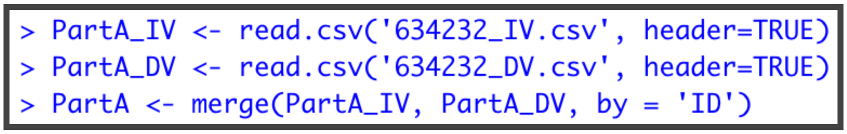
Shreyan Wankavala 112634232

Problem A:

In my research, I am performing an analysis on a set of data contained within two separate files: one containing the independent variable values, and the other containing the dependent variable values. This set of data will be processed statistically with the use of functions given in RStudio. By the end of the analysis, the number of observations and the fraction of missing data in the independent and dependent variables should be known, as well as a complete imputation of the missing data. I will also have additional information such as specific confidence intervals and an estimated regression line for the data set. Since the values are in two separate files, they must first be sorted and merged before any other steps can be performed, and they should be merged numerically based on their given ID. Another step is to find the missing data throughout the set, denoted by “NA”. An algorithm will be used to impute the missing values based on the present independent or dependent variable value.

The program used to complete the statistical analysis was RStudio. I chose RStudio because it provided the methods I needed, as well as an easy to use interface fitting for what I needed to do. I started by merging the two files using the merge() function.



```
> PartA_IV <- read.csv('634232_IV.csv', header=TRUE)
> PartA_DV <- read.csv('634232_DV.csv', header=TRUE)
> PartA <- merge(PartA_IV, PartA_DV, by = 'ID')
```

Figure 1

As seen in Figure 1, the data sets for the IV and DV are assigned to PartA_IV and PartA_DV respectively, using the read.csv() function in R. They are then merged, sorted by ID, and assigned to PartA. At this point in the procedure, there existed one set of data, but it was not yet complete because of the NA values that appeared throughout it. Altogether, there were 578 IDS, including 515 IDs with an independent variable value, 502 IDs with a dependent variable value, 462 IDs with both an independent and dependent variable value, and 555 IDs with at least one independent or dependent variable value. Expanding on this, there were 23 cases for which both the IV and DV values were missing, and 93 cases for which only one, either the IV or the DV value, was missing. This information was found using the pattern() function in R, as well as using variations of the sum() and is.na() function. In Figure 2, the pattern function is shown. We can see that there are 23 IDs for which there are no cases of either IV or ID values on the left, 462 cases of IDs with both an IV and DV value, 63 cases where the IV is missing, and 76 cases where the DV is missing. There is also an example of the sum() and is.na() functions in use in Figure 3, where the sum of the IDs for which there is not an NA under the IV and not an NA under the DV is outputted.

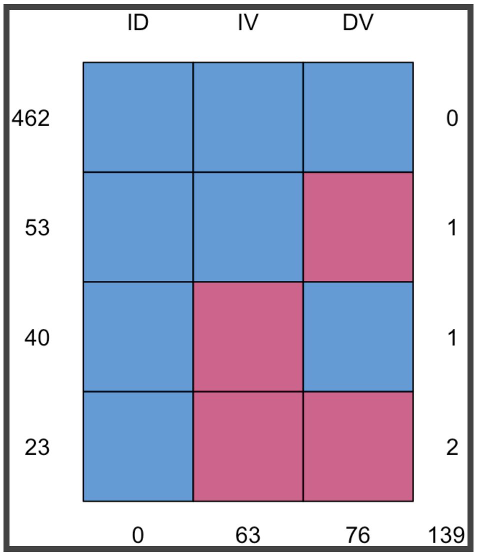
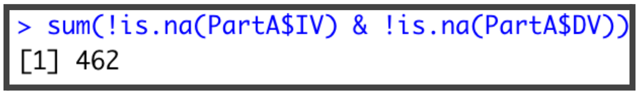


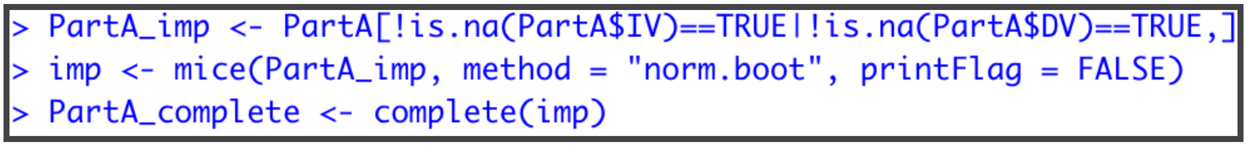
Figure 2



```
> sum(!is.na(PartA$IV) & !is.na(PartA$DV))
[1] 462
```

