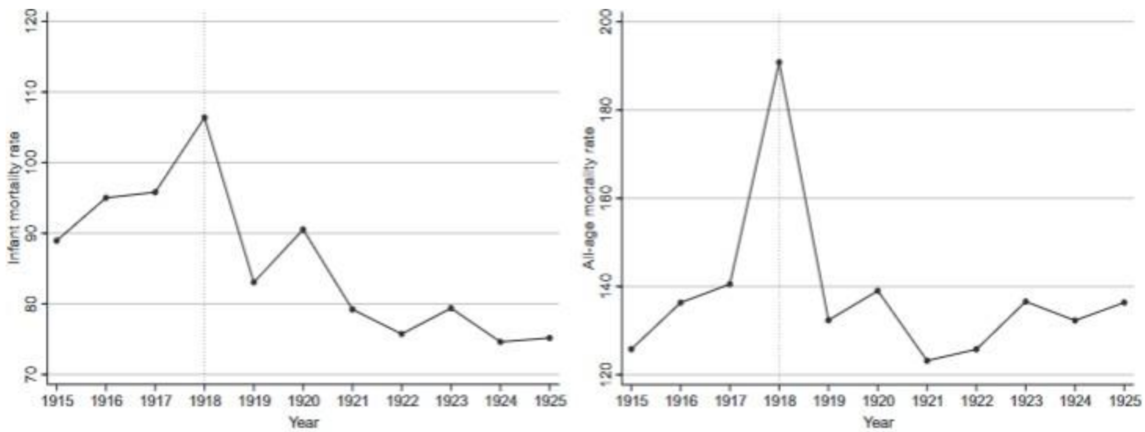
Kenneth Lam

1918 Spanish Influenza Pandemic

Introduction

In the last century and a half there have been multiple epidemics with the 1918 Spanish Flu Epidemic being responsible for the most deaths of them all. This flu epidemic lasted two years between 1918 and 1919, spreading and infecting a large number of people in the United States. The virus was highly contagious, and the lack of effective treatments or vaccines at the time meant the disease could run rampant. It is estimated the Spanish Flu claimed the lives of 675,000 people in the United States. It also had social and economic consequences such as causing disrupted trade, labor shortages, and the closure of public spaces. In our paper, we aim to investigate the factors that contributed to the extremely high mortality rates and find out why this flu epidemic was so deadly. We are especially interested in testing the hypothesis that air pollution, primarily from coal burning, was a key contributor to the high mortality rates. This theory is based on the paper *Pollution, Infectious Disease, and Mortality: Evidence from the 1918 Spanish Influenza Pandemic* which linked more air pollution as the primary reason for higher mortality rates (Clay). Throughout the rest of the paper we will research potential confounding variables for the relationship between air pollution and mortality rates. For example, poor water quality, population density, delayed onset, poverty (looking at average wage per capita), could all have contributed to higher mortality rates during this pandemic. In addition, the timing and effectiveness of public health interventions could have also had an impact on the spread of the virus. We acknowledge that age may also be an important confounding variable but we do not believe we have data on the age of residents in each city. To address the potential confounding variables we will build alternative models and compare them with our original air pollution theory. The following graph displays the infant and all age mortality



(a) Infant mortality (b) All-age mortality

rate between 1915 to 1925.

Figure 1: Infant and All age mortality 1915-1925

As shown in figure 1 the mortality rate of infants and all age groups experienced a significant decline in mortality rate as people started to develop immunities to the disease. For this reason, we choose to focus exclusively on the year 1918 which marks the peak of the Spanish Flu outbreak. Another key point is that the years prior to the pandemic, spanning the years 1915 to 1917, were used to establish a baseline to be able to compare the mortality rate changes during the pandemic. The response variable we consider in our paper is the excess deaths in 1918 which was calculated as the difference between the observed mortality rate in 1918 and the predicted mortality rate in 1918 based on a linear city specific trend from the period 1915 to 1925. By the end of our research, we hope to gain a better understanding of the complex interplay between the environmental factors, socioeconomic conditions, and public health interventions contributing to the severity of the 1918 Spanish Flu. The paper will consist of five main sections. The data sections will be an overview of the data set we will be working with including the data source, and important variables. This section will also include the exploratory analysis we performed in order to determine which variables had the most significant impact on infant mortality rates during this time. Then in the method sections we will explain the methodology behind our analysis including assumptions and computational aspects. In the simulations section our methods will be put to the test on generated data instead of real data. In the analysis section we will give a detailed evaluation of the results of our analysis section. Finally, the discussion section summarizes the results and provides context to the problem as well as give implications to future research that can be conducted.

Data

The *Pollution, Infectious Disease, and Mortality: Evidence from the 1918 Spanish Influenza Pandemic* paper referenced in the introduction provides a CLS Influenza R data file. This data set consists of 185 variables and 1,771 data entries containing a lot of irrelevant or missing data (Clay). In order to make the data set easier to work with it was subsetted into a copy of the data containing only 9 relevant variables with very little missing values. The data set merges data on city coal fired capacity with a panel dataset on mortality. The data on infant and all age deaths was collected from the period of 1915-1925 from the Mortality Statistics, containing a panel of 180 registration cities. The initial data set contained 283 cities with populations of at least 20,000 in

1921. It is worth noting that infant mortality is a common metric in studies for air pollution because infants are at a greater risk of environmental exposure and are representative of lifetime exposure. Data on city level pollution was obtained from a 1915 federal report on the location and capacity of coal fired and hydroelectric power stations. Total coal fired capacity was then calculated for each city centroid within a 30 mile radius since most power plants are local to the city. Other than the amount of coal capacity for each city, to measure air pollution we also looked at TSP (total suspended particles) in the air. This preliminary graph we made during our exploratory analysis shows the relationship between the capacity of coal fired power stations and excess infant mortality rates.

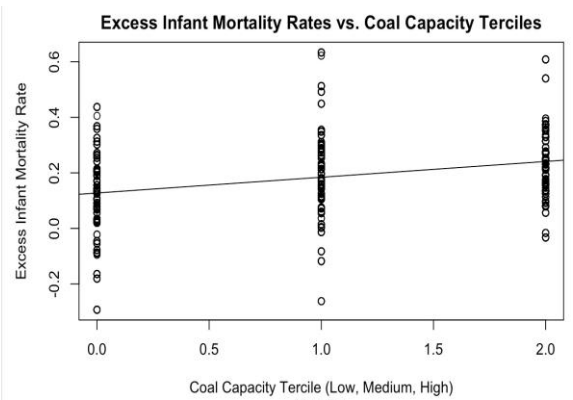


Figure 2

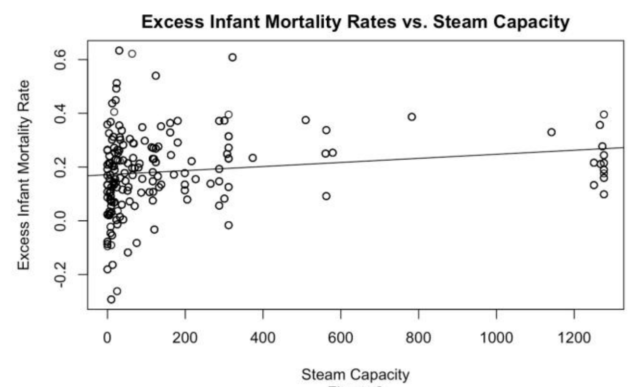


Figure 3

In Figure 2, the values 0, 1, and 2 on the x-axis represent low, medium, and high and high coal capacity. As you can see, there seems to be a positive correlation between coal fired power stations with higher capacities and higher excess infant mortality rates. This is reinforced by Figure 3 which displays total coal fired capacity for each city centroid within a 30 mile radius since most power plants are local to the city. These figures provide evidence that higher air pollution (from cities that burn more coal) could have caused higher infant mortality rates at this time during the epidemic.

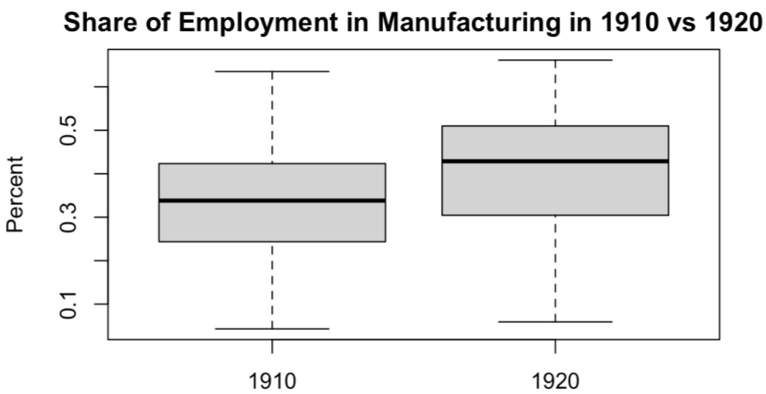


Figure 4

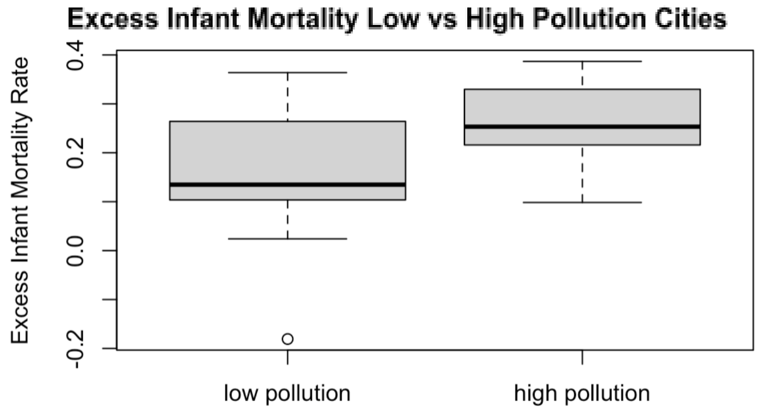


Figure 5

To exemplify the relationship between pollution and infant mortality during this time even further, we looked into factors that caused increased pollution during this time. Figure 4 shows that in between 1910 and 1920, the share of the amount of people who were involved in manufacturing significantly increased. As manufacturing increased, so did the pollution in these cities. By classifying cities as high pollution cities (cities where total coal fired capacity was then calculated for each city centroid within a 30 mile was greater than 500) and low pollution cities (cities where this capacity was 0), we are able to further show the relationship between excess infant mortality deaths in 1918 and pollution. Figure 5 shows that excess infant mortality rates were much higher on average in cities where pollution was prevalent versus cities where pollution was mostly absent.

Conducting exploratory data analysis, we looked into other factors that could have affected infant mortality rate at the time of this epidemic.

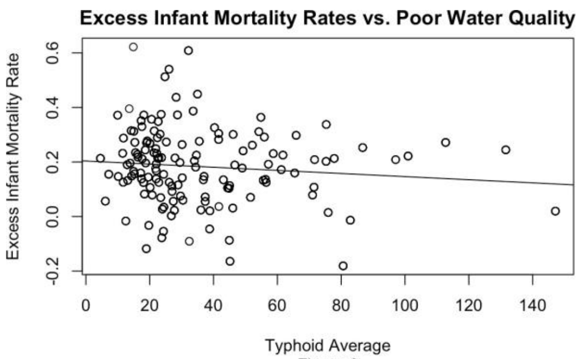


Figure 6

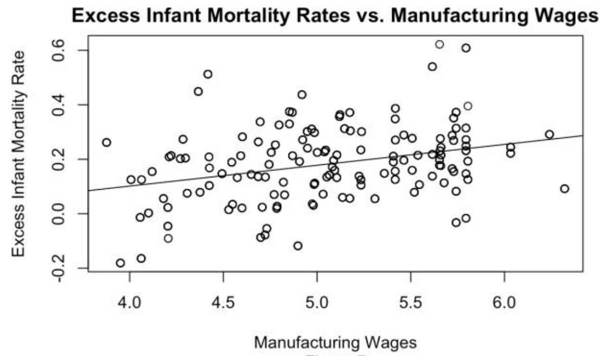


Figure 7

Figures 6 and 7 show that confounding variables such as typhoidave (indicator for poor quality drinking water) and mwage_bls901900 (manufacturing wages made in 1900) also seemed to have relationships with excess infant mortality rate. This gives us evidence that air pollution is not the singular factor that caused the 1918 Spanish flu epidemic to be so deadly. In order to look into this more, we are going to try to take these confounding variables into account and see if air pollution still has the same effect on excess infant mortality rates.

Methods

The first method this paper utilizes is linear regression. Linear regression is a commonly used concept that quantitatively measures the independent variables' effect on the dependent variable. This perfectly aligns with the purpose of this paper since the goal is to determine what variables affected excess infant mortality rate during the 1918 Spanish Flu Epidemic.

Another useful tool of linear regression models is their ability to predict. But before explaining predictions with linear models it is important to understand the assumptions that these models make. The first assumption made by linear regression is that the relationship between the independent variables and the dependent variable is linear. The paper touches on this in the previous section. The next assumption is that observations in our data set are independent of each other. Which means that each city’s excess infant mortality rate recorded in a given year does not affect other excess infant mortality rates in any city or year. This is a very complex assumption to assess but there is no clear evidence that observations are dependent. Another assumption is homoscedasticity where there is no pattern to the residuals.

