MH3510 - Group Project

Jun Kai Lim (N2401763K), Yantong Wu (N2401066J), Jonas Dickhöfer (N2400938A)

November 11, 2024 - due on November 11, 2024

1 Suggested Model

The complete model Ω is trivially

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \beta_3 \cdot x_3 + \beta_4 \cdot x_4 + \epsilon. \tag{1}$$

To find a possible reduced model ω, we plot predictor variable against the response variable for all x_i respectively, hoping, that through the plots we find a not sensible predictor variable to give us a trivial reduced model.

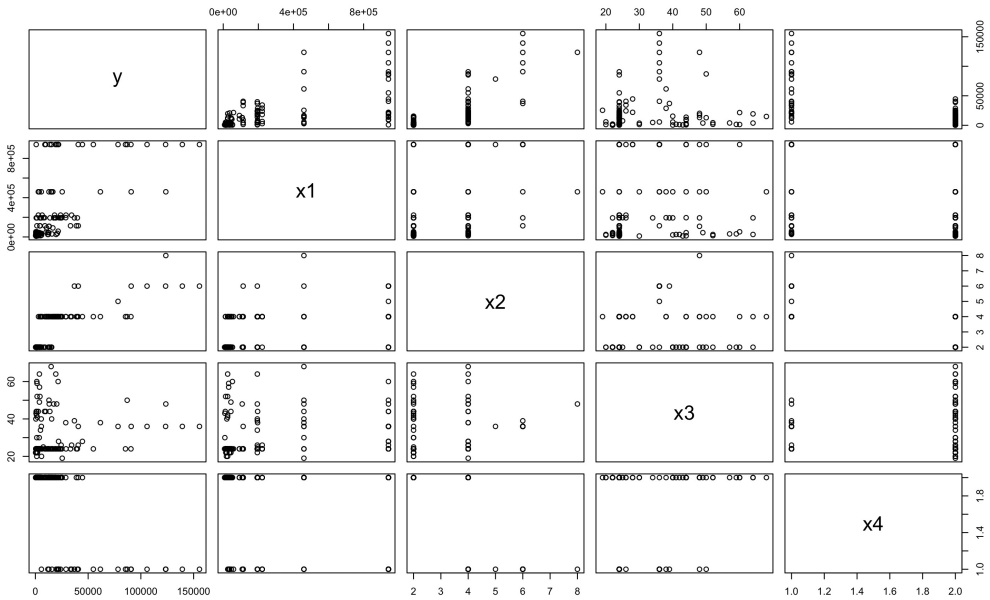

Figure 1: The plot matrix of the data

As seen in Figure 1, we cannot deduce any direct reduced model, for all plots seem quite random and seem to have outliers.

2 Model Fitting

We consider the following R-Code[1].

```
#read the data and create a data.frame
aadt_raw = read.table('aadt.txt',header=FALSE)
aadt = data.frame(y=aadt_raw$V1,x1=aadt_raw$V2,x2=aadt_raw$V3,x3=aadt_raw$V4,x4=as.factor(aadt_raw$V5))
#full model with all predictor variables
Omega = lm(y ~ x1 + x2 + x3 + x4, data = aadt)
```

The model summary for Ω is given in Figure 2.

[1]Note that the `aadt.txt` has to be replaced with an actual file path.

```
> summary(Omega)

Call:
lm(formula = y ~ x1 + x2 + x3 + x4, data = aadt)

Residuals:
   Min     1Q Median     3Q    Max
-36263  -8501   3493   6018  68317

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.425e+03  7.812e+03  -0.310    0.757
x1           3.303e-02  4.708e-03   7.017 1.63e-10 ***
x2           9.158e+03  1.531e+03   5.983 2.49e-08 ***
x3           1.003e+02  1.243e+02   0.807    0.421
x42         -2.361e+04  4.520e+03  -5.223 7.83e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15290 on 116 degrees of freedom
Multiple R-squared:  0.7527,  Adjusted R-squared:  0.7442
F-statistic: 88.29 on 4 and 116 DF,  p-value: < 2.2e-16
```

Figure 2: Summary of Ω model

We note that Ω is a relevant model with its F-statistic of 88.29 on 4 and 116 degrees of freedom, and a p-value of less than $2.2 \cdot 10^{-16}$. In terms of R-statistics, the model is also suitable, for the adjusted R^2 of 74.42% is acceptable.

Despite that, the summary of Ω shows that x_3 is not a sensible predictor variable, i.e. the p-value is large with 0.421. In other words, we fail to reject the null hypothesis for $\beta_3 = 0$.

So we try and fit a MLR without the x_3, i.e. we propose ω to be

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \beta_4 \cdot x_4 + \epsilon. \tag{2}$$

This results in the following R-Code.

```
#reduced model based of the results of Omega-model
omega = lm(y ~ x1 + x2 + x4, data = aadt)
```

The model summary for ω is given by the following figure 3.

```
> summary(omega)

Call:
lm(formula = y ~ x1 + x2 + x4, data = aadt)

Residuals:
   Min     1Q Median     3Q    Max
-35593  -7883   4010   5770  68441

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.487e+02  7.365e+03  -0.047    0.962
x1           3.356e-02  4.655e-03   7.211 5.93e-11 ***
x2           9.310e+03  1.517e+03   6.138 1.18e-08 ***
x42         -2.305e+04  4.460e+03  -5.168 9.85e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15270 on 117 degrees of freedom
Multiple R-squared:  0.7514,  Adjusted R-squared:  0.745
F-statistic: 117.8 on 3 and 117 DF,  p-value: < 2.2e-16
```

Figure 3: Summary of the ω model

We note that ω is a relevant model with its F-statistic of 117.8 on 3 and 117 degrees of freedom, and a p-value of less than $2.2 \cdot 10^{-16}$. In terms of R-statistics, the model is also suitable, for the adjusted R^2 of 74.5% is acceptable.