Figure 3

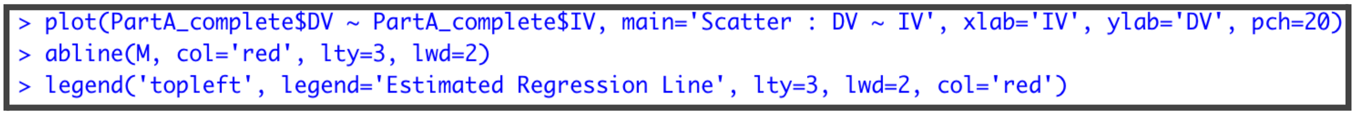
The next step was to impute the missing data using the R function called mice(). Before doing so, the 23 IDs for which there were no IV or DV values were removed. This is because it is impossible to find the missing values when nothing is given at all, as opposed to if only one of the two values is missing. They were removed using the first line of code shown in Figure 4. After that was done, the mice() function as well as linear regression using bootstrapping using the norm.boot function were applied. PartA_complete contains the resulting complete data set after imputing the missing values. The final data set contained a total of 555 IDs, each with its own IV and DV.



```
> PartA_imp <- PartA[!is.na(PartA$IV)==TRUE|!is.na(PartA$DV)==TRUE,]
> imp <- mice(PartA_imp, method = "norm.boot", printFlag = FALSE)
> PartA_complete <- complete(imp)
```

Figure 4

With a complete data set, the next step was to then fit a simple linear regression model onto its values. The lm() function, which is used to fit linear models, was used on the complete set and was assigned to the variable M. With this variable, a scatter plot with an estimated regression line was made. The process can be seen in Figure 5, and the finished product in Figure 6.



```
> plot(PartA_complete$DV ~ PartA_complete$IV, main='Scatter : DV ~ IV', xlab='IV', ylab='DV', pch=20)
> abline(M, col='red', lty=3, lwd=2)
> legend('topleft', legend='Estimated Regression Line', lty=3, lwd=2, col='red')
```

Figure 5

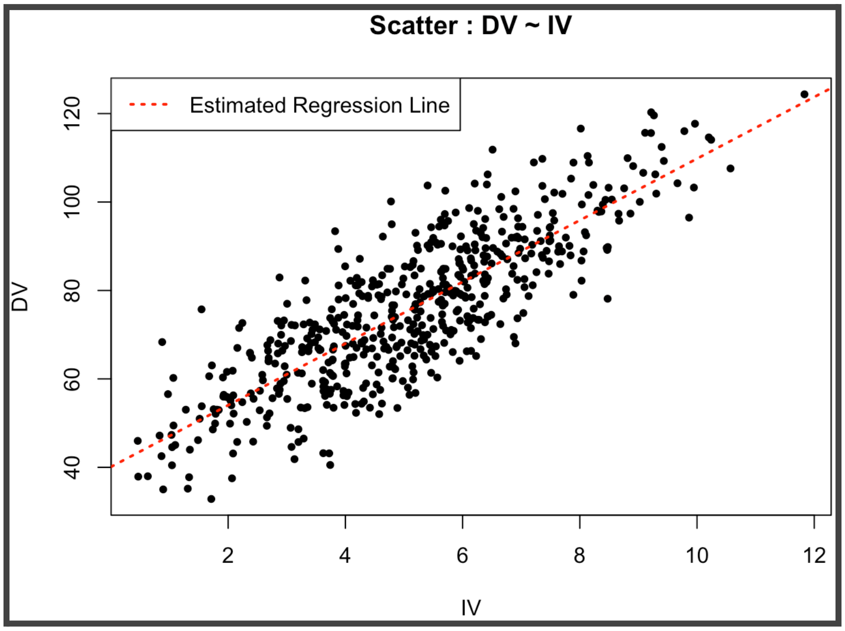


Figure 6

No transformation is needed for this scatter plot, since it appears linear when shown on its own. When I used the summary() function in R on the variable M however, it is comparable to the F-statistic value, which is 1191 on 1 and 553 degrees of freedom, to the p-value which is <

2.2e-16. Since it is significantly smaller than the F-statistic value and less than 0.5, it can be concluded that a full model will offer a much better fit than the one used in Figure 6, or in other words, a lack of fit test must be completed. The ggplot() function is used from the ggplot2 library to make a new linear regression model, shown in Figure 7.

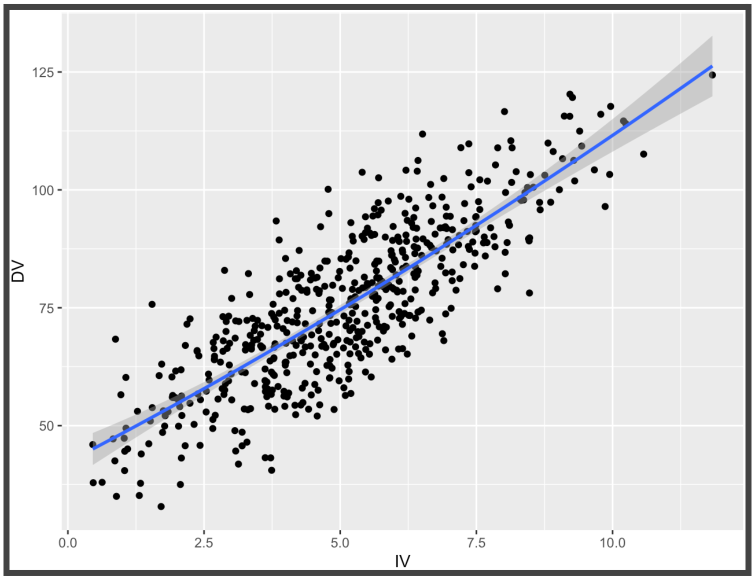
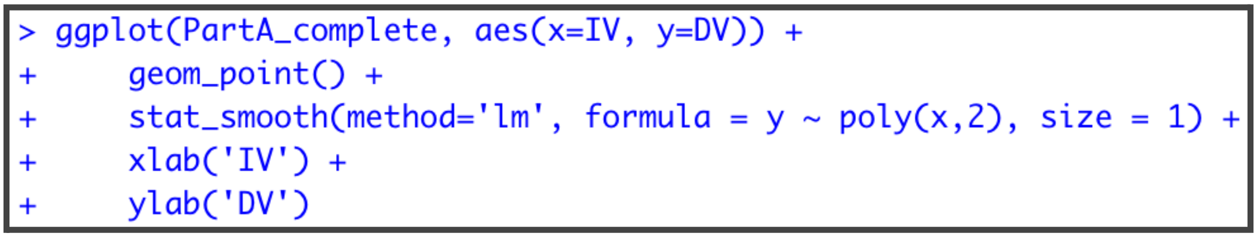