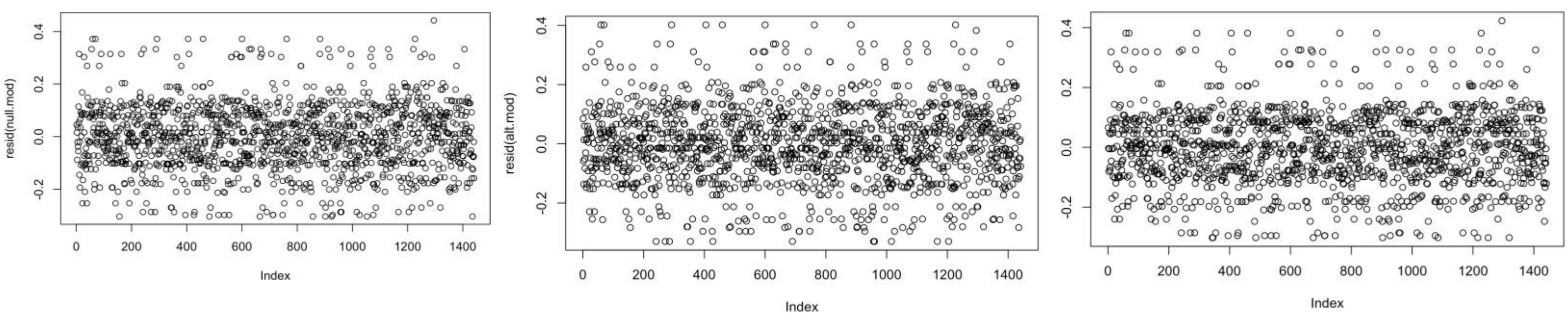


Figure 8

By plotting the residuals in a residual plot as shown by Figure 8 above we can see that none of the 3 models have a pattern in their residuals. Normally distributed residuals is another assumption that linear models make. In order for us to investigate this, we plotted QQ plots for the residuals for all of our 3 linear models. QQ plots with straight lines indicate normally distributed residuals. Figure 9 below shows that all 3 of our linear models have somewhat long tails. This does not invalidate the normality assumption but it is something to be aware of.

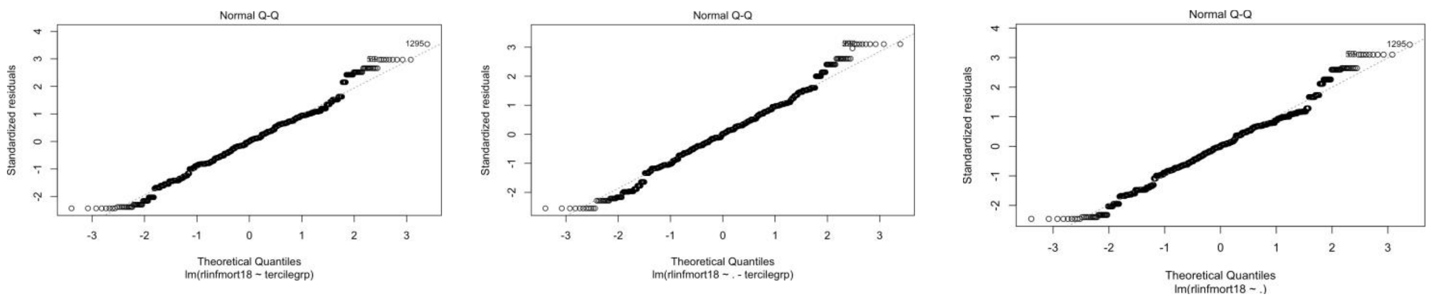


Figure 9

The final assumption is that there are no highly correlated independent variables. Therefore, we created a table of correlation matrices to check if any of the independent variables were highly correlated.

	hydromw_30mile	pop1910	typhoidave	mwage_bls901900	tercilegrp	swhite1910
hydromw_30mile	1.00000000	-0.04531340	0.22354135	-0.08546167	-0.2616483	0.1133711
pop1910	-0.04531340	1.00000000	-0.06884575	0.96474386	0.5341894	0.1132307
typhoidave	0.22354135	-0.06884575	1.00000000	-0.06994646	-0.3104709	-0.3169752
mwage_bls901900	-0.08546167	0.96474386	-0.06994646	1.00000000	0.5645782	0.1199694
tercilegrp	-0.26164825	0.53418943	-0.31047088	0.56457822	1.0000000	0.1378507
swhite1910	0.11337112	0.11323067	-0.31697523	0.11996938	0.1378507	1.0000000

Figure 10

Figure 10 shows that population size and manufacturing wages were highly correlated so we removed the latter from the model. After ensuring that our models pass all assumptions, it can take the next step to prediction (Zach).

One of the methods that we used in order to test how well our models performed was Cross-Validation. Cross-Validation is a method used in statistical analysis in order to try and prevent overfitting while also predicting how well our models will perform. In Cross-Validation, the data is split into an independent training set and validation set.

The training set is for our model to be fit on, and the validation set is used to make predictions. Usually, the training set is larger than the validation set, so the model is trained well. For example, we put 70% of our data into the training set when we performed cross validation. Each of our three models are then trained using the training set and then predictions are made using the validation set. We are then able to effectively calculate the mean squared error (MSE), which is the sum of the squared difference between our predictions and the actual observed test set. Using this mean squared error, we are able to determine how effective our models are at prediction (Irizarry).

There are many assumptions that we need to make sure in order to carry out Cross-Validation. The first assumption is that all of the rows of the data used are random and independent. This makes sure that it is representative of the entire population. Another assumption is that there are no extreme outliers in the data. A third assumption is that when data is split into a training set and testing set, it is split randomly and independently (Martin). We effectively did this, by randomly choosing 70% of our data to be in the validation set.

Another method we used to further our analysis of the performance of our three models was to use bootstrapping. Bootstrapping allows us to use a Monte Carlo simulation without knowing the entire distribution of our data. In order to bootstrap we sample from our original dataset, with replacement, many times. We then will compute the statistic that we are trying to estimate for all of these samples. Through the law of large numbers, we believe that the distribution of the statistics we estimated for all of those samples should mirror the distribution of the true statistic (Irizarry).

There are many assumptions about the underlying data necessary to be made in order to accurately perform bootstrap analysis. The first assumption is that the original dataset is representative of the population. A second assumption is that each time you resample from the population, you resample in the exact same random way. A third assumption is that you calculate the statistic you are trying to estimate the same way every time you get a new bootstrap sample. Other assumptions for bootstrapping relate to what you are trying to calculate, and in our case, overlap with the assumptions for linear regression presented above ("Bootstrapping").

Simulations

Models Used in the Paper

Models	Components
Null	**Tercile Group (Low, Medium, High Coal Capacity)**
Alternative	**Typhoid (Water Quality), Hydro Capacity within 30 Miles, 1910 Population, Proportion of White Population, Manufacturing Wages**
Full	**Null and Alternative Components**

Table 1

We made three different models, shown in Table 1, in order to investigate what factors significantly led to the excess infant mortality in 1918. Our null model is based on the fact that there is a lot of evidence that air pollution led to this excess infant mortality rate. Therefore, this model contains a variable that measures coal capacity in certain areas, therefore, being a great measurement for air pollution. However, we also wanted to investigate other factors that could possibly relate to air pollution that led to the appearance of air pollution causing this excess infant mortality rate. Therefore, our alternative model contains variables relating to water quality, pollution relating to hydropower, proportion of white people in the population, and level of manufacturing wages in these areas. In addition, we create a full model that combines the null and alternative models.

Using the cross validation method described in the previous section, we used simulated data in order to see how well these models performed on data that they have never seen. The purpose of this was to evaluate which was the best model and therefore which variables were significantly impacting the excess infant mortality rate. If we do cross-validation once we can see each model's mean squared error. However, this does not tell us whether or not these models have a significant difference in performance. So, we simulate cross-validation 1000 times in order to form confidence intervals to tell us whether or not the mean squared errors are significantly different between the models.

A similar technique is utilized for resample Bootstrapping. If we resample rows at random with replacement, we will create 1000 different data sets. Computing mean squared error for each of these data sets will allow us to form confidence intervals to see if there is a significant difference in performance between models.

Analysis

Evaluating Models

	Smaller Model	Larger Model	P-value

	Anova		
	Null	Full	0.00001116
	Alternative	Full	0
	Simulated MSE CIs		
<u>Model</u>	<u>Lower</u>	<u>Center</u>	<u>Upper</u>
	Bootstrap		
Null	0.0143	0.0156	0.0169
Alternative	0.0159	0.0173	0.0187
Full	0.0140	0.0153	0.0165
	Cross-Validation		
Null	0.0137	0.0156	0.0175
Alternative	0.0153	0.0175	0.0196
Full	0.0136	0.0155	0.0173

Table 2

This Analysis section will investigate the results of the simulation section shown in Table 2 above. The first part of the table is an Anova Test that determines whether or not the model is significantly different when adding more predictors. We find that adding confounding variables to the null model makes a significant difference. On the other hand, adding the coal capacity terciles to the alternative model makes an even greater difference.

After the preliminary Anova testing, we go to the next step in Table 2 with the results of our simulating data set methods. As described in the Methods section, we use resample Bootstrapping and Cross-Validation in order to find confidence intervals for each models' mean squared error. We find similar results for both methods with the full model having the lowest centered MSE, followed by the null model, then the alternative model. However, there are intersections between the three models' CIs and therefore there is no statistically significant difference in the models' MSEs. This means there is no evidence that one of the three models outperform the others and are all equivalent.

These results of MSEs may seem to contradict the results of the Anova test, but this is not actually the case. What is likely happening is the full model with the most predictors is overfitting the data set and doesn't have the best generalization. This is exposed in Cross-Validation where the full model is tested on new data and fails to significantly outperform the smaller models.

Discussion

The 1918 Spanish Influenza Pandemic was an extremely deadly disease on its own; however, many factors in cities in the United States led to it to be even more fatal. When reading the paper *Pollution, Infectious Disease, and Mortality: Evidence from the 1918 Spanish Influenza Pandemic* the main takeaway was that cities with higher air pollution experienced increased excess mortality. The goal of this paper was to investigate this claim and see if there was a better model that took into account some confounding variables. While looking through this dataset, we were able to confirm that cities with higher air pollution (that were located near places that produced a lot of coal) had experienced excess mortality during this time. After performing more exploratory analysis, we also found that variables such as water contamination, manufacturing wages, and race proved to be significant in explaining the increased excess mortality rates. Therefore, we were able to create linear regression models that investigated actually how significant each of these predictor variables were in predicting the excess death. Combining linear regression with cross-validation and bootstrapping techniques, we were able to further investigate these models. After simulating data sets and forming confidence intervals on mean squared error in order to evaluate the performance of each model, there is no evidence that suggests that coal capacity alone is not the best way to understand the increase in excess infant mortality rates during the 1918 Spanish Influenza Epidemic. This provides evidence that other confounding variables in these cities or a combination of both air pollution and these variables led to these increased rates. Overall, our analysis shows that there are many significant factors that can influence fatality rates during times of pandemics.

There are many implications of this research for future research. The motivation for writing this paper came from the fact that a new pandemic, Covid-19, had recently occurred. Even though this virus was not as fatal as the 1918 Spanish Influenza Epidemic, its consequences were immense. Covid-19 fatality rates and affects differed immensely among different demographic variables, and some of the methods used in this paper could be interesting to use to predict how likely Covid-19 is to be fatal to individuals. Factors such as age, blood types, economic inequality, smoking, gender, race, and urbanization level all greatly affected the fatality of Covid-19 in individuals. Therefore, being able to create a model that takes in these factors and predicts fatality rates would be useful for high risk individuals (Li, Mengyuan).

Works Cited

“Bootstrap Confidence Intervals.” *Bootstrap Confidence Intervals- Principles*.

Clay, Karen, et al. “Pollution, Infectious Disease, and Mortality: Evidence from the 1918 Spanish Influenza Pandemic.” 2015, https://doi.org/10.3386/w21635.

Irizarry, Rafael A. *Introduction to Data Science: Data Analysis and Prediction: Algorithms with R*. CRC Press Taylor & Francis Group, 2020.

Li, Mengyuan, et al. “Identifying Novel Factors Associated with Covid-19 Transmission and Fatality Using the Machine Learning Approach.” 2020, https://doi.org/10.1101/2020.06.10.20127472.

Martin Liebig (Schmitz), PhD. “When Cross Validation Fails.” *Medium*, Towards Data Science, 1 Sept. 2017.

Zach. “The Four Assumptions of Linear Regression.” *Statology*, 21 Jan. 2021, https://www.statology.org/linear-regression-assumptions/.