Note that the R^2 is minimally better for ω than Ω.
We can verify this using R-code.

```
#F-test
anova(omega,Omega)
```

This results in the following table 4, showing us that we indeed cannot reject the null-hypothesis for $\beta_3 = 0$.

```
> anova(omega,Omega)
Analysis of Variance Table

Model 1: y ~ x1 + x2 + as.factor(x4)
Model 2: y ~ x1 + x2 + x3 + as.factor(x4)
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    117 2.7281e+10
2    116 2.7128e+10  1 152302593 0.6512 0.4213
```

Figure 4: ANOVA table of ω vs Ω

3 Model Checking

In section 2, we deduced that the ω model in (2) is the better one. Now we plot the residuals of the fitted model against the excluded variable and also against the theoretical quantiles with the goal of verifying the model assumptions are adequate.

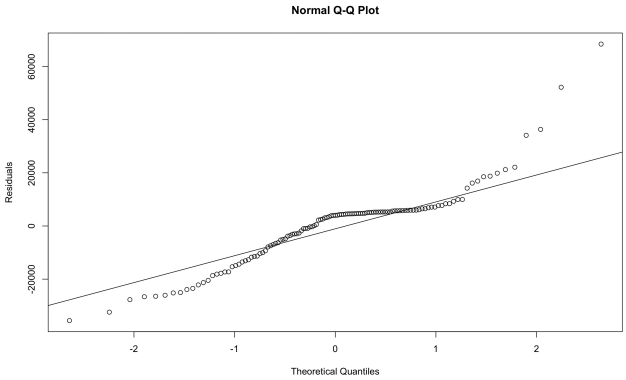

Figure 5: QQ-Plot for ω model

We can see that this is off, but one could say it is within a reasonable margin considering the amount of simplifying assumptions we make when considering the annual average daily traffic and its influencing variables.

Further let us look at the plot of residuals against the excluded information of x_3, which as shown in Figure 6 has satisfactory behaviour.

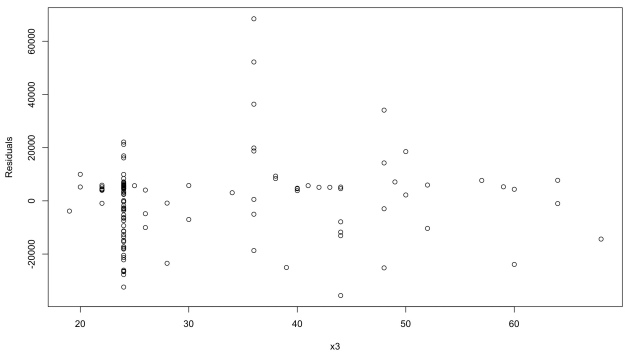

Figure 6: x_3 vs residual plot

Looking at the x_i included in the model plotted against the residuals, we get the plots shown in Figure 7.

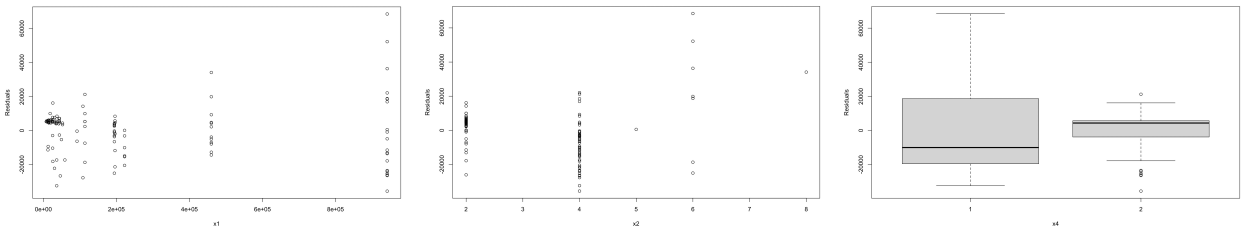

Figure 7: Included variables x_i , $i = 1, 2, 4$ vs residual plot

Lastly, let us plot the fitted values against the residuals as shown in Figure 8.

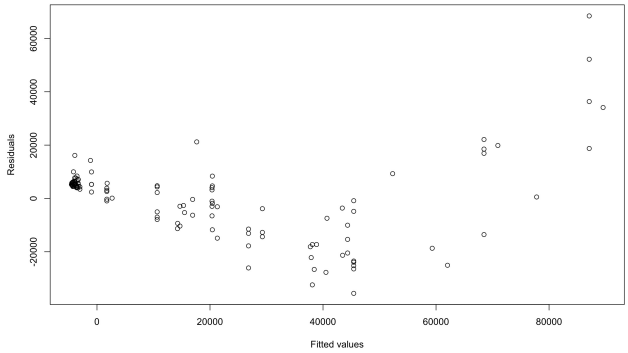

Figure 8: fitted value vs residual plot

Note that Figure 8 seems to imply our model is not correct, for it shows an "almost quadratic form". We can observe the described slope of about -1 for the residuals of fitted values (up to ~ 5000). Beyond that (i.e. for larger fitted values) our model seems to produce unusually large residuals which might imply outliers. Similarly, Figure 7 shows non-constant variance in the predictor variable versus the residuals, also potentially implying outliers. But to verify, one would need domain specific knowledge we do not have.

3.1 Improving the model

Realising, that there are no higher order relationships in the plots of Figure 7, we will focus on data set related improvement of the model [2]. Hence, we try and deal with the outliers in the previous models in several ways, with the goal being to find an improved significant model with higher R^2-value. Note that this means we try and "alter" the data. It does not mean that we design a new model formula.

