Principal Component Analysis

The goal of this situation was to find a good fit for the *uscrime* data set using Principal Component Analysis, compare it to the best linear regression model found, and predict a new point given a set of new parameters.

Data Pre-Processing

Normality

Before fitting a model, the distribution of each variable needed to be examined. For linear regression, each variable following a normal distribution is an important assumption for fitting a model. To examine the normality of each variable, the Shapiro-Wilk Test for normality was used.

```
norm.df = data.frame()
for(i in 1:16){
  norm.df[i,'col'] = colnames(crime[i])
  norm.df[i,'normality p-value'] = round(shapiro.test(crime[[i]])$p.value,4)
}
norm.df
```

```
##       col normality p-value
## 1       M            0.0361
## 2      So            0.0000
## 3      Ed            0.0032
## 4     Po1            0.0043
## 5     Po2            0.0066
## 6      LF            0.1720
## 7     M.F            0.0038
## 8     Pop            0.0000
## 9      NW            0.0000
## 10     U1            0.0086
## 11     U2            0.2133
## 12 Wealth            0.3375
## 13   Ineq            0.0132
## 14   Prob            0.0179
## 15   Time            0.6132
## 16  Crime            0.0019
```

From this, it can be seen that the variables LF, Time, Wealth, and U2 all have p-values below the threshold of 0.05, showing that they are not normally distributed. When examining the residual diagnostics of the linear regression model fit, if the residuals are not normally distributed then any of these variables used will be standardized to a normal distribution to help remedy this problem. During PCA, the predictor variables would be scaled before fitting a model.

Outliers

Next, the data set needed to be examined for outliers. From the results below it can be seen that there are no collective outliers present, another important assumption for fitting a linear regression model.

```
sum(is.na(crime))
```

```
## [1] 0
```

Next, each variable was examined for point outliers. Significant outliers can worsen the fit of a linear regression model, and therefore any predictor variables with point outliers would be examined when fitting a model. To identify possible point outliers, the Grubbs Test was used to find both high and low outliers for each variable in the *uscrime* data set.

```
outlier.df = data.frame()
for(i in 1:16){
  outlier.df[i,'col'] = colnames(crime[i])
  outlier.df[i,'high outlier p-value'] = round(grubbs.test(x = crime[,i],
                                                           type = 10,
                                                           opposite = F,
                                                           two.sided = F)$p.value,4)
  outlier.df[i,'low outlier p-value'] = round(grubbs.test(x = crime[,i],
                                                          type = 10,
                                                          opposite = T,
                                                          two.sided = F)$p.value,4)
}
outlier.df
```

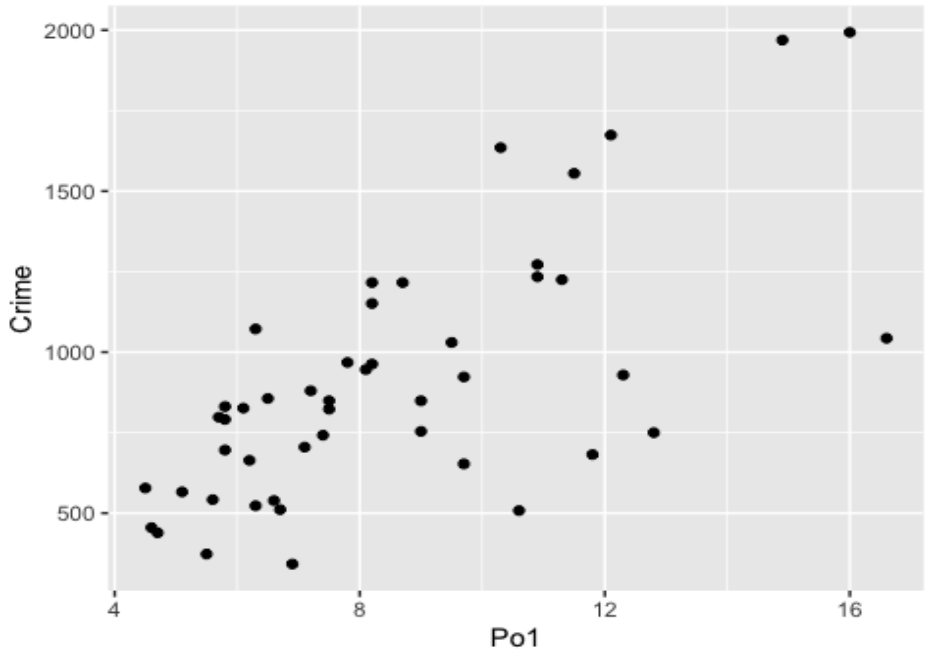
```
##       col high outlier p-value low outlier p-value
## 1       M               0.0303              1.0000
## 2      So               1.0000              1.0000
## 3      Ed               1.0000              1.0000
## 4     Po1               0.1083              1.0000
## 5     Po2               0.1009              1.0000
## 6      LF               0.9608              1.0000
## 7     M.F               0.0405              1.0000
## 8     Pop               0.0051              1.0000
## 9      NW               0.0223              1.0000
## 10     U1               0.1784              1.0000
## 11     U2               0.0702              1.0000
## 12 Wealth               0.2643              1.0000
## 13   Ineq               0.8519              1.0000
## 14   Prob               0.0166              1.0000
## 15   Time               0.2682              0.9063
## 16  Crime               0.0789              1.0000
```