Spanish Influenza Project

2023-03-10

```
library(boot)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages--------------------------------------- tidyverse 1.3.2 --

## v ggplot2 3.4.0     v purrr   1.0.1
## v tibble  3.2.1     v stringr 1.5.0
## v tidyr   1.3.0     v forcats 0.5.2
## v readr   2.1.3
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
if(file.exists("CLS_influenza.rda")){
  load("CLS_influenza.rda")}
data = influenza
head(data)
```

```
##   citycode     ctname year county state      stname stfips point_x point_y
## 1        2 BRIDGEPORT 1915     10     1 Connecticut      9 1879636 2246713
## 2        2 BRIDGEPORT 1916     10     1 Connecticut      9 1879636 2246713
## 3        2 BRIDGEPORT 1917     10     1 Connecticut      9 1879636 2246713
## 4        2 BRIDGEPORT 1918     10     1 Connecticut      9 1879636 2246713
## 5        2 BRIDGEPORT 1919     10     1 Connecticut      9 1879636 2246713
## 6        2 BRIDGEPORT 1920     10     1 Connecticut      9 1879636 2246713
##   sept14 sept1421 sept2128 sept28oct5 oct5 hydromw_30mile hydromw_50mile
```

```
## 1              0              1              0              0      0            5.965599            5.965599
## 2              0              1              0              0      0            5.965599            5.965599
## 3              0              1              0              0      0            5.965599            5.965599
## 4              0              1              0              0      0            5.965599            5.965599
## 5              0              1              0              0      0            5.965599            5.965599
## 6              0              1              0              0      0            5.965599            5.965599
##   steammw_30mile steammw_50mile mort_tot infmort_tot mort_white infmort_white
## 1       117.7403       915.5472     1827         378         NA            NA
## 2       117.7403       915.5472     2354         486         NA            NA
## 3       117.7403       915.5472     2270         445         NA            NA
## 4       117.7403       915.5472     2981         492         NA            NA
## 5       117.7403       915.5472     1975         398         NA            NA
## 6       117.7403       915.5472     1872         384         NA            NA
##   mort_nonwhite infmort_nonwhite population pop1921 fips    statenam  nhgisnam
## 1            NA               NA         NA  143555 9001 Connecticut Fairfield
## 2            NA               NA         NA  143555 9001 Connecticut Fairfield
## 3            NA               NA         NA  143555 9001 Connecticut Fairfield
## 4            NA               NA         NA  143555 9001 Connecticut Fairfield
## 5            NA               NA         NA  143555 9001 Connecticut Fairfield
## 6            NA               NA     143555  143555 9001 Connecticut Fairfield
##   fipsstate pop1900 popurb1900 swhite1900 emp1900 smfg1900 mfg1900
## 1         9  184203     121644  0.9817376   70758 0.398598   28204
## 2         9  184203     121644  0.9817376   70758 0.398598   28204
## 3         9  184203     121644  0.9817376   70758 0.398598   28204
## 4         9  184203     121644  0.9817376   70758 0.398598   28204
## 5         9  184203     121644  0.9817376   70758 0.398598   28204
## 6         9  184203     121644  0.9817376   70758 0.398598   28204
##   mwage_bls901900 pop1910 popurb1910 swhite1910 emp1910  smfg1910 mfg1910
## 1        261772.2  245322     175082  0.9850197  110434 0.4233932   46757
## 2        261772.2  245322     175082  0.9850197  110434 0.4233932   46757
## 3        261772.2  245322     175082  0.9850197  110434 0.4233932   46757
## 4        261772.2  245322     175082  0.9850197  110434 0.4233932   46757
## 5        261772.2  245322     175082  0.9850197  110434 0.4233932   46757
## 6        261772.2  245322     175082  0.9850197  110434 0.4233932   46757
##   pop1920 popurb1920 swhite1920 emp1920  smfg1920 mfg1920 mwage_bls901920
## 1  320936     240751  0.9836136  137721 0.5328018   73378        529692.3
## 2  320936     240751  0.9836136  137721 0.5328018   73378        529692.3
## 3  320936     240751  0.9836136  137721 0.5328018   73378        529692.3
## 4  320936     240751  0.9836136  137721 0.5328018   73378        529692.3
## 5  320936     240751  0.9836136  137721 0.5328018   73378        529692.3
## 6  320936     240751  0.9836136  137721 0.5328018   73378        529692.3
##   pop1930 popurb1930 swhite1930 emp1930  smfg1930 mfg1930 mwage_bls901930
## 1  386702     286648   0.978177  168754 0.3772474   63662        620147.2
## 2  386702     286648   0.978177  168754 0.3772474   63662        620147.2
## 3  386702     286648   0.978177  168754 0.3772474   63662        620147.2
## 4  386702     286648   0.978177  168754 0.3772474   63662        620147.2
## 5  386702     286648   0.978177  168754 0.3772474   63662        620147.2
## 6  386702     286648   0.978177  168754 0.3772474   63662        620147.2
##   state_fips distance       city typhoidave infmort_rate mort_rate linfmort
## 1          9 29.98202 BRIDGEPORT         19     91.81443  127.2683 4.530602
## 2          9 29.98202 BRIDGEPORT         19    118.04712  163.9790 4.779520
## 3          9 29.98202 BRIDGEPORT         19    108.08842  158.1275 4.692159
## 4          9 29.98202 BRIDGEPORT         19    119.50449  207.6556 4.791687
## 5          9 29.98202 BRIDGEPORT         19     96.67233  137.5779 4.581618
```

```
## 6            9  29.98202  BRIDGEPORT          19      93.27180    130.4030  4.546182
##        lmort  d18  ltyphoidave  typhoidXd18  ltyphoidXd18 linfmort15 linfmort16
## 1 7.149668    0     2.944439            0      0.000000   4.530602    4.77952
## 2 7.402933    0     2.944439            0      0.000000   4.530602    4.77952
## 3 7.366619    0     2.944439            0      0.000000   4.530602    4.77952
## 4 7.638947    1     2.944439           19      2.944439   4.530602    4.77952
## 5 7.227502    0     2.944439            0      0.000000   4.530602    4.77952
## 6 7.173981    0     2.944439            0      0.000000   4.530602    4.77952
##   linfmort17 linfmort15Xd18 linfmort16Xd18 linfmort17Xd18 linfmort1517Xd18
## 1   4.692159       0.000000        0.00000       0.000000         0.000000
## 2   4.692159       0.000000        0.00000       0.000000         0.000000
## 3   4.692159       0.000000        0.00000       0.000000         0.000000
## 4   4.692159       4.530602        4.77952       4.692159         4.667427
## 5   4.692159       0.000000        0.00000       0.000000         0.000000
## 6   4.692159       0.000000        0.00000       0.000000         0.000000
##   ldistance  lmfg1910 mwageper1900 pctmanuf1910 lmwage1900 tercile1 tercile2
## 1  3.400598 10.75272     5.598567    0.4233932   1.722511        0        1
## 2  3.400598 10.75272     5.598567    0.4233932   1.722511        0        1
## 3  3.400598 10.75272     5.598567    0.4233932   1.722511        0        1
## 4  3.400598 10.75272     5.598567    0.4233932   1.722511        0        1
## 5  3.400598 10.75272     5.598567    0.4233932   1.722511        0        1
## 6  3.400598 10.75272     5.598567    0.4233932   1.722511        0        1
##   pctforeign1910 pcturb1910 tercile_home infmort15 infmort16 infmort17
## 1      0.2952895  0.7136824            1  91.81443  118.0471  108.0884
## 2      0.2952895  0.7136824            1  91.81443  118.0471  108.0884
## 3      0.2952895  0.7136824            1  91.81443  118.0471  108.0884
## 4      0.2952895  0.7136824            1  91.81443  118.0471  108.0884
## 5      0.2952895  0.7136824            1  91.81443  118.0471  108.0884
## 6      0.2952895  0.7136824            1  91.81443  118.0471  108.0884
##   mortrate15 mortrate16 mortrate17   lmort15   lmort16 infmort1517 tercile1Xd15
## 1   1272.683    1639.79   1581.276  7.148882  7.402323    105.9833            0
## 2   1272.683    1639.79   1581.276  7.148882  7.402323    105.9833            0
## 3   1272.683    1639.79   1581.276  7.148882  7.402323    105.9833            0
## 4   1272.683    1639.79   1581.276  7.148882  7.402323    105.9833            0
## 5   1272.683    1639.79   1581.276  7.148882  7.402323    105.9833            0
## 6   1272.683    1639.79   1581.276  7.148882  7.402323    105.9833            0
##   tercile1Xd16 tercile1Xd17 tercile1Xd18 tercile1Xd19 tercile1Xd20 tercile1Xd21
## 1            0            0            0            0            0            0
## 2            0            0            0            0            0            0
## 3            0            0            0            0            0            0
## 4            0            0            0            0            0            0
## 5            0            0            0            0            0            0
## 6            0            0            0            0            0            0
##   tercile1Xd22 tercile1Xd23 tercile1Xd24 tercile1Xd25 tercile2Xd15 tercile2Xd16
## 1            0            0            0            0            1            0
## 2            0            0            0            0            0            1
## 3            0            0            0            0            0            0
## 4            0            0            0            0            0            0
## 5            0            0            0            0            0            0
## 6            0            0            0            0            0            0
##   tercile2Xd17 tercile2Xd18 tercile2Xd19 tercile2Xd20 tercile2Xd21 tercile2Xd22
## 1            0            0            0            0            0            0
## 2            0            0            0            0            0            0
## 3            1            0            0            0            0            0
```

```
## 4             0             1             0             0             0             0
## 5             0             0             1             0             0             0
## 6             0             0             0             1             0             0
##   tercile2Xd23 tercile2Xd24 tercile2Xd25 lpop1921 poptrend swhitetrend
## 1            0            0            0 11.87447 22739.62    1886.313
## 2            0            0            0 11.87447 22751.49    1887.298
## 3            0            0            0 11.87447 22763.37    1888.283
## 4            0            0            0 11.87447 22775.24    1889.268
## 5            0            0            0 11.87447 22787.12    1890.253
## 6            0            0            0 11.87447 22798.99    1891.238
##   pctforeigntrend pcturbtrend  popXd18 swhiteXd18 pctforeignXd18 pcturbXd18
## 1        565.4793    1366.702  0.00000  0.0000000      0.0000000  0.0000000
## 2        565.7746    1367.416  0.00000  0.0000000      0.0000000  0.0000000
## 3        566.0699    1368.129  0.00000  0.0000000      0.0000000  0.0000000
## 4        566.3652    1368.843 11.87447  0.9850197      0.2952895  0.7136824
## 5        566.6605    1369.557  0.00000  0.0000000      0.0000000  0.0000000
## 6        566.9557    1370.270  0.00000  0.0000000      0.0000000  0.0000000
##   mfgtrend mfgwagetrend homecoaltrend  mfgXd18 mfgwageXd18 homecoalXd18
## 1 20591.46     3298.608          1915  0.00000    0.000000            0
## 2 20602.21     3300.331          1916  0.00000    0.000000            0
## 3 20612.96     3302.053          1917  0.00000    0.000000            0
## 4 20623.71     3303.776          1918 10.75272    1.722511            1
## 5 20634.47     3305.498          1919  0.00000    0.000000            0
## 6 20645.22     3307.221          1920  0.00000    0.000000            0
##       xtrend     ytrend   xXd18   yXd18 inftrend   infXd18 morttrend mortXd18
## 1 3599502336 4302454784       0       0 8676.103 0.000000  13690.11 0.000000
## 2 3601382144 4304701440       0       0 8680.634 0.000000  13697.26 0.000000
## 3 3603261696 4306948096       0       0 8685.164 0.000000  13704.41 0.000000
## 4 3605141248 4309195264 1879636 2246713 8689.694 4.530602  13711.56 7.148882
## 5 3607021056 4311441920       0       0 8694.226 0.000000  13718.71 0.000000
## 6 3608900608 4313688576       0       0 8698.756 0.000000  13725.85 0.000000
##   ltyptrend ltypXd18 ldistXd18 ldisttrend tercilegrp popdensity popdensityXd18
## 1  5638.601 0.000000  0.000000   6512.145          2   8019.833          0.000
## 2  5641.545 0.000000  0.000000   6515.545          2   8019.833          0.000
## 3  5644.489 0.000000  0.000000   6518.946          2   8019.833          0.000
## 4  5647.434 2.944439  3.400598   6522.347          2   8019.833       8019.833
## 5  5650.378 0.000000  0.000000   6525.747          2   8019.833          0.000
## 6  5653.323 0.000000  0.000000   6529.148          2   8019.833          0.000
##   popdensitytrend lpopdensity lpopdensityXd18 lpopdensitytrend rlinfmort18
## 1        15357979    8.989673        0.000000         17215.22   0.2706304
## 2        15365999    8.989673        0.000000         17224.21   0.2706304
## 3        15374019    8.989673        0.000000         17233.20   0.2706304
## 4        15382039    8.989673        8.989673         17242.19   0.2706304
## 5        15390059    8.989673        0.000000         17251.18   0.2706304
## 6        15398078    8.989673        0.000000         17260.17   0.2706304
##    rlmort18 mr_diff1715 mr_diff2515 imr_diff1715 imr_diff2515 late nearww1
## 1 0.4263411    30.85925   -19.92268     16.27399     -51.9796    0       1
## 2 0.4263411    30.85925   -19.92268     16.27399     -51.9796    0       1
## 3 0.4263411    30.85925   -19.92268     16.27399     -51.9796    0       1
## 4 0.4263411    30.85925   -19.92268     16.27399     -51.9796    0       1
## 5 0.4263411    30.85925   -19.92268     16.27399     -51.9796    0       1
## 6 0.4263411    30.85925   -19.92268     16.27399     -51.9796    0       1
##   lateXd18 latetrend nearww1Xd18 nearww1trend sample_JEH markelsample
## 1        0         0           0         1915          1            0
```

```
## 2         0         0         0      1916         1         0
## 3         0         0         0      1917         1         0
## 4         0         0         1      1918         1         0
## 5         0         0         0      1919         1         0
## 6         0         0         0      1920         1         0
##   earlyresponse longintervention longintXd18 earlyrespXd18 linfmort_v1
## 1             0                0           0             0    4.582154
## 2             0                0           0             0    4.670003
## 3             0                0           0             0    4.478486
## 4             0                0           0             0    4.617135
## 5             0                0           0             0    4.494057
## 6             0                0           0             0    4.531631
##   linfmort_v2  frac1844 frac1844trend frac1844Xd18 balancedsample south
## 1    3.308035 0.4379565      838.6866    0.0000000              1     0
## 2    3.551186 0.4379565      839.1246    0.0000000              1     0
## 3    3.465691 0.4379565      839.5626    0.0000000              1     0
## 4    3.563106 0.4379565      840.0005    0.4379565              1     0
## 5    3.357753 0.4379565      840.4385    0.0000000              1     0
## 6    3.323212 0.4379565      840.8765    0.0000000              1     0
##   highwind tercile1Xd18hwind tercile1Xd18lwind tercile2Xd18hwind
## 1        1                 0                 0                 0
## 2        1                 0                 0                 0
## 3        1                 0                 0                 0
## 4        1                 0                 0                 1
## 5        1                 0                 0                 0
## 6        1                 0                 0                 0
##   tercile2Xd18lwind tercile1hydro tercile2hydro tercile1hydroXd18
## 1                 0             1             0                 0
## 2                 0             1             0                 0
## 3                 0             1             0                 0
## 4                 0             1             0                 1
## 5                 0             1             0                 0
## 6                 0             1             0                 0
##   tercile2hydroXd18
## 1                 0
## 2                 0
## 3                 0
## 4                 0
## 5                 0
## 6                 0
```

```
library(ggplot2)
attach(data)
```

```
## The following object is masked from package:tidyr:
##
##     population
##
## The following object is masked from package:boot:
##
##     city
```