Method 1: remove outliers and refit using Z-score

We identify and remove outliers based on the Z-score and then fit a MLR model on the altered data set with the following R-code[3].

[2]One can try higher order models, but when going up to cubic, any combination of higher orders on the predictor variables perform maximally as good as the ω model. Hence we fully exclude this in this report.

[3]Not that there are only two outliers by Z-score, so the reduced data set is only minimally smaller.

```r
#standardise the resiudals
std_res = rstandard(omega)
#identify the indices of the outlier points
out_index = which(abs(std_res) > 3)
#remove the outliers from teh data
aadt_no = aadt [-out_index]
#fit the model with the omega formula to the reduced data set
mlr_no = lm(y~x1 + x2 + x4, data = aadt_no)
summary(mlr_no)
```

This results in the following model summary in figure 9, showing that the model is significant by F-statistics. But the improvement is minimal to non-existent (R^2 is almost exactly the same as before with 74.72%).

```
> summary(mlr_no)

Call:
lm(formula = y ~ x1 + x2 + x4, data = aadt_no)

Residuals:
   Min     1Q Median     3Q    Max
-35593  -7883   4010   5770  68441

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.487e+02  7.365e+03  -0.047    0.962
x1           3.356e-02  4.655e-03   7.211 5.93e-11 ***
x2           9.310e+03  1.517e+03   6.138 1.18e-08 ***
x42         -2.305e+04  4.460e+03  -5.168 9.85e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15270 on 117 degrees of freedom
Multiple R-squared:  0.7514,  Adjusted R-squared:  0.745
F-statistic: 117.8 on 3 and 117 DF,  p-value: < 2.2e-16
```

Figure 9: model summary for the ω model on the Z-score outlier excluding data set

Method 2: remove outliers and refit using interquartile range (IQR)

Here, we try and improve the model by removing outliers using interquartile range (IQR). This is done using the following R-code[4].

```r
# outliers (IQR)
res = residuals(omega)
Q1 = quantile(res, 0.25)
Q3 = quantile(res, 0.75)
IQR_val = Q3-Q1
# compute the lower/ upper bound
L = Q1-1.5*IQR_val
U= Q3+1.5*IQR_val
# find out the outliers
IQR_out = res[res < L | res > U]
IQR_out
IQR_out_index = which(res < L | res > U)
aadt_no_IQR = aadt[-IQR_out_index,]
# then we fit a new mlr model using the reduced dataset
mlr_no_IQR = lm(y~x1+x2+x4,data=aadt_no_IQR)
summary(mlr_no_IQR)
```

The model summary is shown in the following figure 10.

[4]We find a total of 6 outliers in this method.

```
> summary(mlr_no_IQR)

Call:
lm(formula = y ~ x1 + x2 + x4, data = aadt_no_IQR)

Residuals:
   Min     1Q Median     3Q    Max
-24985  -5606   1416   3022  33669

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  8.300e+03  5.346e+03   1.553    0.123
x1           2.886e-02  3.394e-03   8.505 9.53e-14 ***
x2           6.117e+03  1.140e+03   5.367 4.44e-07 ***
x42         -2.197e+04  3.189e+03  -6.888 3.58e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10600 on 111 degrees of freedom
Multiple R-squared:  0.7653,  Adjusted R-squared:  0.7589
F-statistic: 120.6 on 3 and 111 DF,  p-value: < 2.2e-16
```

Figure 10: Model summary using the ω model and the IQR adapted data set

This model on this data set is significant (see F-statistics), and shows a small improvement in terms of R-statistics, i.e. the adjusted R^2 is slightly higher with 75.89%.

Method 3: Robust multiple linear regression

Trivially, this method is by design the best at handling outliers in the data. We use the following R-code to fit the robust MLR.

```
#install package if not installed
install.packages("robustbase")
#now fit robust MLR
library(robustbase)
MLR_rob = lmrob(y~x1+x2+x4, data = aadt)
summary(MLR_rob)
```

The summary 11 shows us, that indeed the robust approach to MLR gives us the best model. The model is significant with a good improvement in the R-statistics, R^2 of 79.45%.

```
Call:
lmrob(formula = y ~ x1 + x2 + x4, data = aadt)
 \--> method = "MM"
Residuals:
     Min       1Q   Median       3Q      Max
-16637.1  -1526.4   -276.4   4807.5 112933.1

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.404e+02  5.448e+03   0.099    0.921
x1           1.058e-02  2.272e-03   4.656 8.57e-06 ***
x2           5.352e+03  9.303e+02   5.753 7.14e-08 ***
x42         -9.472e+03  4.143e+03  -2.286    0.024 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Robust residual standard error: 3892
Multiple R-squared:  0.7997,  Adjusted R-squared:  0.7945
Convergence in 29 IRWLS iterations

Robustness weights:
 15 observations c(58,59,60,62,63,64,65,66,67,68,69,72,80,91,102) are outliers with |weight| <= 5.6e-05 ( < 0.00083);
 9 weights are ~= 1. The remaining 97 ones are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.02801 0.79530 0.97630 0.85130 0.99200 0.99900
Algorithmic parameters:
       tuning.chi              bb       tuning.psi        refine.tol           rel.tol         scale.tol         solve.tol          zero.tol
        1.548e+00         5.000e-01         4.685e+00         1.000e-07         1.000e-07         1.000e-10         1.000e-07         1.000e-10
      eps.outlier             eps.x warn.limit.reject warn.limit.meanrw
        8.264e-04         1.712e-06         5.000e-01         5.000e-01
     nResample         max.it       best.r.s       k.fast.s          k.max    maxit.scale      trace.lev            mts     compute.rd
           500             50              2              1            200            200              0           1000              0
fast.s.large.n
          2000
                  psi           subsampling                 cov compute.outlier.stats
            "bisquare"         "nonsingular"         ".vcov.avar1"                 "SM"
seed : int(0)
```

Figure 11: Model summary for the robust ω model

3.2 Model Checking for Robust MLR

Now that we have identified that the robust approach, unsurprisingly does work best, let us repeat the model checking process for the `MLR_rob` model.