From this, it can be seen that the predictor variables M, M.F, Pop, NW, Prob all contained possible high point outliers, as their p-values for high outliers were below the threshold of

0.05. If any of these variables are selected for the final set of predictors, models with and without the outliers will be fit to examine the effect these point outliers may have.

Initial Plot

Next, an initial plot was created for the *uscrime* data set to examine if a linear relationship was present in the data.



From this, a linear relationship between these two variables can be seen, which meets another important assumption for fitting a linear regression model.

Best Linear Regression Model

Through examination of relevant predictor variables and possible outliers, the best linear regression model (seen below) was fit. Only the relevant predictor variables were kept, and high point outliers in variables M and Prob were removed.

```
best.reg.fit = lm(formula = Crime ~ Ineq + Ed + Prob + M + U2 + Po1,
                  data = clean.crime)
```

The summary for this model can be seen below:

```
##
## Call:
## lm(formula = Crime ~ Ineq + Ed + Prob + M + U2 + Po1, data = clean.crime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -467.62  -92.55   -3.83  110.23  554.15
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5227.79     911.44  -5.736 1.31e-06 ***
## Ineq           70.30      14.44   4.868 2.00e-05 ***
## Ed            200.25      45.86   4.366 9.39e-05 ***
## Prob        -5520.41    2113.34  -2.612  0.01281 *
## M             120.57      36.44   3.309  0.00206 **
## U2             92.39      41.18   2.244  0.03075 *
## Po1           109.17      15.17   7.195 1.34e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 200.9 on 38 degrees of freedom
## Multiple R-squared:  0.7769, Adjusted R-squared:  0.7416
## F-statistic: 22.05 on 6 and 38 DF,  p-value: 5.479e-11
```

Accuracy Metrics

Next, the fit of this model was examined. For this process, the SSE, AIC, and BIC were all examined to gain the most information possible about the fit of the model.

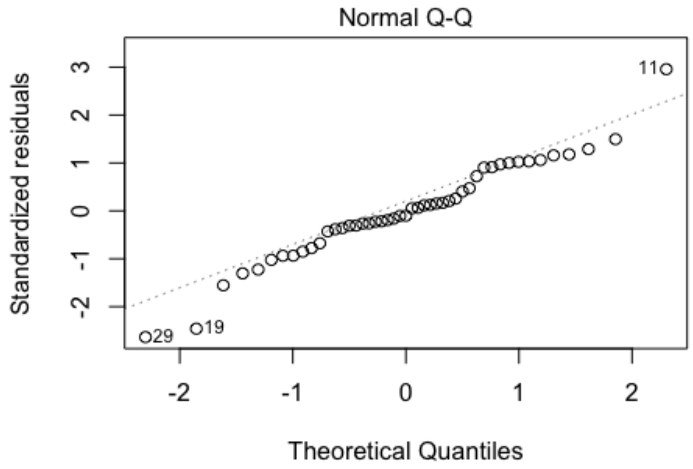
```
# SSE
best.reg.fit.SSE = sum(best.reg.fit$residuals^2)
best.reg.fit.SSE

## [1] 1534109

# AIC
best.reg.fit.AIC = AICc(best.reg.fit)
best.reg.fit.AIC

## [1] 617.3604
## attr(,"nall")
## [1] 45

# BIC
best.reg.fit.BIC = BIC(best.reg.fit)
best.reg.fit.BIC

## [1] 627.8137

# Adj R Squared
best.reg.fit.sum$adj.r.squared

## [1] 0.7416166
```

From this, it can be seen that the full model accounts for 74.16% of the variance in the data.