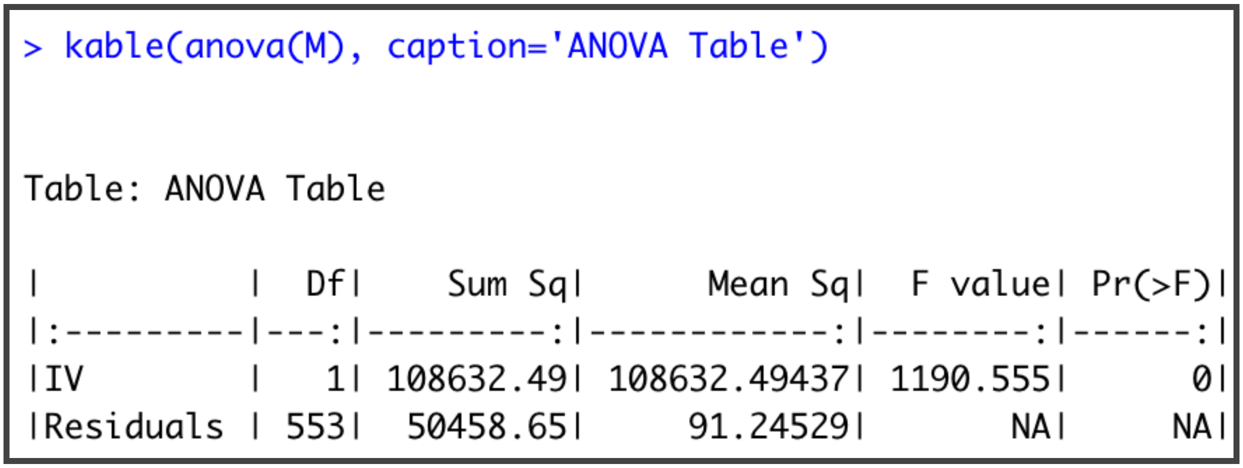
Figure 7



```
> ggplot(PartA_complete, aes(x=IV, y=DV)) +
+     geom_point() +
+     stat_smooth(method='lm', formula = y ~ poly(x,2), size = 1) +
+     xlab('IV') +
+     ylab('DV')
```