As seen in Figure 12, the normality holds with the exception of ~16 data points. Note that the robust model summary 11 states that it identifies 16 outliers. Hence the normality assumption holds for this model.

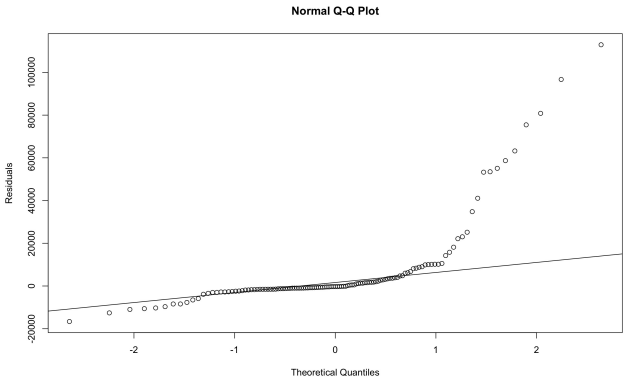

Figure 12: qqline of `MLR_rob`

Again, we can see that the exclusion of x_3 is validated via the plot of x_3 vs the residuals, as seen in figure 13

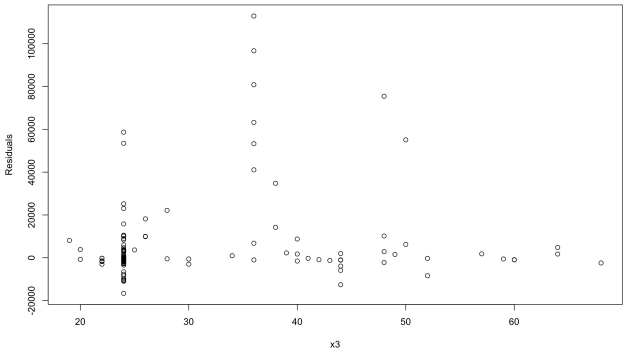

Figure 13: x_3 vs robust residual plot

And in terms of predictor variables vs residuals, all non-constant variance disappears when removing 16 data points which make the plot seem to have non-constant variance. This can be seen in Figure 14 below.

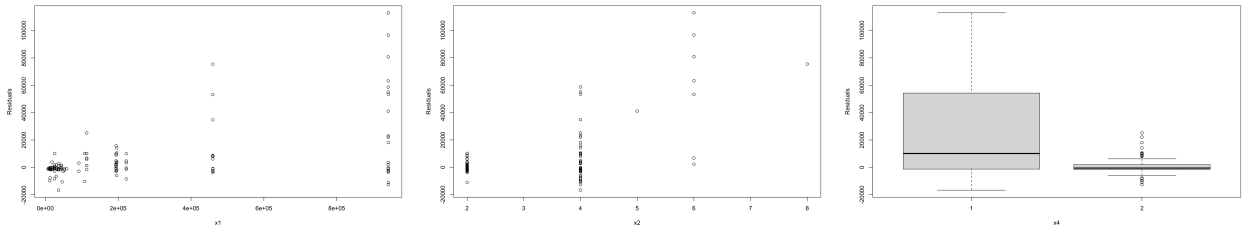

Figure 14: Included variables x_i , $i = 1, 2, 4$ vs robust model residual plot

Finally, the plot of residuals of the robust model against the fitted values has the desired form (see Figure 15). Especially when we remove the 16 data points that make it not seem the case.

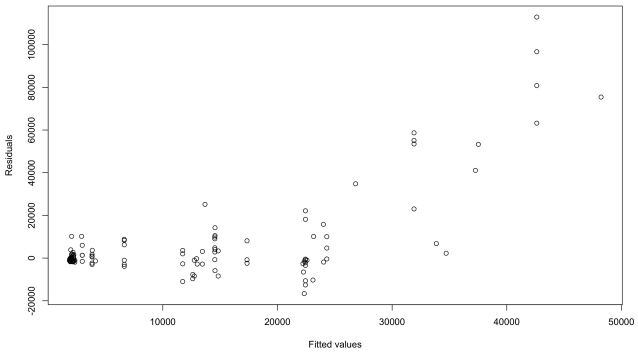

Figure 15: Fitted values of the robust model vs residuals of the robust model

3.3 Improving the Model - Checking for interaction of variables

Additionally, given the discussion in the last lecture/tutorial on checking for interactions, we use our best model and check for interactions of predictor variables in the residual plots. More explicitly, we will check whether any product of x_i 's can create a linear behaviour when plotted against the residuals of the robust MLR model. This results in the following plots.

