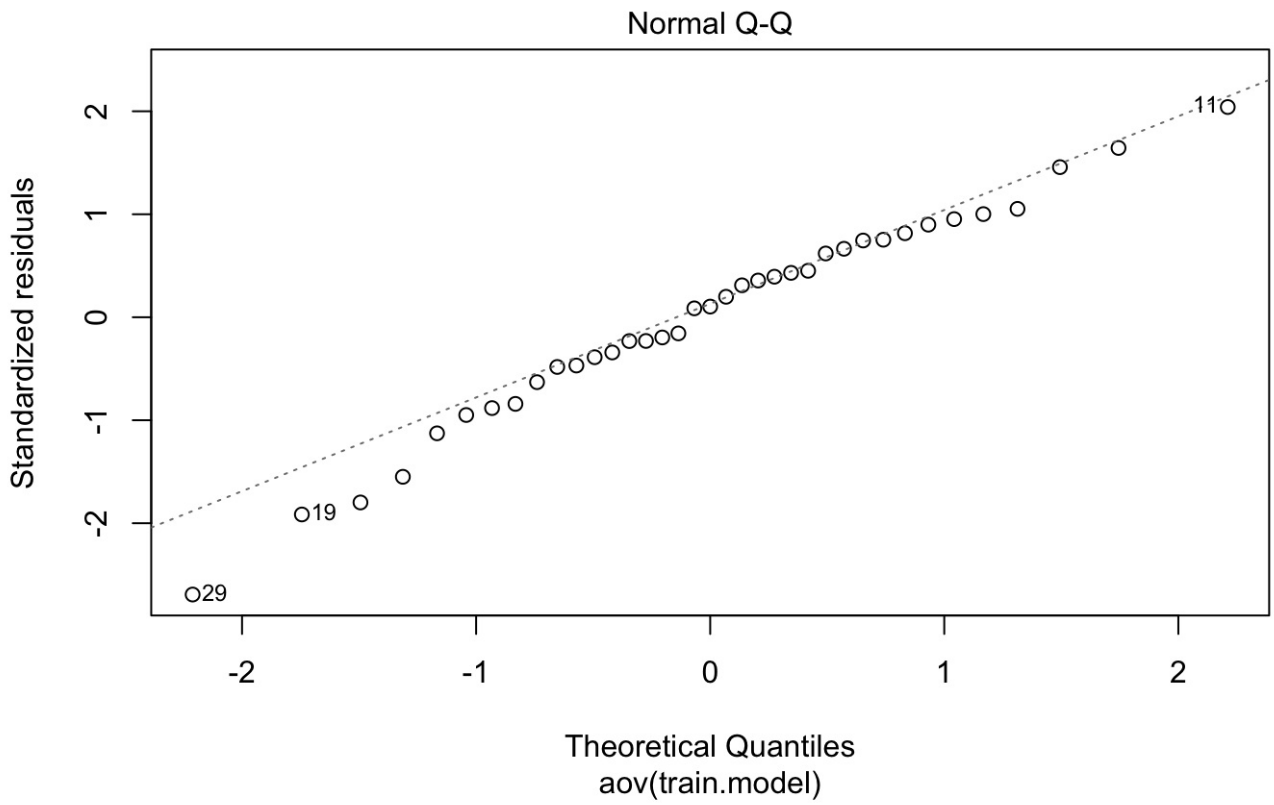
242.13

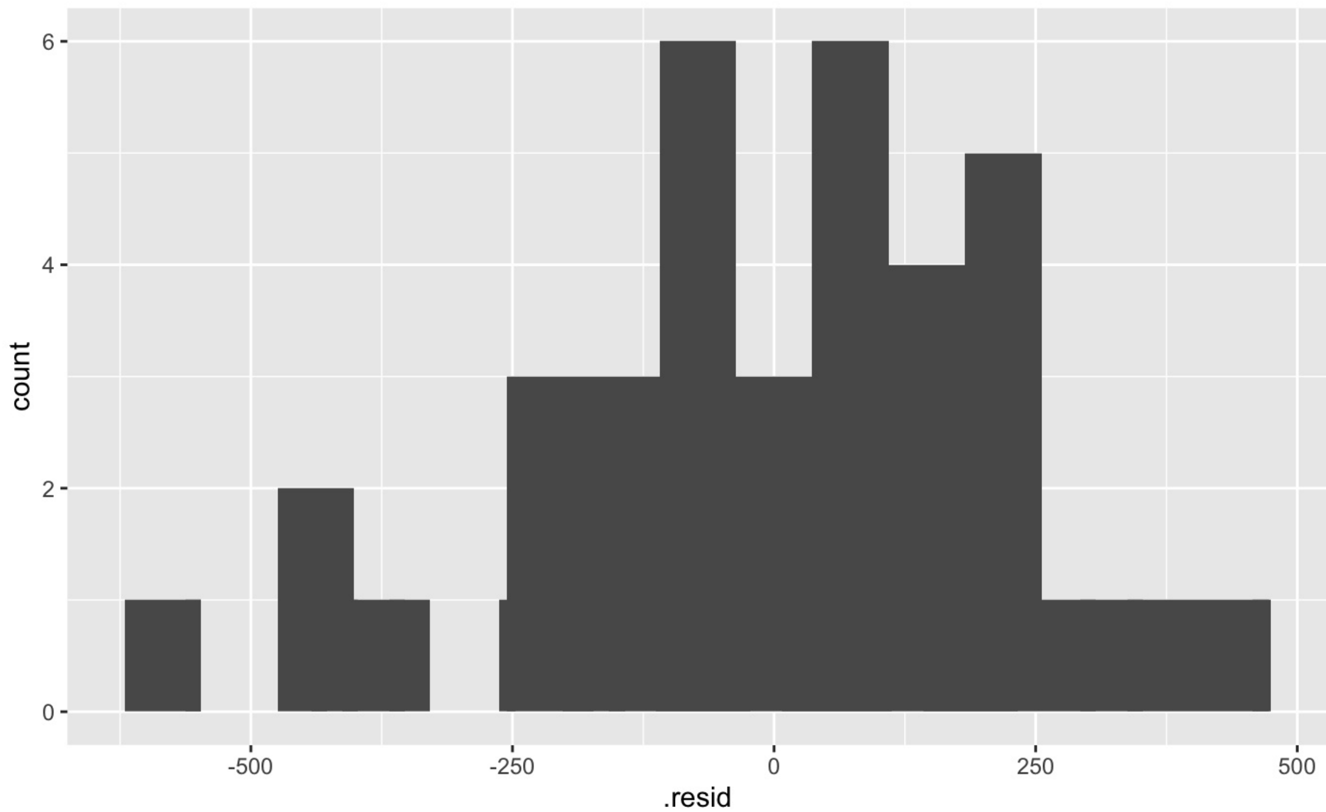
The new RMSE value on test set -

135.07

From the residuals, QQplot, and histogram, we derive that the graph is homoskedastic and the variance is constant, the residuals are normal except at the extremes, and the outliers are not very alarming. U2 and prob are badly performing predictors. The graphs are as follows -







The RMSE value is reduced on the test set, which is good and the following well-performing predictors are retained. The reduced set of predictors -

Ed + Po1 + Ineq

And then we use this scaled data instance to find the output value of the data instance given -

Ed = -0.504

P01 = 1.177

Ineq = 0.175

And the crime predicted using the test model is **1607.626**.

4 - We used RMSE, F-stat, p-value and R-Sqaured to check the quality of our model, found out there was an overfit, and hence used ANOVE analysis to fix this problem. This analysis helped us reduce our predictor set to (Ed, Po1, Ineq) and then we plugged in the

coefficients in the equation -

Crime = 897.96 + 363.09(Ed) + 670.37(P01) + 592.25(Ineq)

- to get the predicted crime as 1607.626.