Figure 8

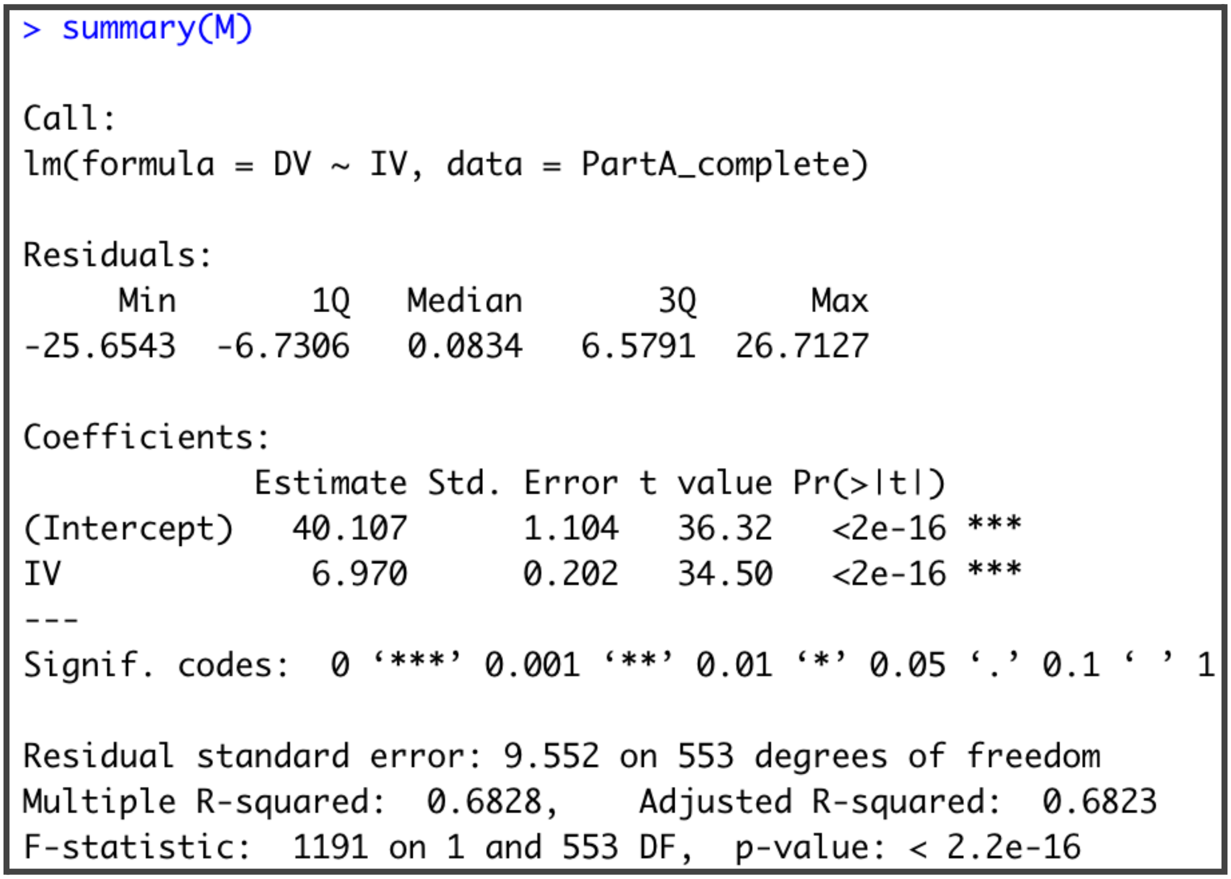
For my results, multiple tables and graphs were made and analyzed. The first table that was made was the ANOVA table, or the analysis of variance table. This was made using the kable() function as seen in Figure 9 (which also includes the ANOVA table). The F value for the IV is 1190.555.



```
> kable(anova(M), caption='ANOVA Table')


Table: ANOVA Table

|          |  Df|    Sum Sq|      Mean Sq|  F value| Pr(>F)|
|:---------|---:|---------:|------------:|--------:|------:|
|IV        |   1| 108632.49| 108632.49437| 1190.555|      0|
|Residuals | 553|  50458.65|     91.24529|       NA|     NA|
```

Figure 9

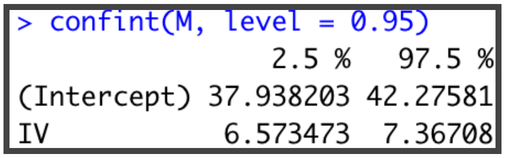
The summary(M) was also used on M to return a table of values as seen in Figure 10. In the summary table, we have values such as R-squared, which enables us to see the fraction of the variation of the dependent variable that was explained. We can either use the Multiple or Adjusted R-Squared value, but it is better to use the adjusted value in this scenario since we only have one IV. The value is 0.6823, which means that approximately 68.23% of the variance of the dependent variable was explained, while the other 31.77% could be due to random variability.



```
> summary(M)

Call:
lm(formula = DV ~ IV, data = PartA_complete)

Residuals:
     Min       1Q   Median       3Q      Max 
-25.6543  -6.7306   0.0834   6.5791  26.7127 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   40.107      1.104   36.32   <2e-16 ***
IV             6.970      0.202   34.50   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 9.552 on 553 degrees of freedom
Multiple R-squared:  0.6828,	Adjusted R-squared:  0.6823 
F-statistic:  1191 on 1 and 553 DF,  p-value: < 2.2e-16
```

Figure 10

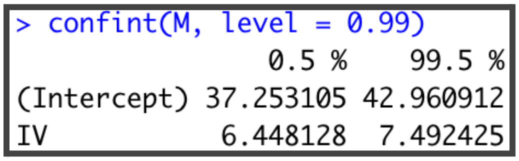
The confidence interval of the slope was also calculated using the confint() function. The 95% confidence interval of the slope is given in Figure 10. We can see that it is 6.57 to 7.37.



```
> confint(M, level = 0.95)
                2.5 %   97.5 %
(Intercept) 37.938203 42.27581
IV           6.573473  7.36708
```

Figure 10

The 99% confidence interval of the slope is given in Figure 11. We can see that it is 6.45 to 7.49.

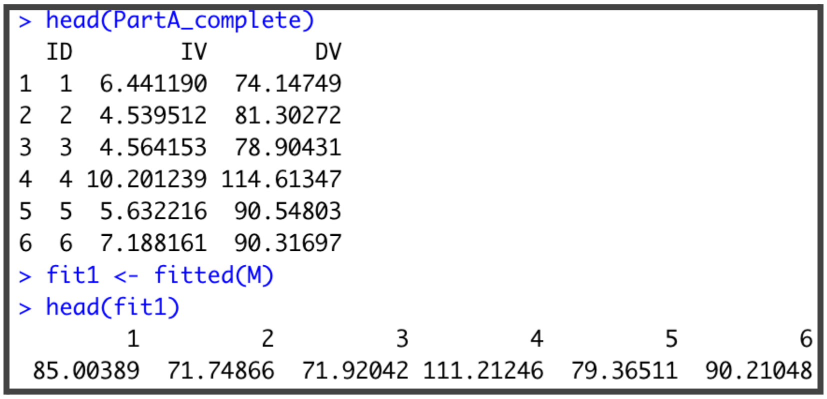


```
> confint(M, level = 0.99)
                0.5 %    99.5 %
(Intercept) 37.253105 42.960912
IV           6.448128  7.492425
```

Figure 11

We can also reject the null hypothesis that the slope was zero after looking at the graph, which has a nonzero slope. This means that there was a significant linear relationship between the independent and dependent variables.