Data

Expand: Our data set has 185 variables and 1,771 data entries. We could choose whether we want to group by city, county, or state. We have mortality total and infant mortality total. Population variables to standardize a city's mortality rate per capita.

Hydro, steam 30, 50 mile?

```
# Alternatives
# poor water quality = typhoidave
# city density = pop1921
# delayed onset = (point_x,point_y) data points - Citi Bike MC samplings
# public health effort = ?
# race = swhite1920
# income = mwage_bls901920
```

Possible confounding variables: poor water quality (typhoid mortality), city's density, race, wages, delayed onset, public health effort.

Researchers have claimed that the virus weakened over the course of the fall of 1918, so that locations that experienced a delayed onset were exposed to a less virulent strain. The ability of public officials to respond to the outbreak may also have been related to the timing of local onset. We assess whether factors related to the timing of onset were related to pandemic mortality.

Some researchers have argued that other local public interventions, such as quarantines and bans on public gatherings, influenced severity (Markel et al. 2007). To assess the role of the local public health effort, we use data from Markel et al. (2007) on local interventions for a subsample of 32 cities and construct indicators for early and long-term interventions following their classification.

EDA

```
data.copy = data.frame(pop1910, typhoidave, rlmort18, rlinfmort18, tercilegrp, swhite1910)

head(data.copy)
```

```
##   pop1910 typhoidave  rlmort18  rlinfmort18  tercilegrp  swhite1910
## 1  245322         19 0.4263411    0.2706304           2   0.9850197
## 2  245322         19 0.4263411    0.2706304           2   0.9850197
## 3  245322         19 0.4263411    0.2706304           2   0.9850197
## 4  245322         19 0.4263411    0.2706304           2   0.9850197
## 5  245322         19 0.4263411    0.2706304           2   0.9850197
## 6  245322         19 0.4263411    0.2706304           2   0.9850197
```

```
table(data.copy$tercilegrp)
```

```
##
##   0   1   2
## 577 588 606
```

```
nrow(data.copy)
```

```
## [1] 1771
```

```
data.copy = na.omit(data.copy)
nrow(data.copy)
```

```
## [1] 1441
```

```
null.mod = lm(rlinfmort18 ~ tercilegrp, data=data.copy)
summary(null.mod)
```

```
##
## Call:
## lm(formula = rlinfmort18 ~ tercilegrp, data = data.copy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29841 -0.08647  0.00147  0.08672  0.44242
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.117624   0.005422   21.69   <2e-16 ***
## tercilegrp  0.061911   0.004058   15.26   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1248 on 1439 degrees of freedom
## Multiple R-squared:  0.1393, Adjusted R-squared:  0.1387
## F-statistic: 232.8 on 1 and 1439 DF,  p-value: < 2.2e-16
```

```
alt.mod = lm(rlinfmort18 ~ .-tercilegrp-rlmort18, data=data.copy)
summary(alt.mod)
```

```
##
## Call:
## lm(formula = rlinfmort18 ~ . - tercilegrp - rlmort18, data = data.copy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33642 -0.07937  0.00513  0.08308  0.42411
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.900e-02  4.295e-02   0.442    0.658
## pop1910      5.952e-08  1.190e-08   5.001 6.41e-07 ***
## typhoidave  -3.374e-04  1.503e-04  -2.244    0.025 *
## swhite1910   1.667e-01  4.254e-02   3.918 9.35e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.1319 on 1437 degrees of freedom
## Multiple R-squared: 0.04059,	Adjusted R-squared: 0.03859
## F-statistic: 20.27 on 3 and 1437 DF,  p-value: 7.257e-13
```

```
full.mod = lm(rlinfmort18 ~ .-rlmort18, data=data.copy)
summary(full.mod)
```

```
##
## Call:
## lm(formula = rlinfmort18 ~ . - rlmort18, data = data.copy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.30271 -0.08246  0.00215  0.08430  0.44887
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.734e-02  4.070e-02  -1.409  0.15915
## pop1910     -4.011e-08  1.328e-08  -3.022  0.00256 **
## typhoidave   3.090e-04  1.486e-04   2.079  0.03778 *
## tercilegrp   7.003e-02  5.035e-03  13.910  < 2e-16 ***
## swhite1910   1.723e-01  3.995e-02   4.313 1.72e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1238 on 1436 degrees of freedom
## Multiple R-squared:  0.1545, Adjusted R-squared:  0.1522
## F-statistic: 65.61 on 4 and 1436 DF,  p-value: < 2.2e-16
```

```
anova(null.mod,  full.mod)
```

```
## Analysis of Variance Table
##
## Model 1: rlinfmort18 ~ tercilegrp
## Model 2: rlinfmort18 ~ (pop1910 + typhoidave + rlmort18 + tercilegrp +
##     swhite1910) - rlmort18
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1   1439 22.422
## 2   1436 22.024  3   0.39759 8.641 1.102e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(alt.mod, full.mod)
```

```
## Analysis of Variance Table
##
## Model 1: rlinfmort18 ~ (pop1910 + typhoidave + rlmort18 + tercilegrp +
##     swhite1910) - tercilegrp - rlmort18
## Model 2: rlinfmort18 ~ (pop1910 + typhoidave + rlmort18 + tercilegrp +
##     swhite1910) - rlmort18
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1   1437 24.992
```

```
## 2    1436  22.024  1      2.9676 193.49 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
null.mod.all = lm(rlmort18 ~ tercilegrp, data=data.copy)
summary(null.mod)
```

```
##
## Call:
## lm(formula = rlinfmort18 ~ tercilegrp, data = data.copy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29841 -0.08647  0.00147  0.08672  0.44242
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.117624   0.005422   21.69   <2e-16 ***
## tercilegrp  0.061911   0.004058   15.26   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1248 on 1439 degrees of freedom
## Multiple R-squared:  0.1393, Adjusted R-squared:  0.1387
## F-statistic: 232.8 on 1 and 1439 DF,  p-value: < 2.2e-16
```

```
alt.mod.all = lm(rlmort18 ~ .-tercilegrp-rlinfmort18, data=data.copy)
summary(alt.mod)
```

```
##
## Call:
## lm(formula = rlinfmort18 ~ . - tercilegrp - rlmort18, data = data.copy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33642 -0.07937  0.00513  0.08308  0.42411
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.900e-02  4.295e-02   0.442    0.658
## pop1910      5.952e-08  1.190e-08   5.001 6.41e-07 ***
## typhoidave  -3.374e-04  1.503e-04  -2.244    0.025 *
## swhite1910   1.667e-01  4.254e-02   3.918 9.35e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1319 on 1437 degrees of freedom
## Multiple R-squared:  0.04059,    Adjusted R-squared:  0.03859
## F-statistic: 20.27 on 3 and 1437 DF,  p-value: 7.257e-13
```

```
full.mod.all = lm(rlmort18 ~ .-rlinfmort18, data=data.copy)
summary(full.mod)
```

```
##
## Call:
## lm(formula = rlinfmort18 ~ . - rlmort18, data = data.copy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.30271 -0.08246  0.00215  0.08430  0.44887
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.734e-02  4.070e-02  -1.409  0.15915
## pop1910     -4.011e-08  1.328e-08  -3.022  0.00256 **
## typhoidave   3.090e-04  1.486e-04   2.079  0.03778 *
## tercilegrp   7.003e-02  5.035e-03  13.910  < 2e-16 ***
## swhite1910   1.723e-01  3.995e-02   4.313 1.72e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1238 on 1436 degrees of freedom
## Multiple R-squared:  0.1545, Adjusted R-squared:  0.1522
## F-statistic: 65.61 on 4 and 1436 DF,  p-value: < 2.2e-16
```

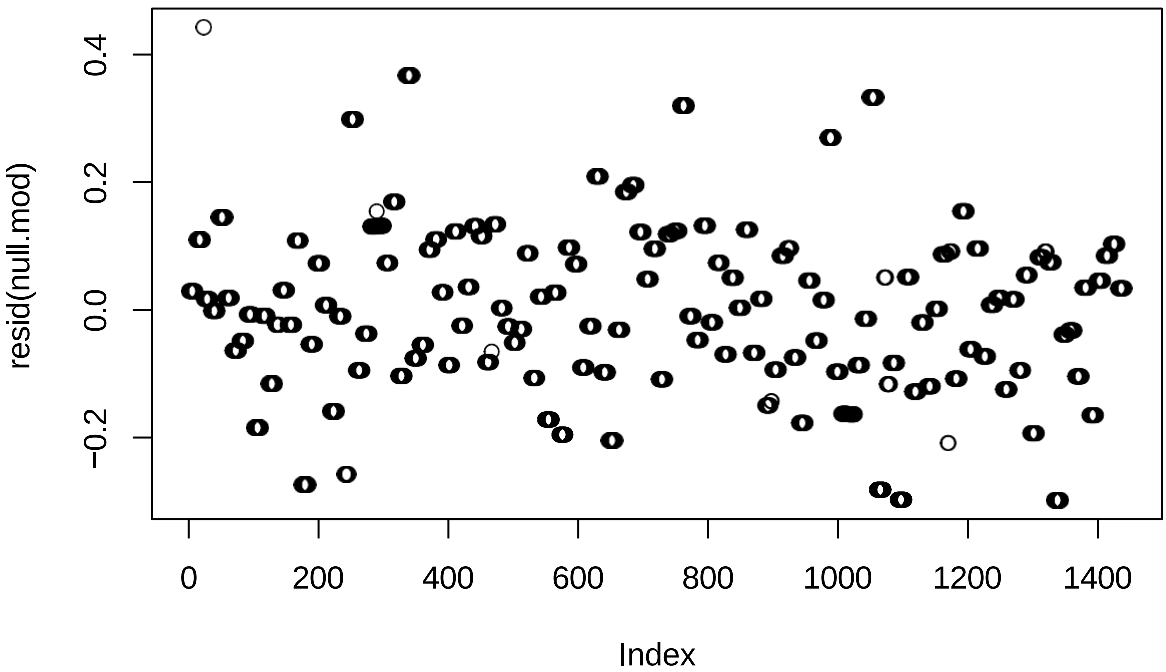
```
anova(null.mod.all, full.mod.all)
```

```
## Analysis of Variance Table
##
## Model 1: rlmort18 ~ tercilegrp
## Model 2: rlmort18 ~ (pop1910 + typhoidave + rlinfmort18 + tercilegrp +
##     swhite1910) - rlinfmort18
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   1439 16.952
## 2   1436 16.123  3   0.82875 24.604 1.59e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

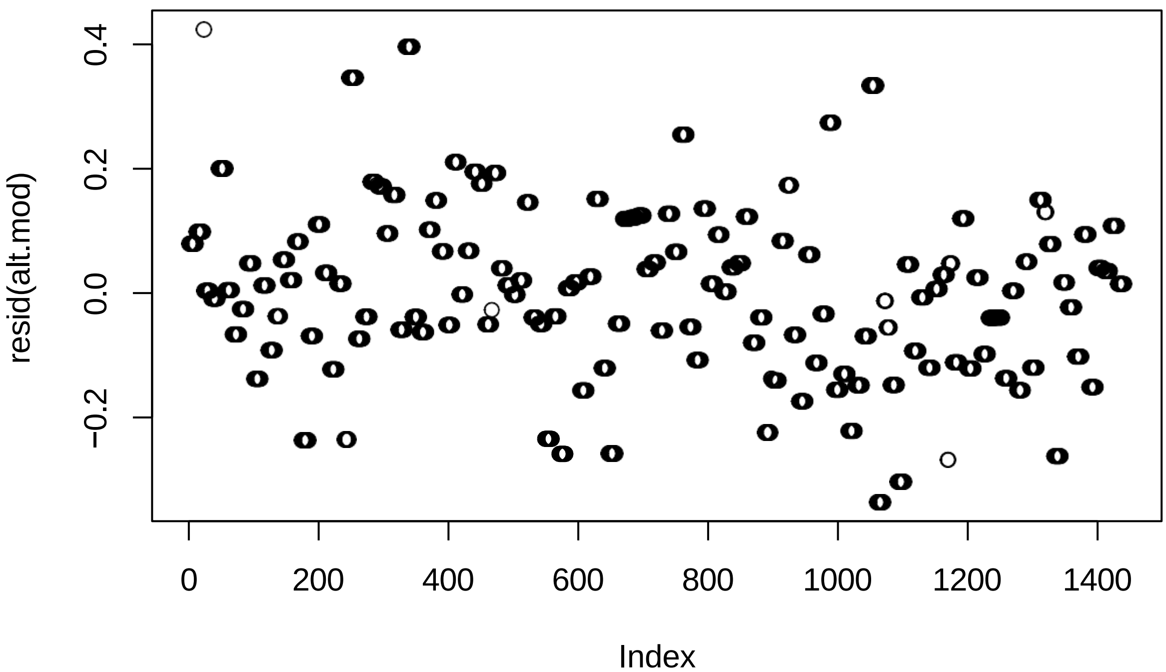
```
anova(alt.mod.all, full.mod.all)
```