CODE FOR LINEAR REGRESSION MODEL WITHOUT TEST-TRAIN SPLIT AND SIGNIFICANT PREDICTOR CHECKS

In [ ]:
```
df <- read.table("uscrime.txt",stringsAsFactors = F, header=T)
#head(df,2)
#stats<- basicStats(df)[c("Minimum", "Maximum", "1. Quartile", "3. Quartile", "Mean", "Median","Variance", "Stdev
#kable(stats)

# Visualizations - boxplots and density graphs
melted <- melt(df)
plotted <- ggplot(melted, aes(factor(variable), value))
plotted + geom_boxplot() + facet_wrap(~variable, scale="free")+theme_economist()+scale_colour_economist()
d <- df %>%
  gather() %>%
  ggplot(aes(value)) +
    facet_wrap(~ key, scales = "free") +
    geom_density()+theme_economist()+scale_colour_economist()

set.seed(123)
#Preparing data for scaled test train for later on
df.scaled <- scale(df[,1:15])
df2.scaled<- data.frame(df.scaled)

#80% train
df3<- df2.scaled%>% mutate(Crime=df[,16])
random_row<- sample(1:nrow(df3),as.integer(0.8*nrow(df3)))
trainData = df3[random_row,]
#20% test
testData = df3[-random_row,]

#base lm model
base.model <- lm(Crime ~. ,data = df)
summary(base.model)
print(sprintf("RMSE of Base Model = %0.2f", sigma(base.model)))

#better model
better1 <- lm( Crime ~  M + Ed + Po1 + U2 + Ineq + Prob, data = df)
summary(better1)
print(sprintf("RMSE of Better Model = %0.2f", sigma(better1) ))
model1 <- predict(better1, instance)
```

SOFTWARE OUTPUT

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
```

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 Better Model = 200.69"

CODE FOR INSTANCE EVALUATION

In [ ]:
```
instance <-data.frame(M = 14.0, Ed = 10.0, Po1 = 12.0,, U2 = 3.6, Ineq = 20.1, Prob = 0.040)
pred <- predict(base.model, instance)
```

```
pred
```

SOFTWARE OUTPUT

1304.245

CODE FOR STATISTICAL QUALITY OF MODEL

In [ ]:
```
# Graph for analysis of variance
res.aov.train <- aov(train.model, data = trainData)
summary(res.aov.train)
# 1. Homogeneity of variances graph
plot(res.aov.train, 1)
# 2. Normality of residual graph
plot(res.aov.train, 2)
#histogram of residuals
ggplot(train.model, aes(x=.resid))+geom_histogram(binwidth = 15)+ geom_histogram(bins=15)
```

CODE FOR FINAL MODEL WITH TEST-TRAIN AND VALIDATION QUALITY VALIDATION

In [ ]:
```
#Training model
set.seed(123)
train <-lm( Crime ~  Ed + Po1 + U2 + Ineq + Prob, data = trainData)
summary(train)

#Statistical significance checked and predictors pruned and followed by testing model
set.seed(123)
test<-lm( Crime ~  Ed + Po1  + Ineq , data = testData)
summary(test)
print(sprintf("RMSE of Test Model = %0.2f", sigma(test) ))
```

SOFTWARE OUTPUT

Training Results -

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
```

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

Testing Results -

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
```

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: 0.005655

"RMSE of Test Model = 135.07"

CODE FOR INSTANCE EVALUATION USING FINAL MODEL

In [ ]:

```
testpt <-data.frame(Ed = -0.504, Po1 = 1.177, Ineq = 0.175)
last <- predict(test, testpt)
cat("The predicted crime using the test model = ",last,sep="",fill=TRUE)
```

FINAL SOFTWARE OUTPUT

The predicted crime using the test model = 1607.626

THE END---