The conclusion can be made that there was a significant linear relationship between the independent and dependent variables. This is because the slope was nonzero by looking at the linear regression model that was made based on the data. If there was no relationship between the two variables, then the slope would be zero. The association between the variables was strong as seen by our R-squared value, which showed us that approximately 68.23% of the variance of the dependent variable was explained. More specifically, that amount of the variation in the dependent variable was explained by the independent variable in the data set. On top of this, I was able to use the fitted() function in R to find fitted values for the linear regression model. This was in conjunction with the scatter plot and line that were already graphed. In Figure 12, we see that the first six fitted values, corresponding to the first six values in the data set, are outputted.



```
> head(PartA_complete)
  ID        IV        DV
1  1  6.441190  74.14749
2  2  4.539512  81.30272
3  3  4.564153  78.90431
4  4 10.201239 114.61347
5  5  5.632216  90.54803
6  6  7.188161  90.31697
> fit1 <- fitted(M)
> head(fit1)
        1         2         3         4         5         6 
 85.00389  71.74866  71.92042 111.21246  79.36511  90.21048 
```

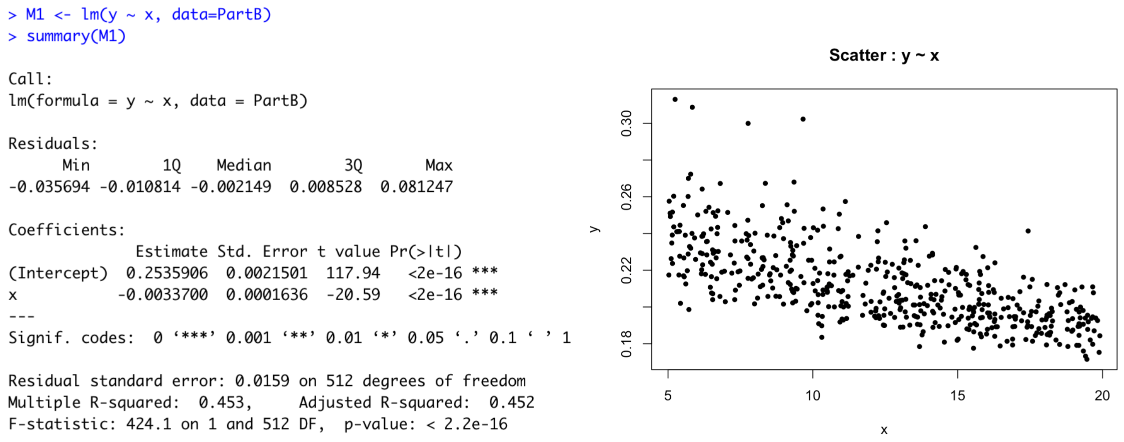
Figure 12

The merging, imputation of missing values, and analysis of this data set was successful at the end.

Problem B:

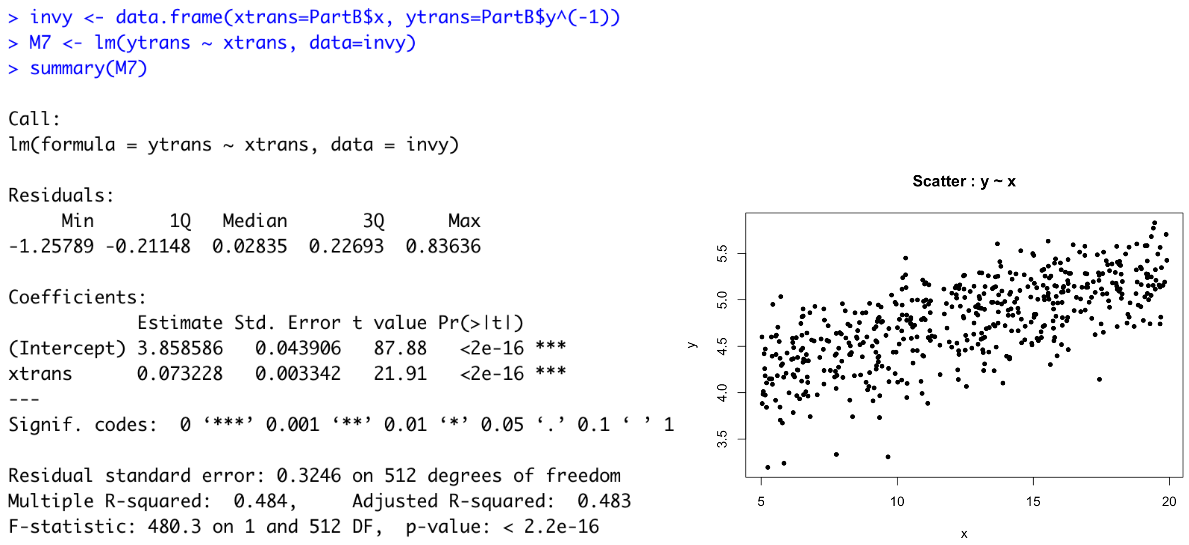
For Problem B, a new set of data needed to be analyzed. There was just one file this time, which included both the independent and dependent variable values, so no merging was required. Given the data set, I needed to recover the function that was used to generate the dependent variable value based on the value of the independent variable. Before I could do this however, I first needed to find if the data set required a transformation of either the independent variable, dependent variable, or both. After that, I figured out whether or not there were repeated or near repeated independent variable values; the ones that were needed to be binned into one level. Finally, a lack of fit test needed to be done on the data. There were many steps taken along the way, but these were necessary if I wanted to find a properly fitted regression model for the data set that was given. Using the built in summary and ANOVA table functions in r, I was also able to analyze information regarding the variance, confidence intervals, and r-squared values.