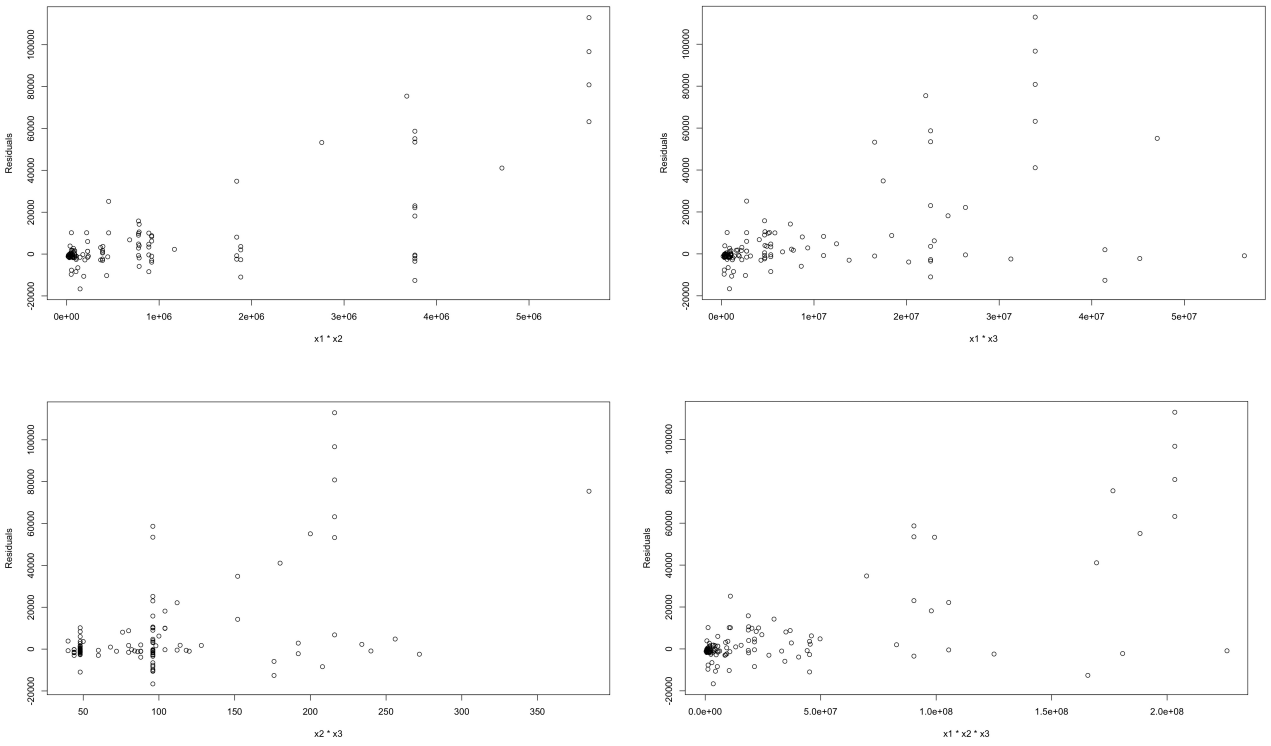

Figure 16: Interaction of x_i 's vs residuals of the robust MLR model

From the plots in Figure 16 we deduce that $x_1 \cdot x_2$ has a linear influence on the residual, while $x_1 \cdot x_3$, and $x_2 \cdot x_3$ do not influence the residual. Note that $x_1 \cdot x_2 \cdot x_3$ also shows a linear trend, which is due to $x_1 \cdot x_2$ showing the linear trend.

Based on this empiric result, we propose the model to actually be

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \underbrace{\beta_{12} \cdot x_1 \cdot x_2}_{\text{interaction between } x_1 \text{ and } x_2} + \beta_4 \cdot x_4 + \epsilon. \tag{3}$$

This translates to the following R-Code for the model, where we immediately also do the robust approach, as the data problems do not vanish by changing the formula.

```
mlr_i = lm(y ~ x1 + x2 + x1*x2 + x4, data = aadt)
summary(mlr_i)
mlr_i_rob = lmrob(y ~ x1 + x2 + x1*x2 + x4, data = aadt)
summary(mlr_i_rob)
```

The model summaries are given in the following Figure 17.

```
Call:
lm(formula = y ~ x1 + x2 + x1 * x2 + x4, data = aadt)

Residuals:
   Min     1Q Median     3Q    Max
-38658  -2219   -835   4121  36693

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.079e+04  5.818e+03   3.573 0.000516 ***
x1          -5.363e-02  9.369e-03  -5.725 8.27e-08 ***
x2           1.387e+03  1.369e+03   1.013 0.313164
x42         -2.083e+04  3.290e+03  -6.332 4.73e-09 ***
x1:x2        2.483e-02  2.483e-03  10.000  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 11240 on 116 degrees of freedom
Multiple R-squared:  0.8665,  Adjusted R-squared:  0.8619
F-statistic: 188.2 on 4 and 116 DF,  p-value: < 2.2e-16
```

```
Call:
lmrob(formula = y ~ x1 + x2 + x1 * x2 + x4, data = aadt)
 \--> method = "MM"
Residuals:
      Min        1Q    Median        3Q       Max
-70424.8  -3303.6   -953.3   1769.3  24160.1

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.211e+03  4.370e+03   1.421   0.1580
x1          -5.739e-02  5.556e-03 -10.330   <2e-16 ***
x2           1.980e+03  8.540e+02   2.318   0.0222 *
x42         -8.429e+03  3.680e+03  -2.291   0.0238 *
x1:x2        3.415e-02  1.821e-03  18.756   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Robust residual standard error: 3903
Multiple R-squared:  0.9765,  Adjusted R-squared:  0.9757
Convergence in 30 IRWLS iterations

Robustness weights:
 14 observations c(59,62,63,72,80,81,82,85,91,95,96,102,118,119) are outliers with |weight| = 0 ( < 0.00083);
 9 weights are ~= 1. The remaining 98 ones are summarized as
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
0.004787 0.872300 0.964900 0.858600 0.988600 0.998800
Algorithmic parameters:
        tuning.chi              bb         tuning.psi        refine.tol           rel.tol         scale.tol         solve.tol           zero.tol
         1.548e+00       5.000e-01        4.685e+00        1.000e-07        1.000e-07        1.000e-10        1.000e-07        1.000e-10
      eps.outlier             eps.x warn.limit.reject warn.limit.meanrw
         8.264e-04        1.027e-05        5.000e-01        5.000e-01
       nResample         max.it       best.r.s       k.fast.s          k.max    maxit.scale      trace.lev               mts     compute.rd
             500             50              2              1            200            200              0              1000              0
fast.s.large.n
            2000
              psi          subsampling                 cov compute.outlier.stats
        "bisquare"        "nonsingular"        ".vcov.avar1"               "SM"
seed : int(0)
```

Figure 17: Normal (left) and robust (right) model summary with interaction between x_1 and x_2

We once again do model checking on these models, already noting that the R^2 of the robust version is very high (almost 1) with 97.57%[5].