```
## Analysis of Variance Table
##
## Model 1: rlmort18 ~ (pop1910 + typhoidave + rlinfmort18 + tercilegrp +
##     swhite1910) - tercilegrp - rlinfmort18
## Model 2: rlmort18 ~ (pop1910 + typhoidave + rlinfmort18 + tercilegrp +
##     swhite1910) - rlinfmort18
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   1437 18.924
## 2   1436 16.123  1    2.8008 249.45 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

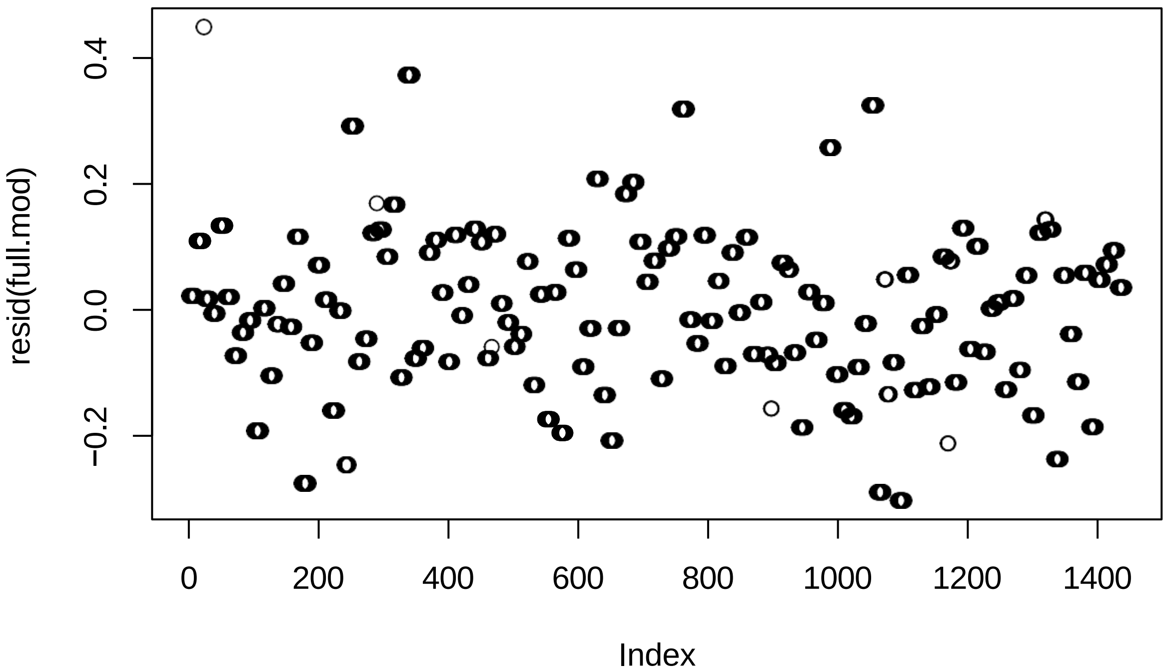
```
plot(resid(null.mod))
```



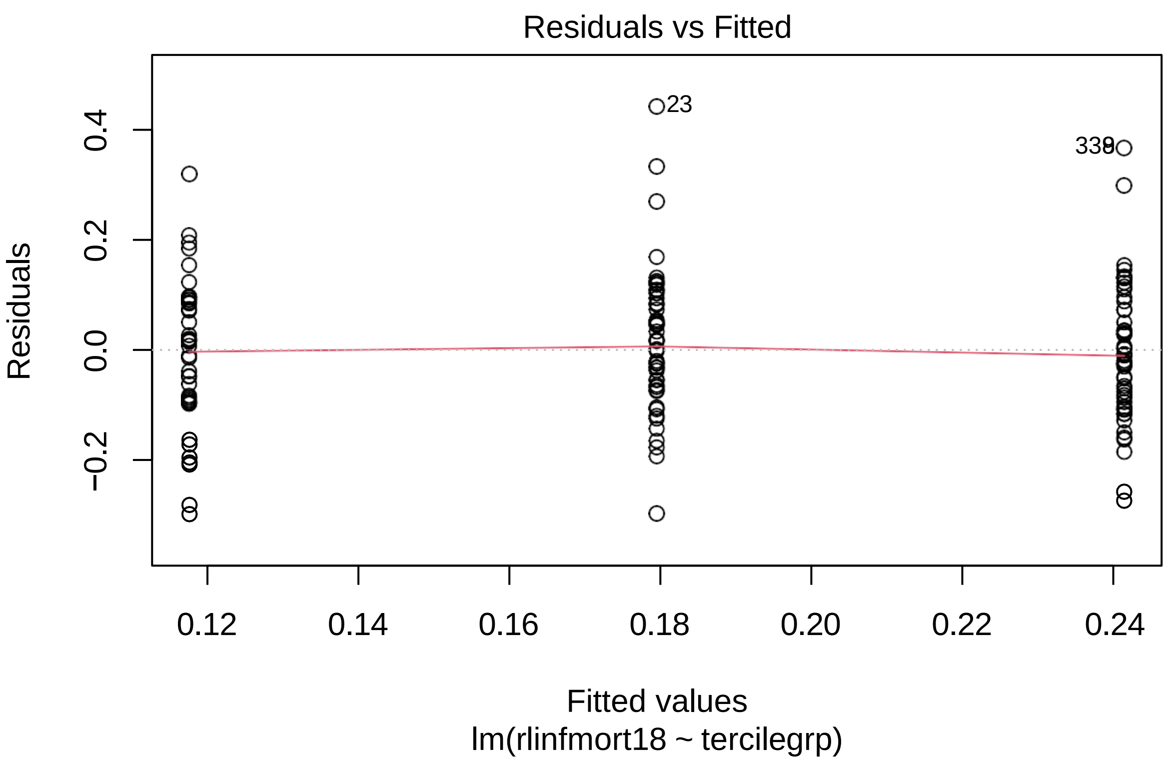

```
plot(resid(alt.mod))