The first step I took was to calculate the r-squared value of the data set in the process of finding out whether or not a transformation of the IV or DV was needed. The r-squared value of the original data set was 0.452, which means that approximately 45.2% of the variation of the dependent variable was explained. From here, I needed to guess and check transformations of the graph to see if I could find a more linear graph of the data. After trying and graphing a transformation of the log10 of x and y, inverse of x and y, square root of x and y, square of x and y, exponential of x and y, the highest r-squared value was resulted when I found only the inverse of the y value (each one was done separately and together). We can see the original summary and scatter plot for the data in Figure 1, and the transformation of the dependent variable in Figure 2. We can see that the r-squared value for the transformation is 48.3% instead of 45.2%, which is a very slight increase. In terms of the graph itself, there does not seem to be much change other than the sign of its slope.



```
> M1 <- lm(y ~ x, data=PartB)
> summary(M1)

Call:
lm(formula = y ~ x, data = PartB)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.035694 -0.010814 -0.002149  0.008528  0.081247 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.2535906  0.0021501  117.94   <2e-16 ***
x           -0.0033700  0.0001636  -20.59   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0159 on 512 degrees of freedom
Multiple R-squared:  0.453,	Adjusted R-squared:  0.452 
F-statistic: 424.1 on 1 and 512 DF,  p-value: < 2.2e-16
```

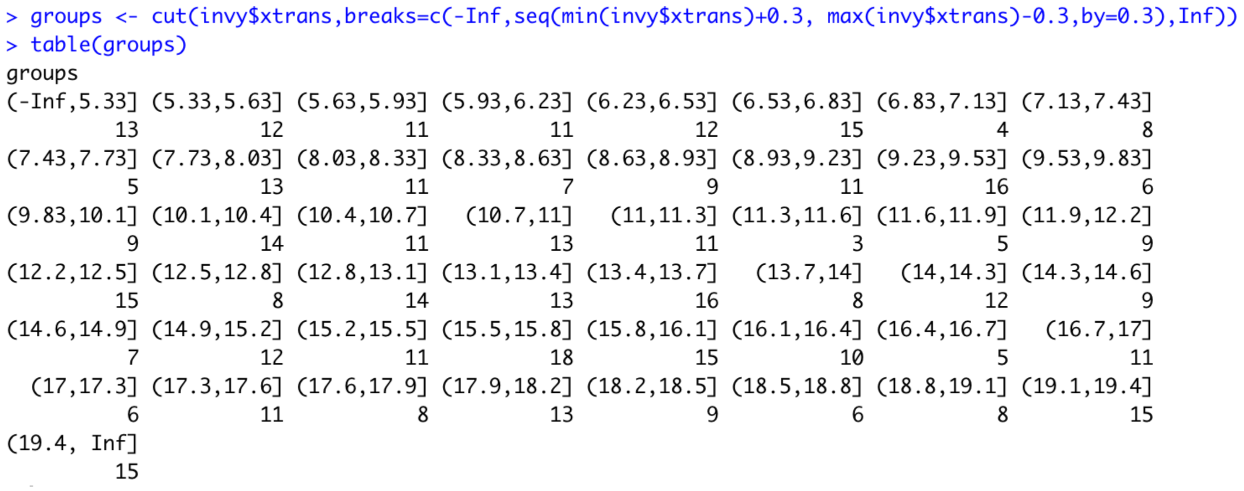
Figure 1



```
> invy <- data.frame(xtrans=PartB$x, ytrans=PartB$y^(-1))
> M7 <- lm(ytrans ~ xtrans, data=invy)
> summary(M7)

Call:
lm(formula = ytrans ~ xtrans, data = invy)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.25789 -0.21148  0.02835  0.22693  0.83636 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 3.858586   0.043906   87.88   <2e-16 ***
xtrans      0.073228   0.003342   21.91   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3246 on 512 degrees of freedom
Multiple R-squared:  0.484,	Adjusted R-squared:  0.483 
F-statistic: 480.3 on 1 and 512 DF,  p-value: < 2.2e-16
```