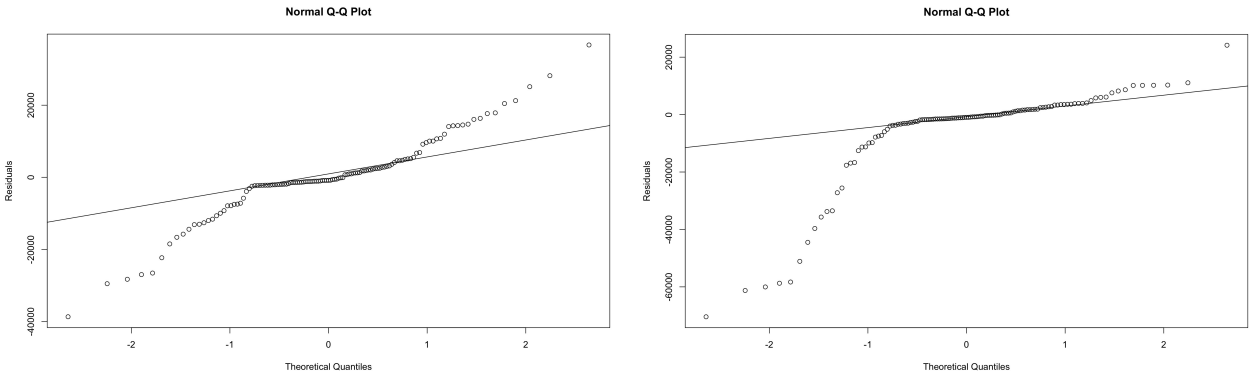

Figure 18: QQplot of Normal (left) and robust (right) model

As seen above (Figure 18), the normality assumption seems reasonable, with deviations when there are not many data points. I.e. in the robust model, the normality assumption holds better, as we know there exist outliers (removing the 14 outliers in the graph would give a very good normality plot).

Like in previous model checking we once again plot the models against the excluded variable x_3, the included variables, and the residuals. The results are shown below in Figures 19,20,21,22,23. We will not go through all the plots one by one, for the observations are about the same for all plots, i.e. the plots behave close to what is desired but have slight deviations. However we note that both models are better than any that is previously discussed, with the robust version having an advantage over the normal model[6].

[5]This might be over-fitting, but as we technically not discuss this issue in the course, we do not further investigate beyond the simple remark that it might be over-fitted

[6]Note: We did not remove the outliers in the plots for the robust model, but if we would, the plots for the robust model are close to ideal.

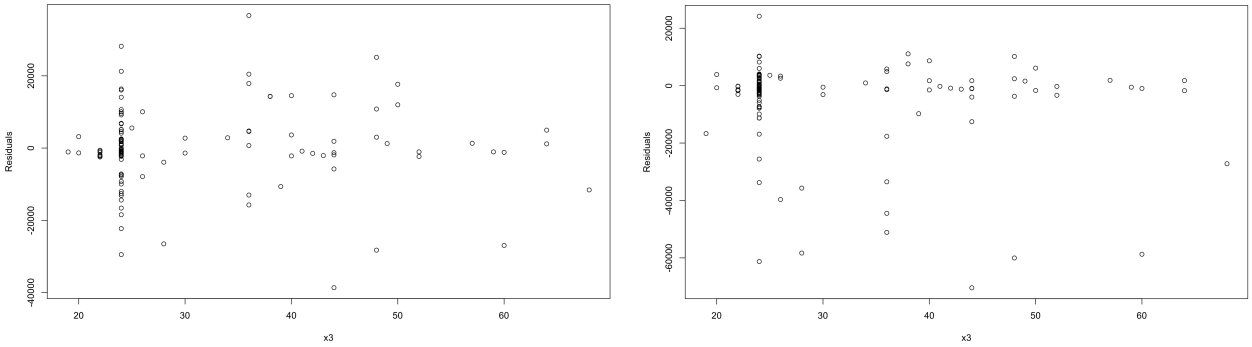

Figure 19: Residuals of Normal (left) and robust (right) model vs excluded variable x_3