```

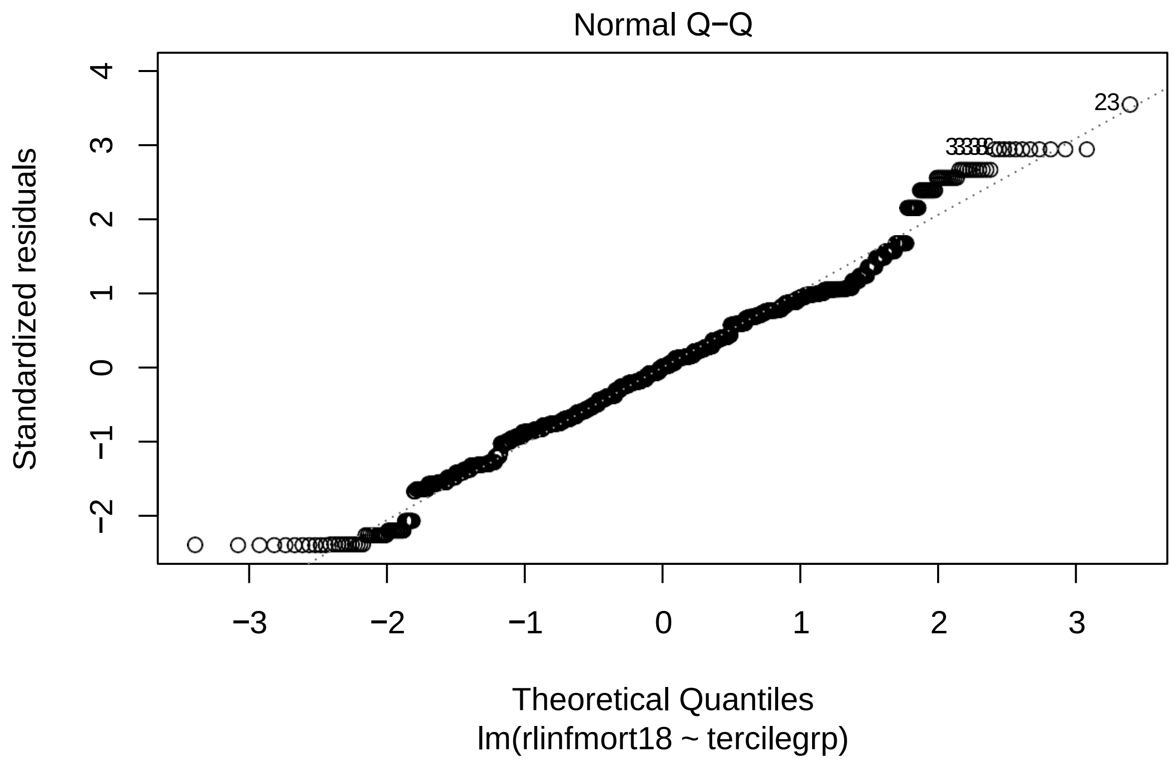


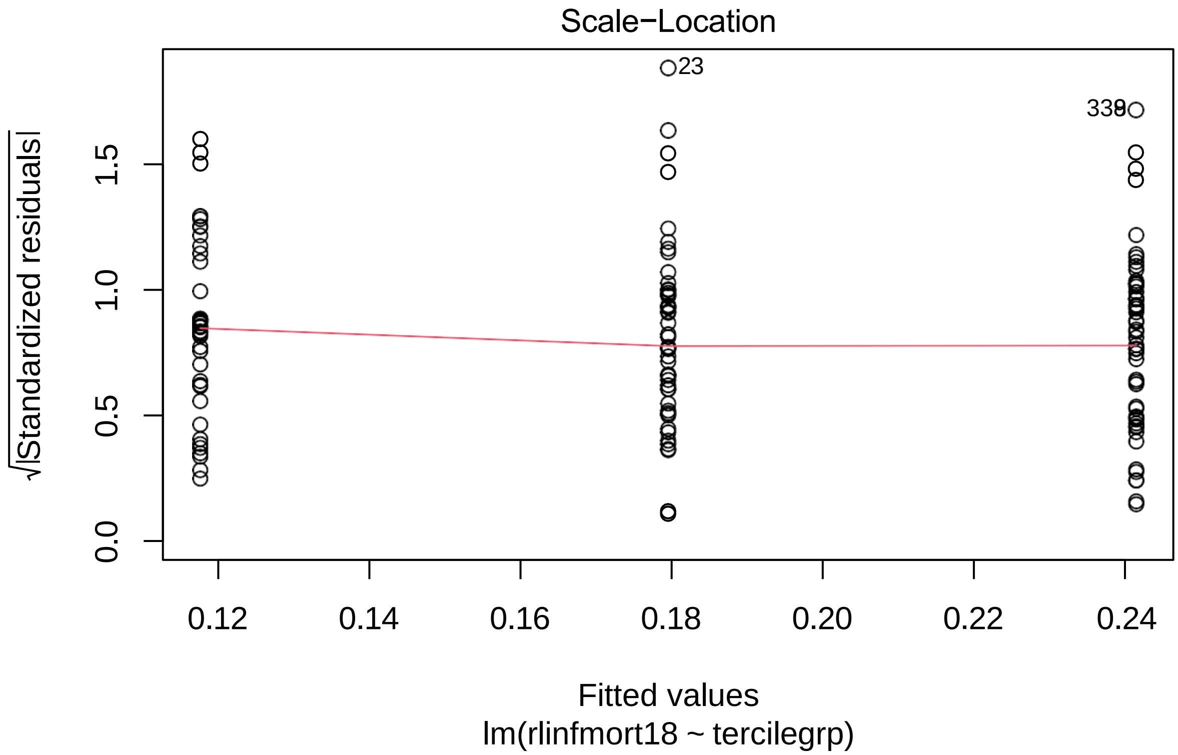

```
plot(resid(full.mod))
```

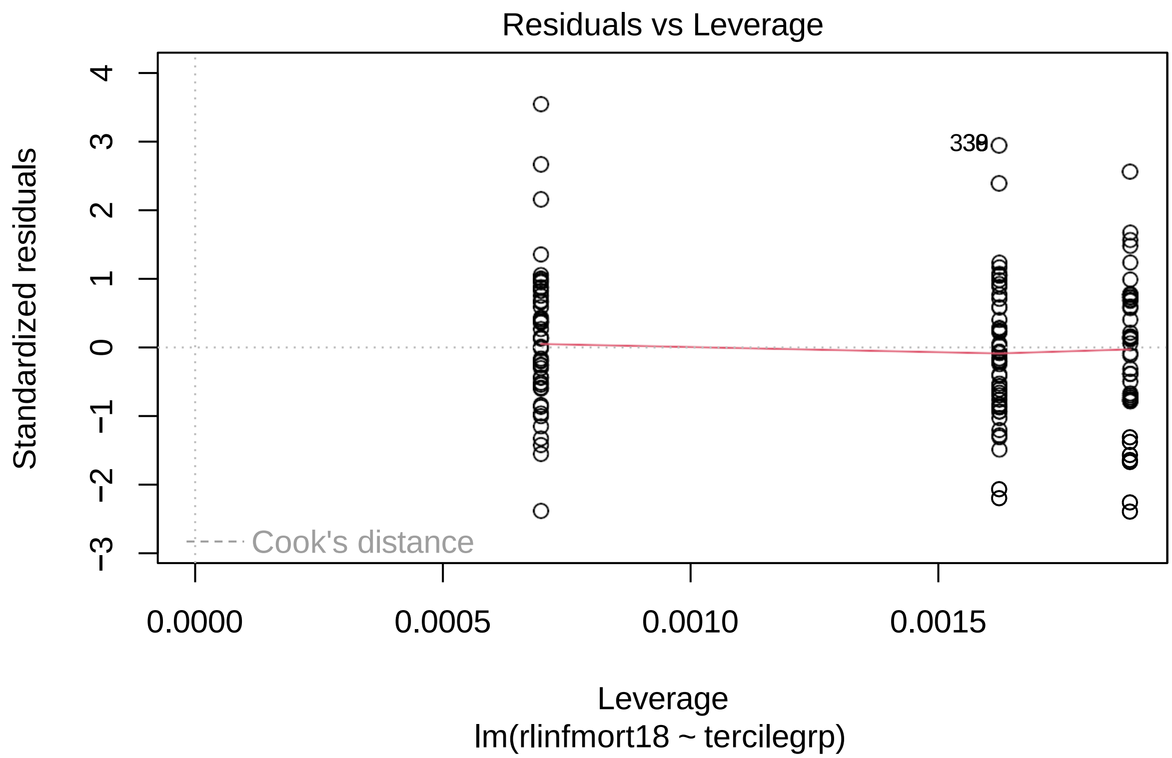



```
plot(null.mod)
```

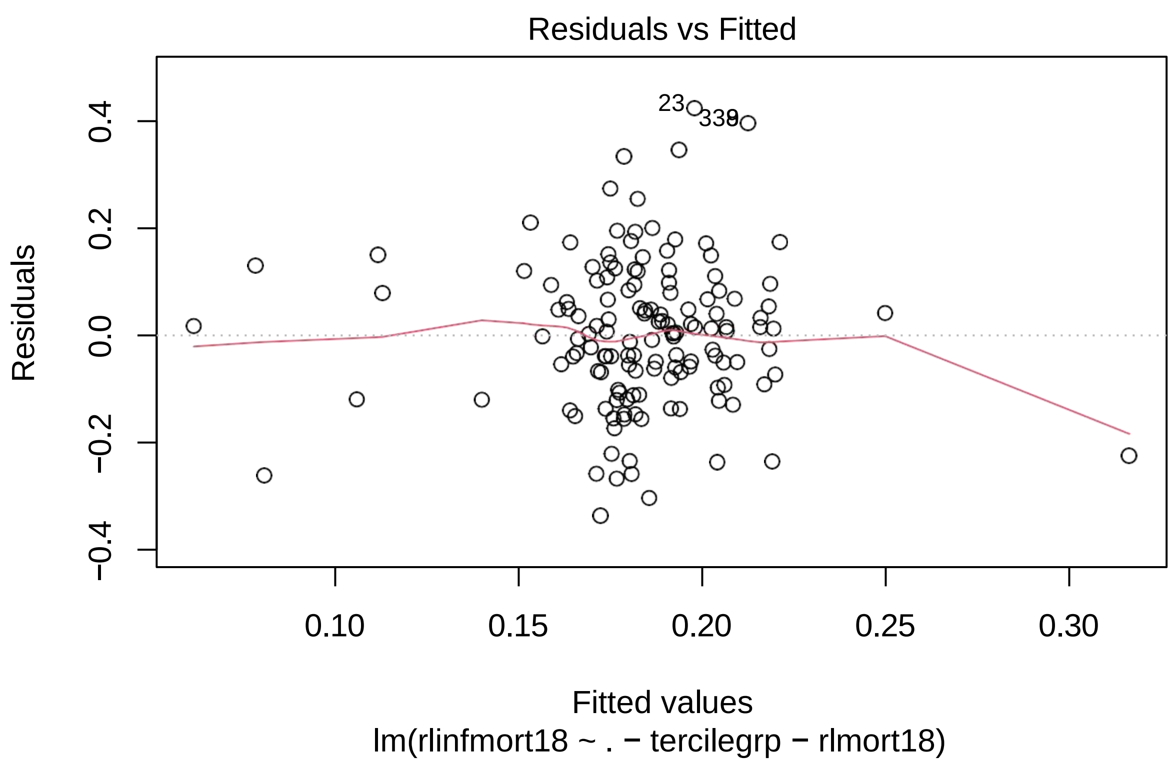


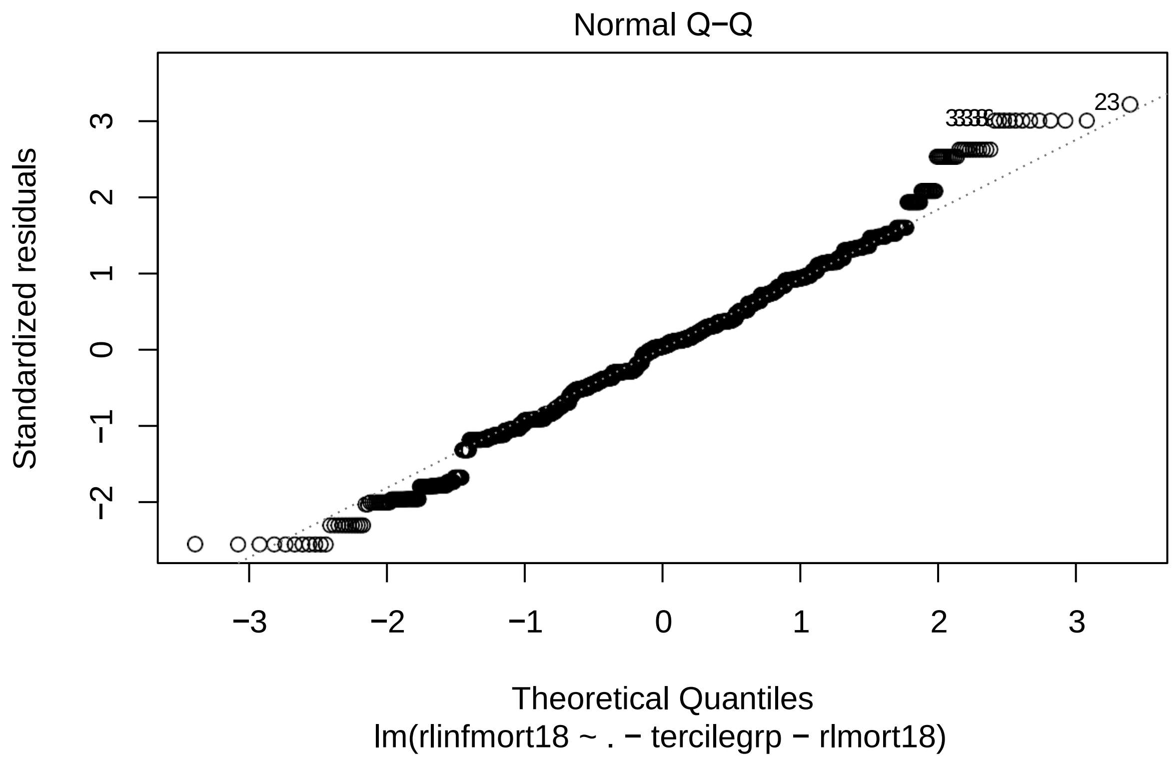

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(rlinfmort18 ~ tercilegrp)
23
−3
−2
−1
0
1
2
3
4

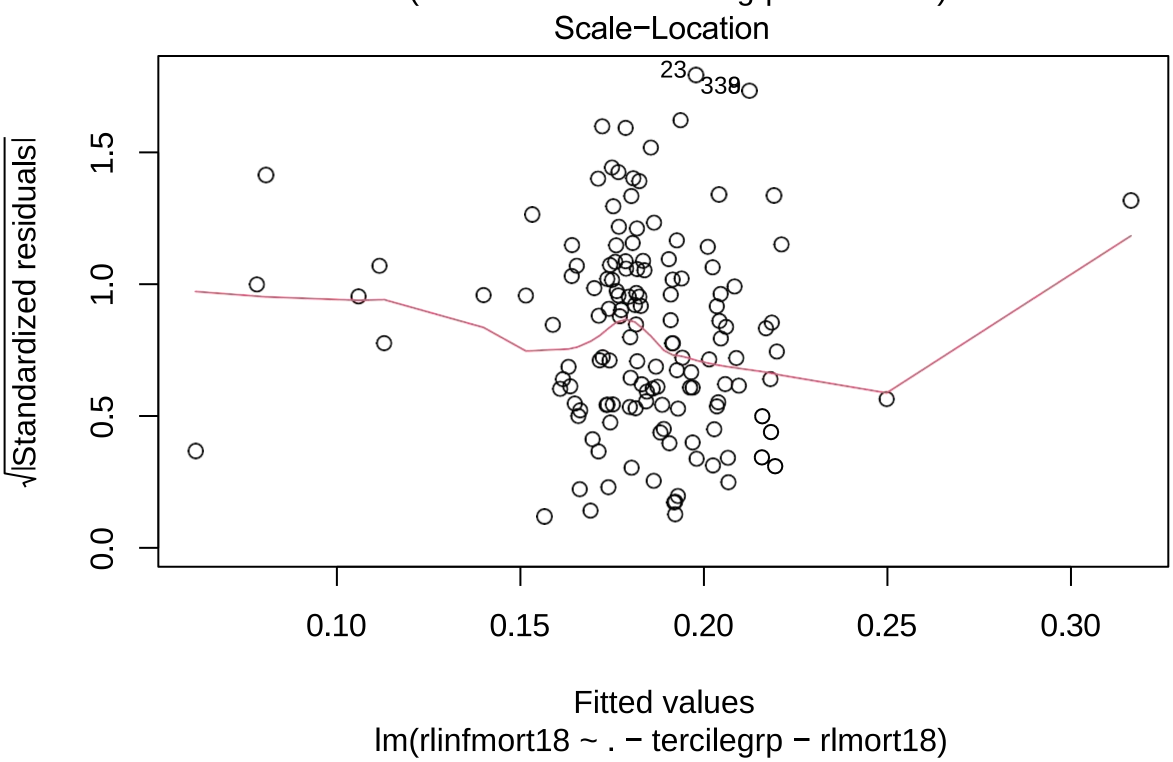

Scale−Location
√|Standardized residuals|
Fitted values
lm(rlinfmort18 ~ tercilegrp)
23
0.0
0.5
1.0
1.5
0.12
0.14
0.16
0.18
0.20
0.22
0.24