Figure 2

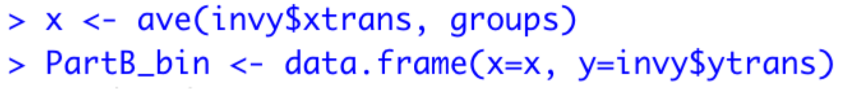
After finding the transformations of the independent and dependent variables, the next step was to bin the data into groups. The purpose of this was to look at repetitions or near repetitions of the independent variable and bin them into the same level. This was done using the cut() function in R, which is useful for breaking variables up into categories. The process can be seen in Figure 3, where the data is put into groups and then displayed using the table() function.



```
> groups <- cut(invy$xtrans,breaks=c(-Inf,seq(min(invy$xtrans)+0.3, max(invy$xtrans)-0.3,by=0.3),Inf))
> table(groups)
groups
(-Inf,5.33] (5.33,5.63] (5.63,5.93] (5.93,6.23] (6.23,6.53] (6.53,6.83] (6.83,7.13] (7.13,7.43] 
         13          12          11          11          12          15           4           8 
(7.43,7.73] (7.73,8.03] (8.03,8.33] (8.33,8.63] (8.63,8.93] (8.93,9.23] (9.23,9.53] (9.53,9.83] 
          5          13          11           7           9          11          16           6 
(9.83,10.1] (10.1,10.4] (10.4,10.7]   (10.7,11]   (11,11.3] (11.3,11.6] (11.6,11.9] (11.9,12.2] 
          9          14          11          13          11           3           5           9 
(12.2,12.5] (12.5,12.8] (12.8,13.1] (13.1,13.4] (13.4,13.7]   (13.7,14]   (14,14.3] (14.3,14.6] 
         15           8          14          13          16           8          12           9 
(14.6,14.9] (14.9,15.2] (15.2,15.5] (15.5,15.8] (15.8,16.1] (16.1,16.4] (16.4,16.7]   (16.7,17] 
          7          12          11          18          15          10           5          11 
  (17,17.3] (17.3,17.6] (17.6,17.9] (17.9,18.2] (18.2,18.5] (18.5,18.8] (18.8,19.1] (19.1,19.4] 
          6          11           8          13           9           6           8          15 
(19.4, Inf] 
         15 
```

Figure 3

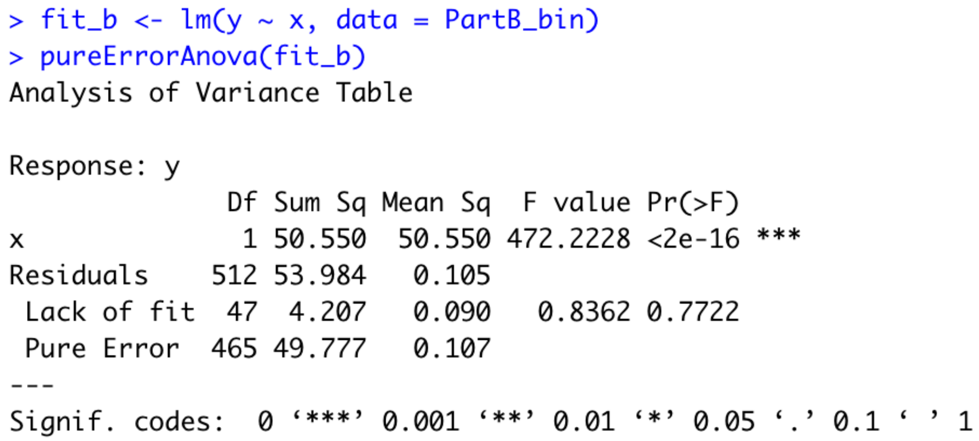
Afterward, the data was binned. This was only done to the independent variable because the dependent variable does not need to be binned. The groups were used for the x value, while the y values were kept the same from my original transformation This can be seen in Figure 4, where the ave() function was used to average the groups I created, and then the averages were used as the independent variables in the bin on the second line of code.



```
> x <- ave(invy$xtrans, groups)
> PartB_bin <- data.frame(x=x, y=invy$ytrans)
```