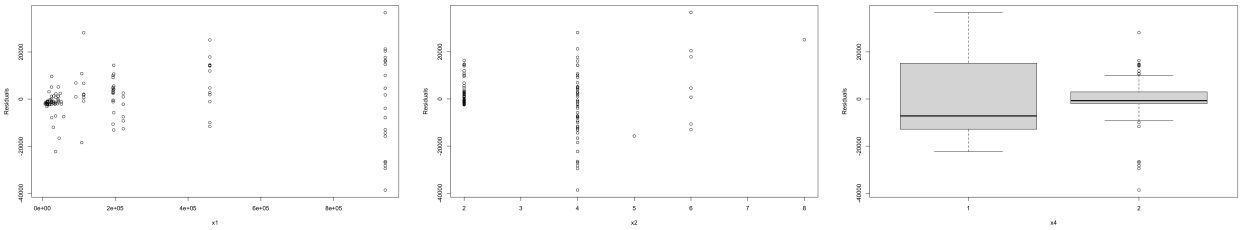

Figure 20: Included variables x_i , $i = 1, 2, 4$ vs Normal model residual plot

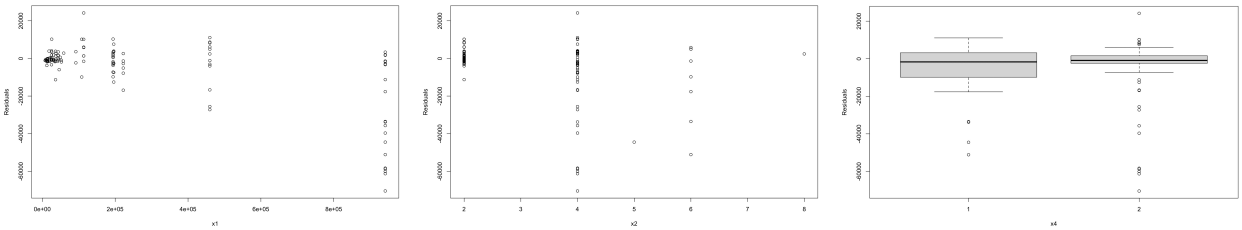

Figure 21: Included variables x_i , $i = 1, 2, 4$ vs robust model residual plot

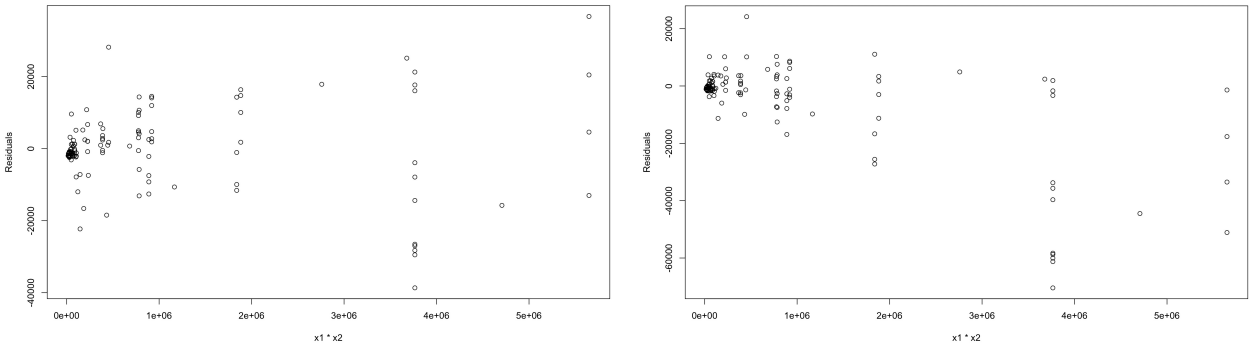

Figure 22: Residuals of Normal (left) and robust (right) model vs $x_1 \cdot x_2$

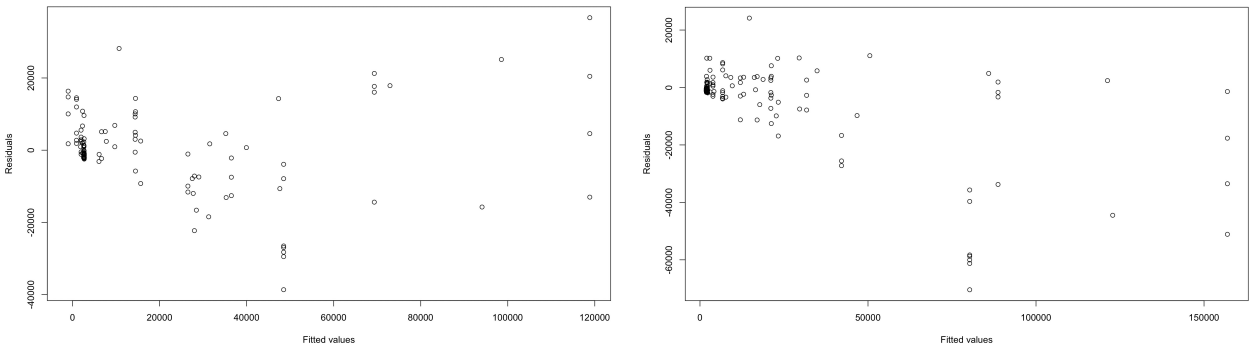

Figure 23: Residuals of Normal (left) and robust (right) model vs fitted values

With all the above things taken into consideration, we conclude that the robust model based on the formula in (3) is the best model available to us.

4 Prediction

Since we are working with R here, we can skip the theoretical considerations and employ the functionalities of R to predict directly, i.e. we predict using the following code snippet.

```
new_data = data.frame(x1 = 50000, x2 = 3, x3 = 60 x4 = factor(2, levels=c(1, 2)))
prediction_ci = predict(mlr_i_rob, newdata=new_data, interval="confidence", level=0.95)
prediction_ci
prediction_pi = predict(mlr_i_rob, newdata=new_data, interval="prediction", level=0.95)
prediction_pi
```

This results in the following output

```
      fit       lwr       upr
1 5974.385 4598.211 7350.559
```

```
      fit       lwr       upr
1 5974.385 -1877.958 13826.73
```

This directly gives us the confidence interval $x_0 \in [4598.211, 7350.559]$ with 95% confidence, and the prediction interval $x_0 \in [-1877.958, 13826.73]$ with 95% confidence.