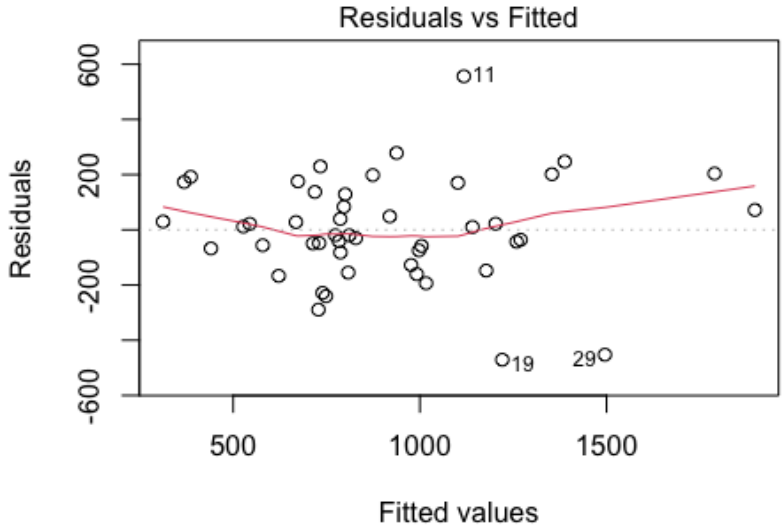
Quality of Fit

After determining the best set of predictors for the model, the residual diagnostics needed to be examined to ensure the assumptions for linear regression were met.

Below, it can be seen that while perhaps long-tailed and with a few possible outliers, the residuals for this model mostly followed a normal distribution.



Additionally, it can be seen below that the residuals were fairly homogeneous with a few possible outliers present.



Since the assumptions appeared to be met from the residual diagnostics, this linear regression model would be used to compare the model found from Principal Component Analysis.

Principal Component Analysis

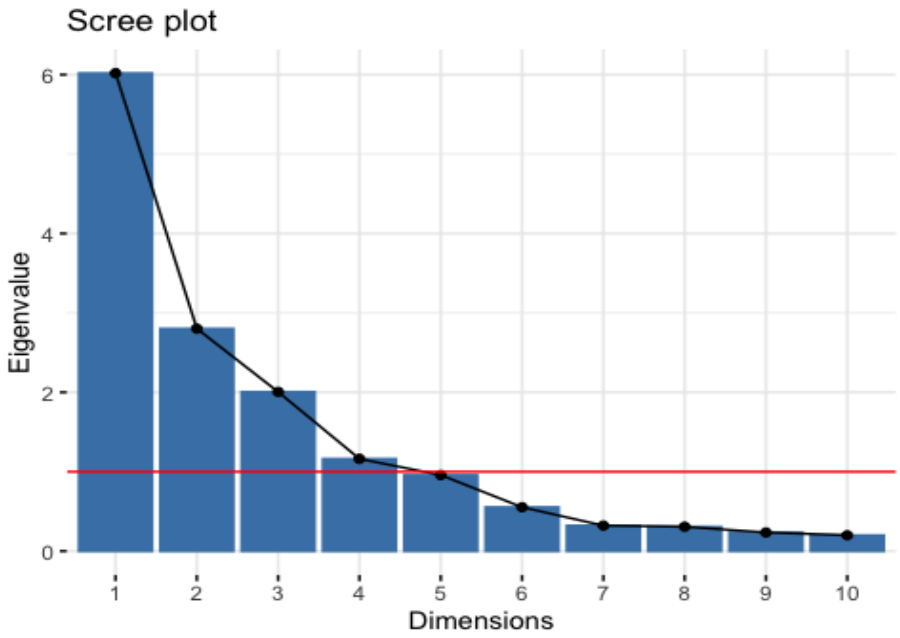
To begin, PCA was conducted for all variables except the response variable, with data scaling.

```
crime.pca = prcomp(crime[,1:15], scale = T)
```