Figure 4

The last step was to apply a lack of fit test to the data. The built-in pureErrorAnova() function in R enabled me to do this easily. In Figure 5, the values under the F-value and p-value were compared for the row 'labeled lack of fit'. As seen in the figure, the F-value is 0.8362, while the p-value is 0.7722. Since it is lower, we can conclude that there is a lack of fit in our regression model.



```
> fit_b <- lm(y ~ x, data = PartB_bin)
> pureErrorAnova(fit_b)
Analysis of Variance Table

Response: y
              Df Sum Sq Mean Sq  F value Pr(>F)    
x              1 50.550  50.550 472.2228 <2e-16 ***
Residuals    512 53.984   0.105                    
 Lack of fit  47  4.207   0.090   0.8362 0.7722    
 Pure Error  465 49.777   0.107                    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 5

To find a new fitted function, the ggplot() and geom_point() functions were used in R. A new regression model was made from the binned data, which is shown in Figure 6.

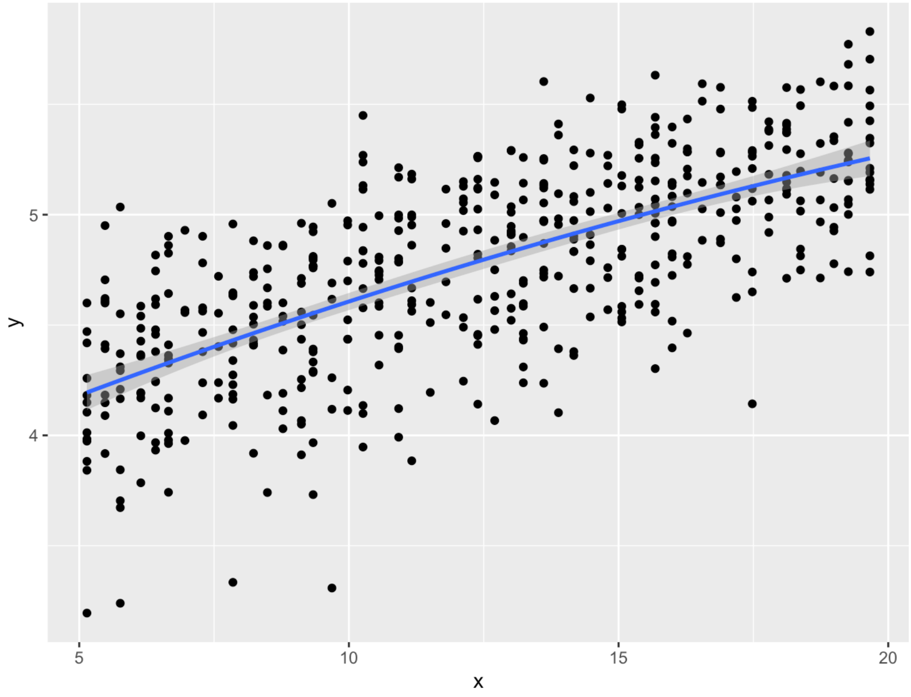
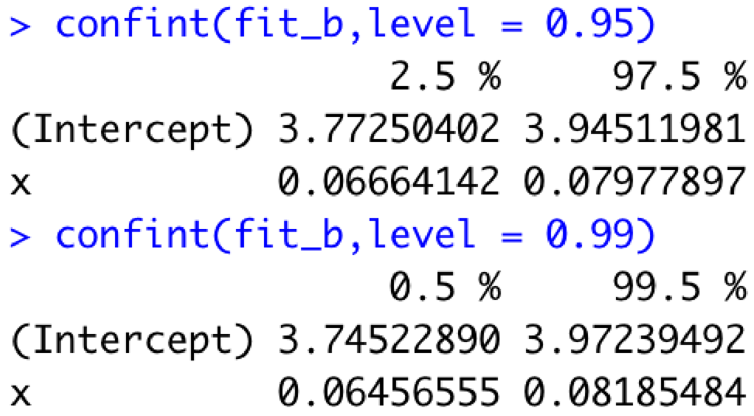


Figure 6

In the beginning of this analysis, I transformed the data by looking for the highest r-squared value I could find, which was 0.483. This value represents the fraction of the variation of the dependent variable that was explained, in this case being 48.3%. In Figure 5, an Analysis of Variance table was given, or an ANOVA table. Using this table is how we found the F and p values, and completed our lack of fit test. We can reject the null hypothesis that the slope was zero because the slope in our regression was not zero. What this means is that there was a strong relationship between the x and y values in our data set, which was also seen by the r-squared value which was nearing 50%. The 95% and 99% confidence intervals were also calculated. They can both be seen in Figure 7. The 95% confidence interval is shown to be 0.06664142 to 0.07977897, and the 99% confidence interval is shown to be 0.06456555 to 0.08185484.



```
> confint(fit_b,level = 0.95)
                 2.5 %     97.5 %
(Intercept) 3.77250402 3.94511981
x           0.06664142 0.07977897
> confint(fit_b,level = 0.99)
                 0.5 %     99.5 %
(Intercept) 3.74522890 3.97239492
x           0.06456555 0.08185484
```

Figure 7

I also found the correlation value between the independent and dependent variables, after they were binned. This value was about 0.6954. This value is close to positive 1, which means that the two variables were positively correlated.

The conclusion can be drawn that there was a significant linear relationship between the independent and dependent variables, x and y respectively. This is because the slope was nonzero by looking at the linear regression model that was made based on the data. The association between the variables was strong as seen by our r-squared value, which showed us that approximately 48.3% of the variance of the dependent variable was explained. I produced a fitted function after finding a transformation for the data, making groups out of the data values and averaging certain x values, and binning the data set. I also had to complete a lack of fit test to compare the F-value to the p-value, which turned out to be higher than the p-value which means the test failed. It was proven that R is a great tool to perform a data analysis of this type, as it provided the necessary functions for what I needed, and I accomplished everything I sought out to do for this data analysis project successfully.