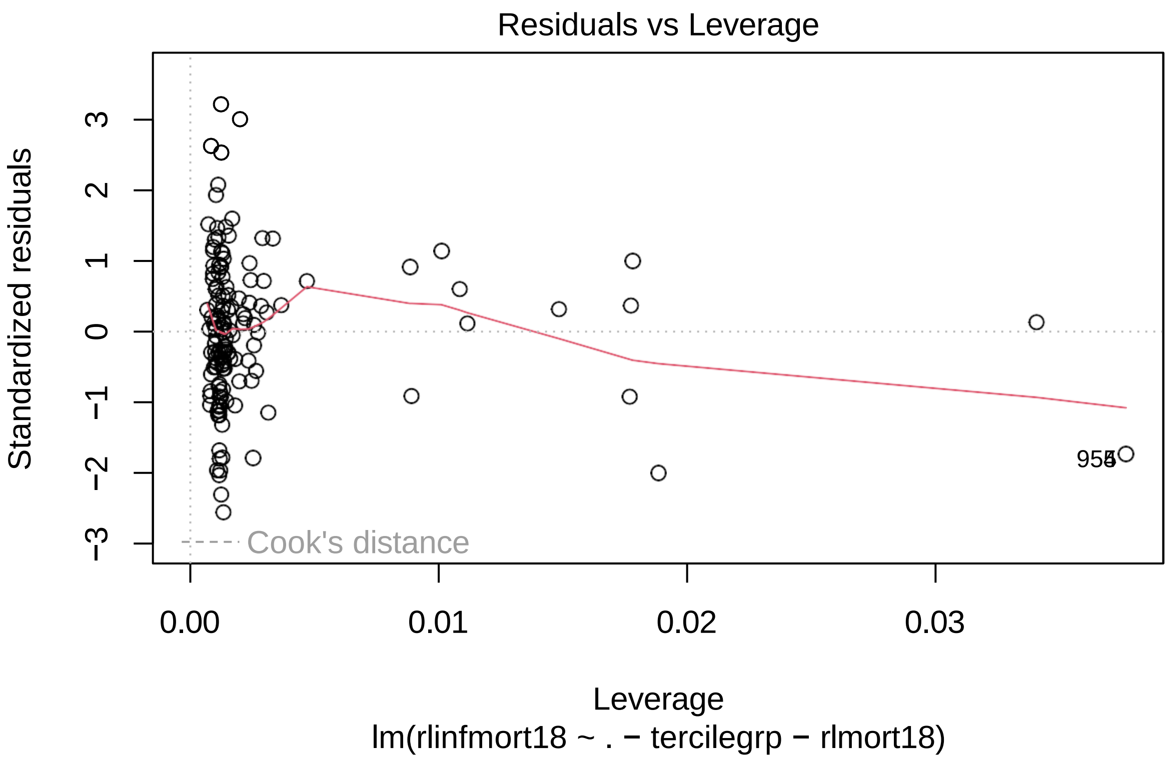

Residuals vs Leverage
Standardized residuals
Leverage
lm(rlinfmort18 ~ tercilegrp)
Cook's distance
339
4
3
2
1
0
−1
−2
−3
0.0000
0.0005
0.0010
0.0015

```
plot(alt.mod)
```

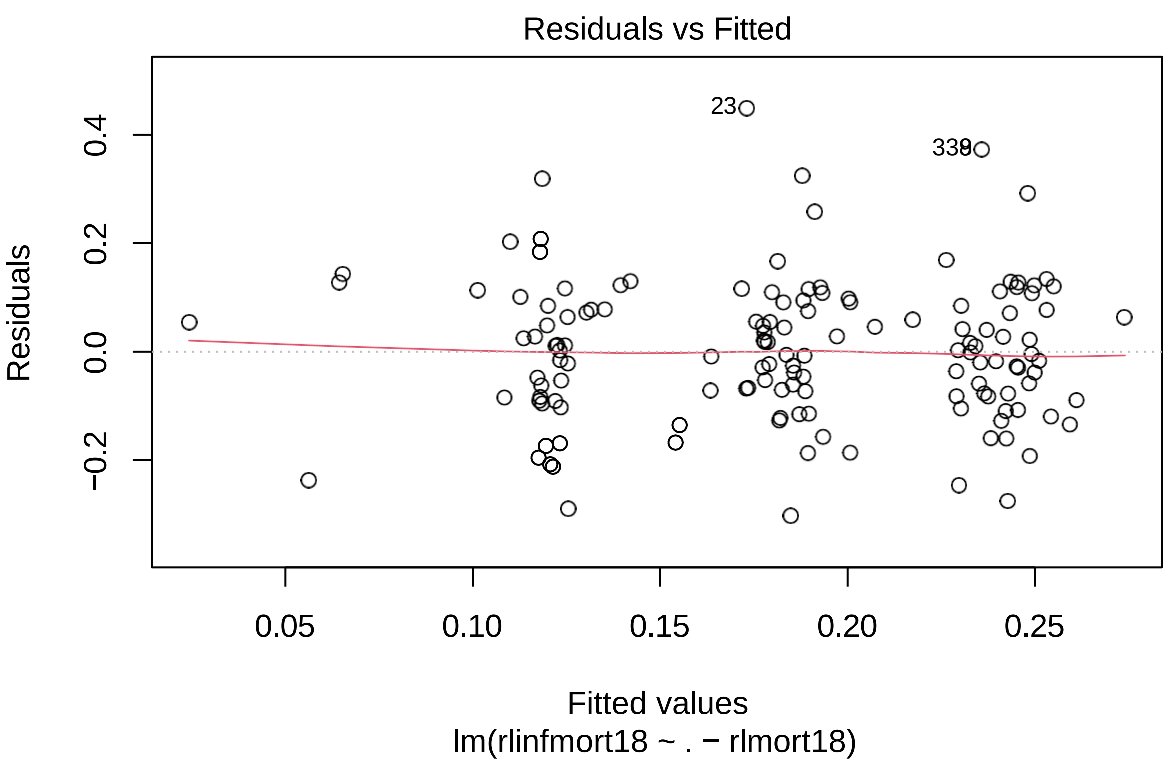

Residuals vs Fitted
Residuals
Fitted values
lm(rlinfmort18 ~ . − tercilegrp − rlmort18)
23
339
0.4
0.2
0.0
−0.2
−0.4
0.10
0.15
0.20
0.25
0.30

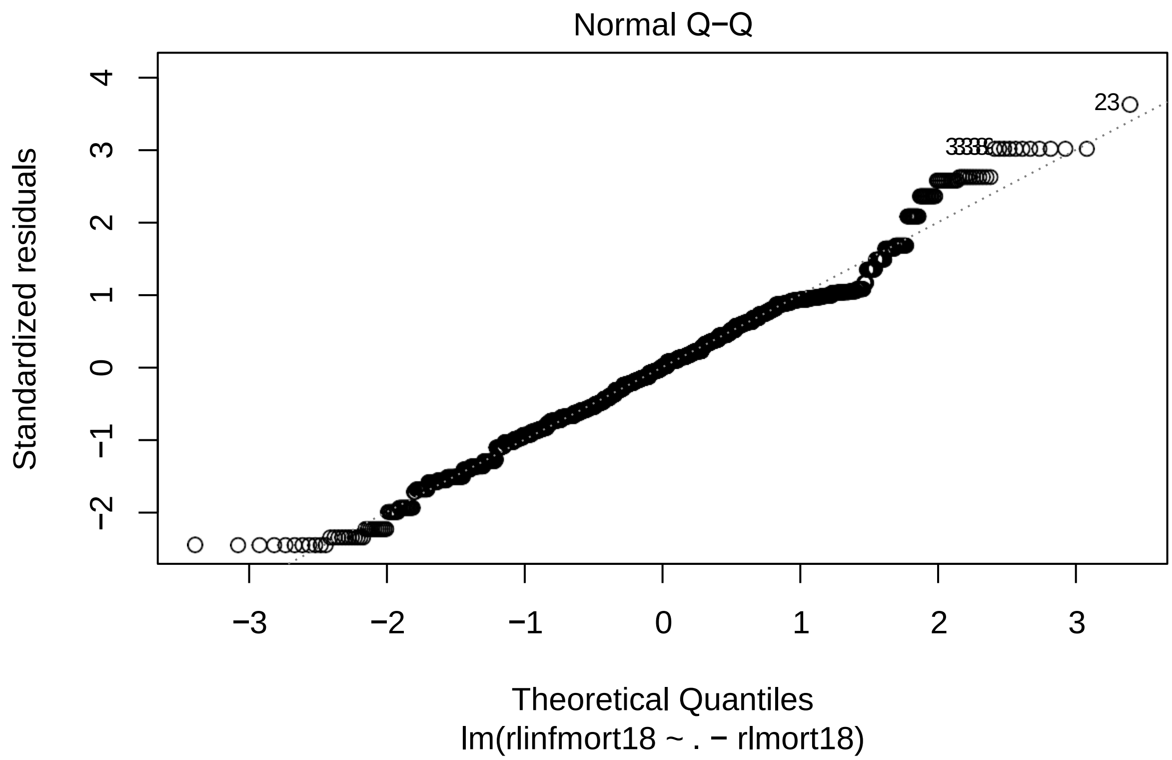

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(rlinfmort18 ~ . − tercilegrp − rlmort18)
23

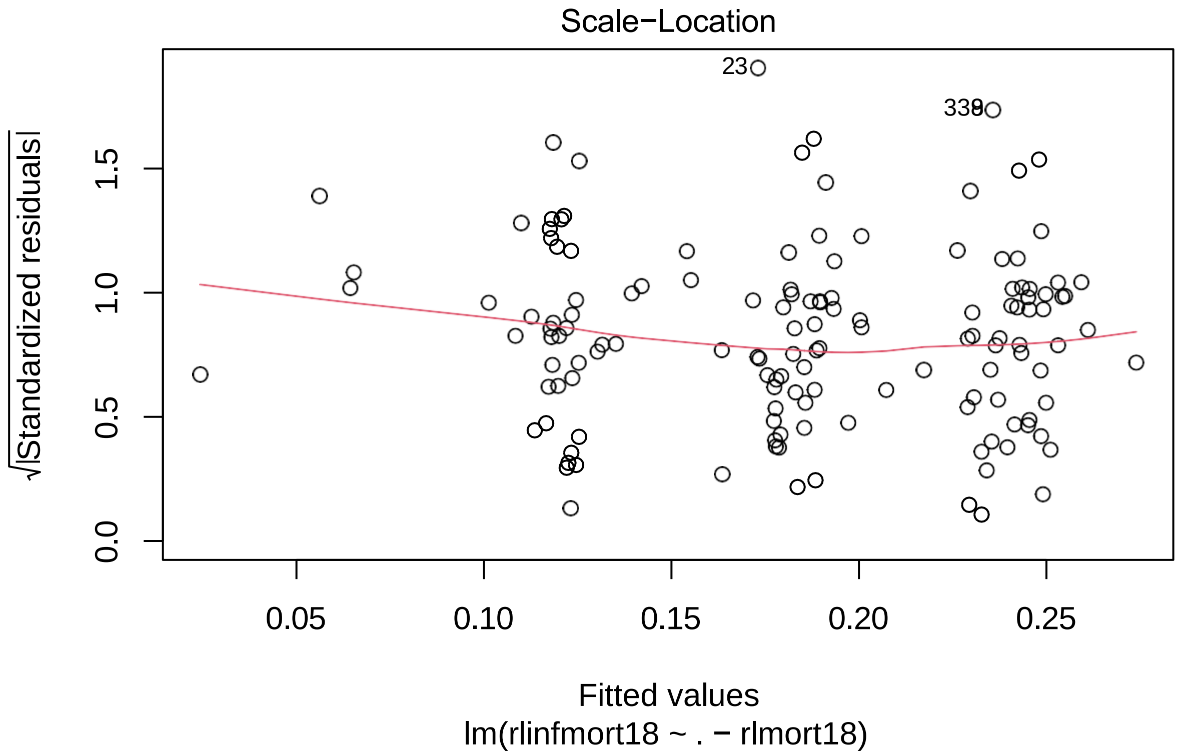

Scale−Location
√|Standardized residuals|
Fitted values
lm(rlinfmort18 ~ . − tercilegrp − rlmort18)
23

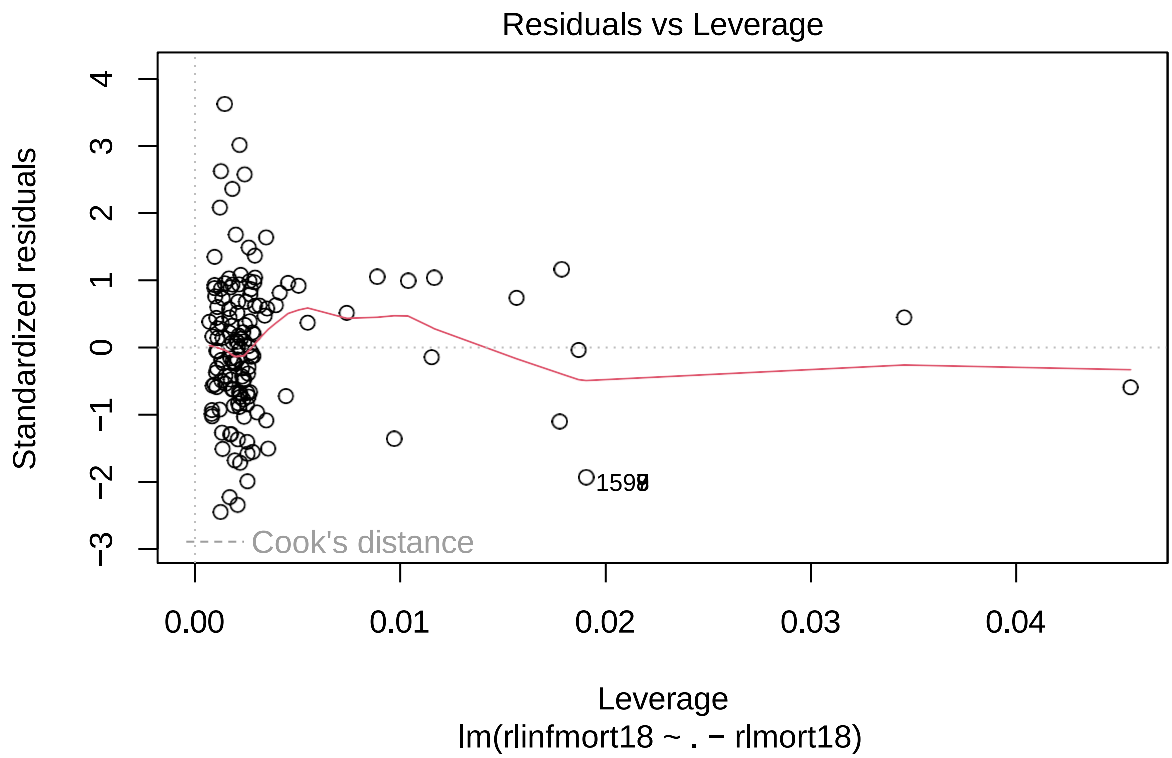



```
plot(full.mod)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(rlinfmort18 ~ . − rlmort18)
23


Scale−Location
√|Standardized residuals|
Fitted values
lm(rlinfmort18 ~ . − rlmort18)
23