The summary from PCA can be seen below:

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6     PC
7
## Standard deviation     2.4534 1.6739 1.4160 1.07806 0.97893 0.74377 0.5672
9
## Proportion of Variance 0.4013 0.1868 0.1337 0.07748 0.06389 0.03688 0.0214
5
## Cumulative Proportion  0.4013 0.5880 0.7217 0.79920 0.86308 0.89996 0.9214
2
##                            PC8     PC9    PC10    PC11    PC12    PC13   P
C14
## Standard deviation     0.55444 0.48493 0.44708 0.41915 0.35804 0.26333 0.2
418
## Proportion of Variance 0.02049 0.01568 0.01333 0.01171 0.00855 0.00462 0.0
039
## Cumulative Proportion  0.94191 0.95759 0.97091 0.98263 0.99117 0.99579 0.9
997
##                           PC15
## Standard deviation     0.06793
## Proportion of Variance 0.00031
## Cumulative Proportion  1.00000
```

Next, the eigenvalues from PCA were examined. Since factors with an eigenvalue of 1 account for as much variance as a single variable, any dimensions below this threshold would not be used. The scree plot can be seen below:



From this, it can be seen that roughly only through the 5th dimension met this threshold. For this reason, only the first five dimensions of PCA would be used to fit a regression model. This regression model can be seen below:

Model Fitting

```
# Fit Model
pca = crime.pca$x[,1:5]
crime.pca.full = cbind(pca,crime[,16])
pca.fit =  lm( V6 ~., data = as.data.frame(crime.pca.full))
```

Below is the summary from the PCA regression model:

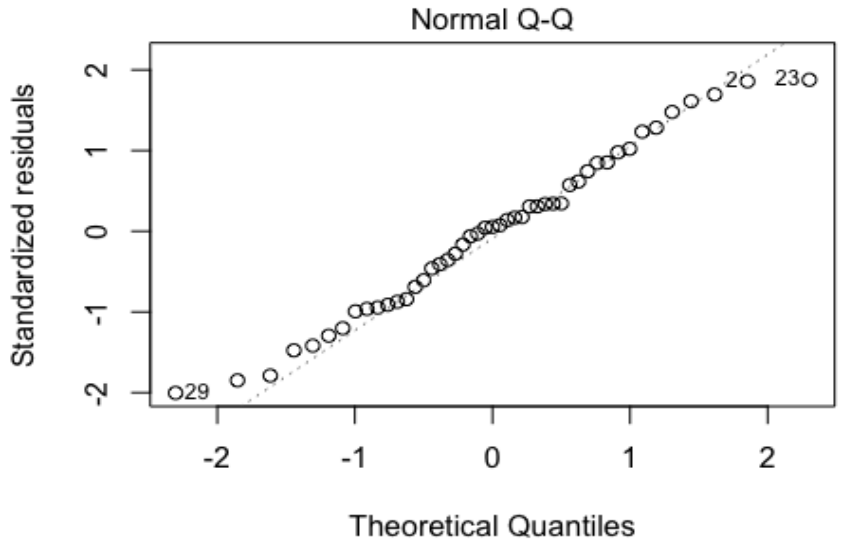
```
##
## Call:
## lm(formula = V6 ~ ., data = as.data.frame(crime.pca.full))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -420.79 -185.01   12.21  146.24  447.86
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   905.09      35.59  25.428  < 2e-16 ***
## PC1            65.22      14.67   4.447 6.51e-05 ***
## PC2           -70.08      21.49  -3.261  0.00224 **
## PC3            25.19      25.41   0.992  0.32725
## PC4            69.45      33.37   2.081  0.04374 *
## PC5          -229.04      36.75  -6.232 2.02e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 244 on 41 degrees of freedom
## Multiple R-squared:  0.6452, Adjusted R-squared:  0.6019
## F-statistic: 14.91 on 5 and 41 DF,  p-value: 2.446e-08
```

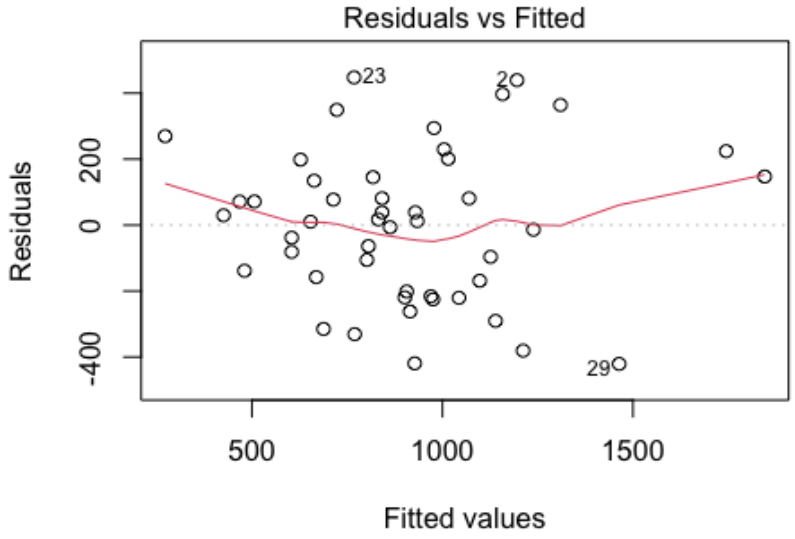
Residual Diagnostics

After fitting the PCA regression model, the residual diagnostics needed to be examined to ensure the assumptions for linear regression were met.

Below, it can be seen that the residuals were relativelt normally distirbuted with a few possible outliers.



Additionally, it can be seen below that the residuals were fairly homogeneous with a few possible outliers present.



Because of this, it was determiend that the assumptions were met for the PCA regression model.

Accuracy Metrics

Next the accuracy metrics from the PCA regression model were compared to that of the best linear regression model fit previously.