Appendix

This is the R-Code as a continuous piece instead of cut apart as in the above sections[7].

```
#read the data and create a data.frame
aadt_raw = read.table('aadt.txt',header=FALSE)
aadt = data.frame(y=aadt_raw$V1,x1=aadt_raw$V2,x2=aadt_raw$V3,x3=aadt_raw$V4,x4=as.factor(aadt_raw$V5))
#full model with all predictor variables
Omega = lm(y ~ x1 + x2 + x3 + x4, data = aadt)
#reduced model based on the heuristic argument
omega = lm(y ~ x1 + x2 + x4, data = aadt)
#model summaries
summary(Omega)
summary(omega)
#F-test
anova(omega,Omega)
#plots for model checking
#normality
qqnorm(residuals(omega), ylab='Residuals')
qqline(residuals(omega))
#residual vs excluded variable
plot(aadt$x3,residuals(omega),ylab='Residuals', xlab='x3')
#residual vs used vaiables
plot(aadt$x1,residuals(omega),ylab='Residuals', xlab='x1')
plot(aadt$x2,residuals(omega),ylab='Residuals', xlab='x2')
plot(as.factor(aadt$x4),residuals(omega),ylab='Residuals', xlab='x4')
#residual vs fitted values
plot(fitted(omega),residuals(omega),ylab='Residuals', xlab='Fitted values')
#Improving the Model
#Method 1: using Z-score
#standardise the resiudals
std_res = rstandard(omega)
#identify the indices of the outlier points
out_index = which(abs(std_res) > 3)
#remove the outliers from the data
aadt_no = aadt [-out_index]
#fit the model with the omega formula to the reduced data set
mlr_no = lm(y~x1 + x2 + x4, data = aadt_no)
summary(mlr_no)
#Method 2: using interquartile range
# outliers (IQR)
res = residuals(omega)
Q1 = quantile(res, 0.25)
Q3 = quantile(res, 0.75)
IQR_val = Q3-Q1
# compute the lower/ upper bound
L = Q1-1.5*IQR_val
U= Q3+1.5*IQR_val
# find out the outliers
IQR_out = res[res < L | res > U]
IQR_out
IQR_out_index = which(res < L | res > U)
aadt_no_IQR = aadt[-IQR_out_index,]
# then we fit a new mlr model using the reduced dataset
mlr_no_IQR = lm(y~x1+x2+x4,data=aadt_no_IQR)
summary(mlr_no_IQR)
#Method 3: using robust MLR
#install package if not installed
install.packages("robustbase")
#now fit robust MLR
library(robustbase)
MLR_rob = lmrob(y~x1+x2+x4, data = aadt)
summary(MLR_rob)
#model checking the robust model
#normality
qqnorm(residuals(MLR_rob),ylab='Residuals')
qqline(residuals(MLR_rob))
#residual vs excluded
plot(aadt$x3,residuals(MLR_rob),ylab='Residuals', xlab='x3')
#residuals vs used variables
plot(aadt$x1,residuals(MLR_rob),ylab='Residuals', xlab='x1')
plot(aadt$x2,residuals(MLR_rob),ylab='Residuals', xlab='x2')
plot(aadt$x4,residuals(MLR_rob),ylab='Residuals', xlab='x4')
```

[7] Again note that `aadt.txt` has to be replaced with the actual path to the file.

```
#residuals vs fitted values
plot(fitted(MLR_rob),residuals(MLR_rob),ylab='Residuals', xlab='Fitted values')
#interaction vs robust residuals
plot(aadt$x1*aadt$x2,residuals(MLR_rob),ylab='Residuals', xlab='x1 * x2')
plot(aadt$x1*aadt$x3,residuals(MLR_rob),ylab='Residuals', xlab='x1 * x3')
plot(aadt$x2*aadt$x3,residuals(MLR_rob),ylab='Residuals', xlab='x2 * x3')
plot(aadt$x1*aadt$x2*aadt$x3,residuals(MLR_rob),ylab='Residuals', xlab='x1 * x2 * x3')
#models considering interaction x1 with x2
mlr_i = lm(y ~ x1 + x2 + x1*x2 + x4, data = aadt)
summary(mlr_i)
mlr_i_rob = lmrob(y ~ x1 + x2 + x1*x2 + x4, data = aadt)
summary(mlr_i_rob)
#model checking for the interaction models
#plots for model checking
#normality
qqnorm(residuals(mlr_i), ylab='Residuals')
qqline(residuals(mlr_i))
#residual vs excluded variable
plot(aadt$x3,residuals(mlr_i),ylab='Residuals', xlab='x3')
#residual vs used vaiables
plot(aadt$x1,residuals(mlr_i),ylab='Residuals', xlab='x1')
plot(aadt$x2,residuals(mlr_i),ylab='Residuals', xlab='x2')
plot(as.factor(aadt$x4),residuals(mlr_i),ylab='Residuals', xlab='x4')
#residual vs fitted values
plot(fitted(mlr_i),residuals(mlr_i),ylab='Residuals', xlab='Fitted values')
#plots for model checking
#normality
qqnorm(residuals(mlr_i_rob), ylab='Residuals')
qqline(residuals(mlr_i_rob))
#residual vs excluded variable
plot(aadt$x3,residuals(mlr_i_rob),ylab='Residuals', xlab='x3')
#residual vs used vaiables
plot(aadt$x1,residuals(mlr_i_rob),ylab='Residuals', xlab='x1')
plot(aadt$x2,residuals(mlr_i_rob),ylab='Residuals', xlab='x2')
plot(as.factor(aadt$x4),residuals(mlr_i_rob),ylab='Residuals', xlab='x4')
#residual vs fitted values
plot(fitted(mlr_i_rob),residuals(mlr_i_rob),ylab='Residuals', xlab='Fitted values')
#prediction for best model, i.e. mlr_i_rob
new_data = data.frame(x1 = 50000, x2 = 3, x3=60, x4 = factor(2, levels=c(1, 2)))
prediction_ci = predict(mlr_i_rob, newdata=new_data, interval="confidence", level=0.95)
prediction_ci
prediction_pi = predict(mlr_i_rob, newdata=new_data, interval="prediction", level=0.95)
prediction_pi
```