```
pca.fit.SSE = sum(pca.fit$residuals^2)
pca.fit.SSE

## [1] 2441394

best.reg.fit.SSE

## [1] 1534109
```

From this, it could be seen that the linear regression model had a lower SSE than that of the PCA regression model.

```
pca.fit.AIC = AIC(pca.fit)
pca.fit.AIC

## [1] 657.703

best.reg.fit.AIC

## [1] 617.3604
## attr(,"nall")
## [1] 45

exp((best.reg.fit.AIC - pca.fit.AIC) / 2)

## [1] 1.736614e-09
## attr(,"nall")
## [1] 45
```

Similarly, the AIC of this model was lower, and there was an incredibly small probability that the PCA regression model was more accurate.

```
pca.fit.BIC = BIC(pca.fit)
pca.fit.BIC

## [1] 670.6541

best.reg.fit.BIC

## [1] 627.8137

pca.fit.BIC - best.reg.fit.BIC

## [1] 42.84039
```

Finally, the BIC of this model was also lower and it was very likely that the linear regression model was better than the PCA regression model.

```
pca.fit.sum$adj.r.squared
```

```
## [1] 0.601925
```

```
best.reg.fit.sum$adj.r.squared
```

```
## [1] 0.7416166
```

Additionally, the model linear regression model accounted for more variation in the data.

It appears that the best linear regression model found previously was more accurate than the PCA regression model, however, predictions would still be calculated using the PCA regression model to examine its results.

Unscaling the Data

Before creating predictions, however, the PCA model had to be unscaled. To do so, the intercept and coefficient values for this model were unscaled.

```
# Store Scaled Values
Intercept.Scaled = pca.fit$coefficients[1]
Coefficients.Scaled = pca.fit$coefficients[2:6]
# Unscale Values
a = crime.pca$rotation[,1:5]%*%Coefficients.Scaled
Coefficients.Unscaled = a/crime.pca$scale
Intercept.Unscaled = Intercept.Scaled-sum(a*crime.pca$center/crime.pca$scale)
```

The unscaled coefficients and intercept for the PCA regression model can be seen below:

```
Coefficients.Unscaled
```

```
##                 [,1]
## M       4.837374e+01
## So      7.901922e+01
## Ed      1.783120e+01
## Po1     3.948484e+01
## Po2     3.985892e+01
## LF      1.886946e+03
## M.F     3.669366e+01
## Pop     1.546583e+00
## NW      9.537384e+00
## U1      1.590115e+02
## U2      3.829933e+01
## Wealth  3.724014e-02
## Ineq    5.540321e+00
## Prob   -1.523521e+03
## Time    3.838779e+00
```

```
Intercept.Unscaled

## (Intercept)
##   -5933.837
```

Prediction

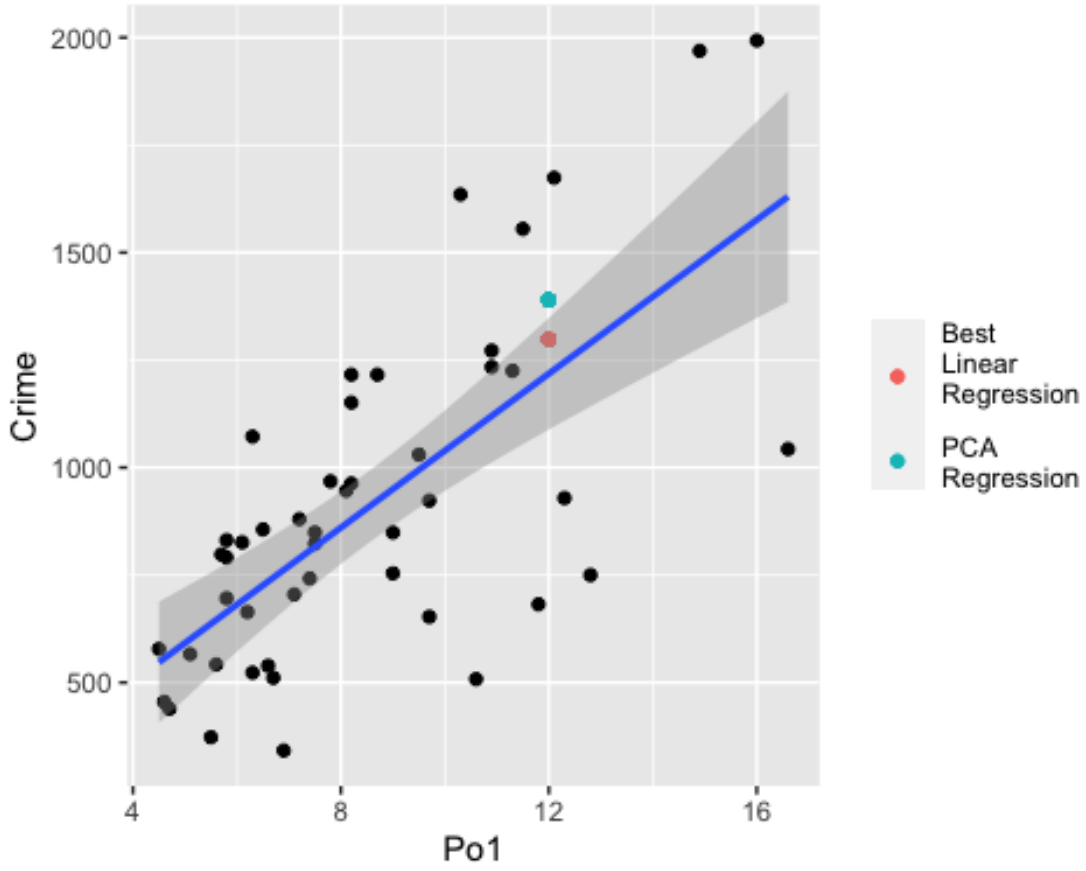
After unscaling the regression model, a predcition could be created with a new set of data points.

```
new.param = 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 = 0.04,
                       Time = 39.0)
```

The results of the prediction can be seen below (rounded to the nearest integer):

```
pca.pred = round(Intercept.Unscaled+as.matrix(new.param)%*%Coefficients.Unsca
led)
pca.pred

##      [,1]
## [1,] 1389
```

Predictions created with the same new data points can be seen for the PCA regression model and best linear regression model below:



From this, it can be seen that the prediction created form the best linear regression mode was closer to the regression line than the prediction of the PCA regression model.