```
cor(data.copy[, c(1,2,5,6)])
```

```
##                 pop1910  typhoidave tercilegrp  swhite1910
## pop1910      1.00000000 -0.06884575  0.5341894   0.1132307
## typhoidave  -0.06884575  1.00000000 -0.3104709  -0.3169752
## tercilegrp   0.53418943 -0.31047088  1.0000000   0.1378507
## swhite1910   0.11323067 -0.31697523  0.1378507   1.0000000
```

```
plot(data.copy$tercilegrp, data.copy$rlinfmort18, main="Excess Infant Mortality Rates vs. Coal Capacity
abline(lm(data.copy$rlinfmort18 ~ data.copy$tercilegrp))
```

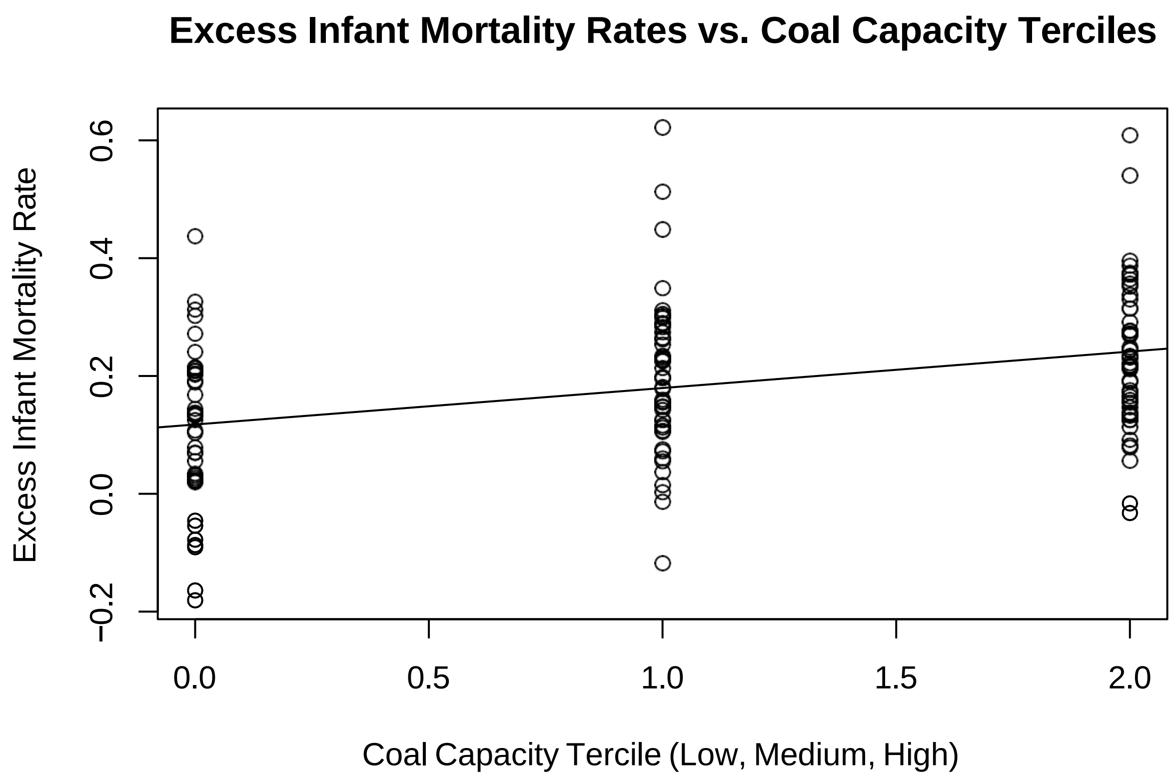


Figure 2

```
plot(data.copy$rlinfmort18 ~ data.copy$typhoidave, main="Excess Infant Mortality Rates vs.  Poor Water Q
abline(lm(data.copy$rlinfmort18 ~ data.copy$typhoidave))
```

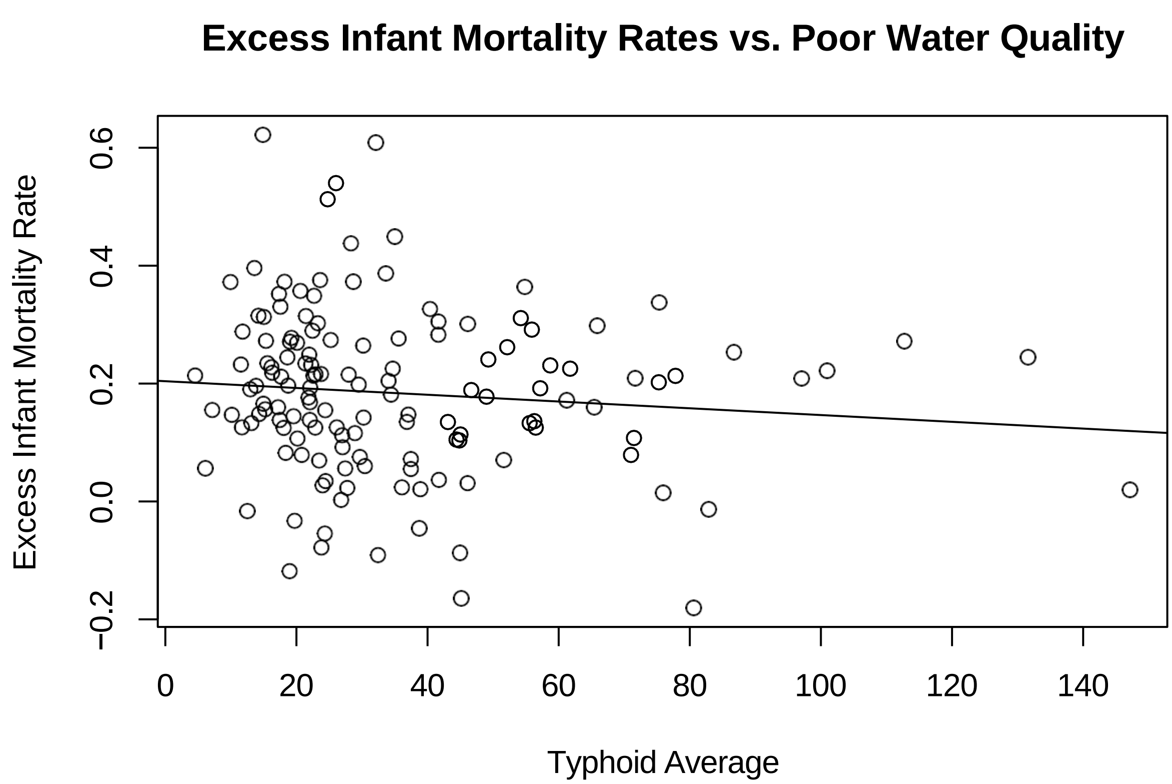
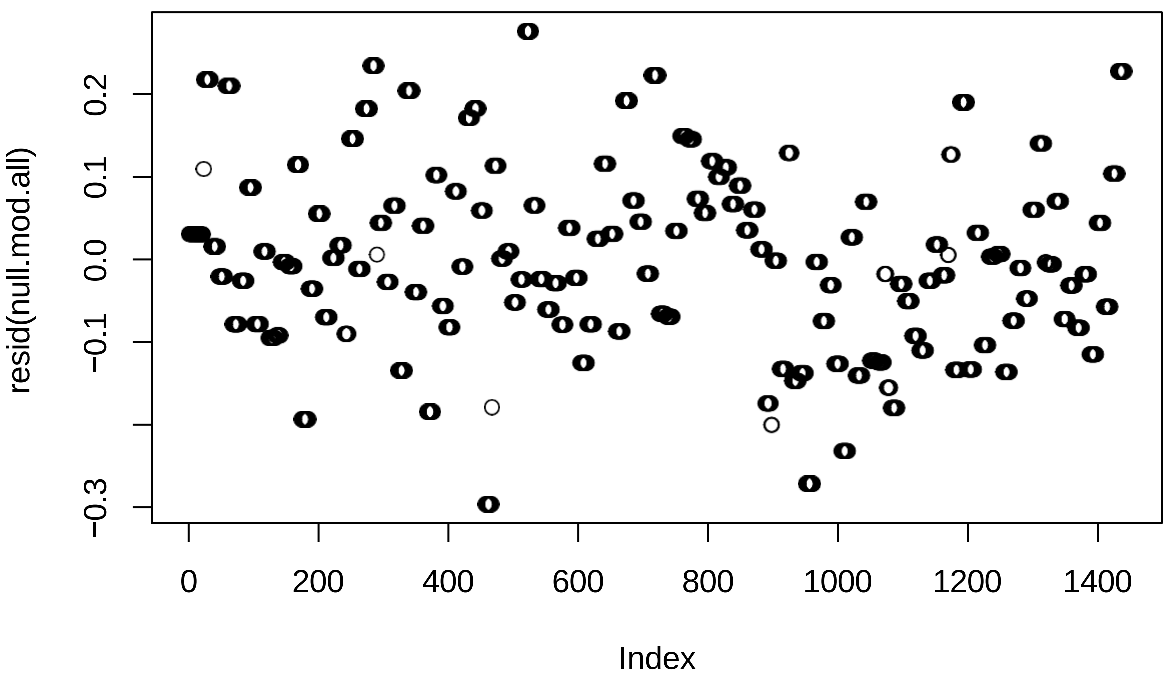
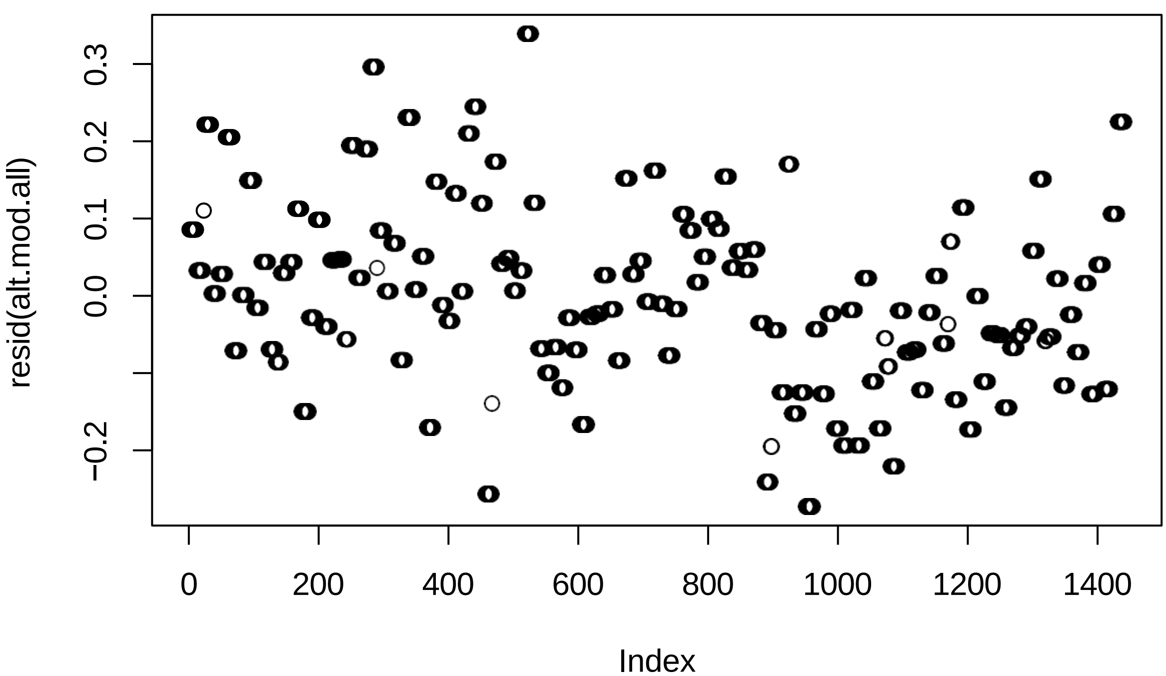


Figure 6

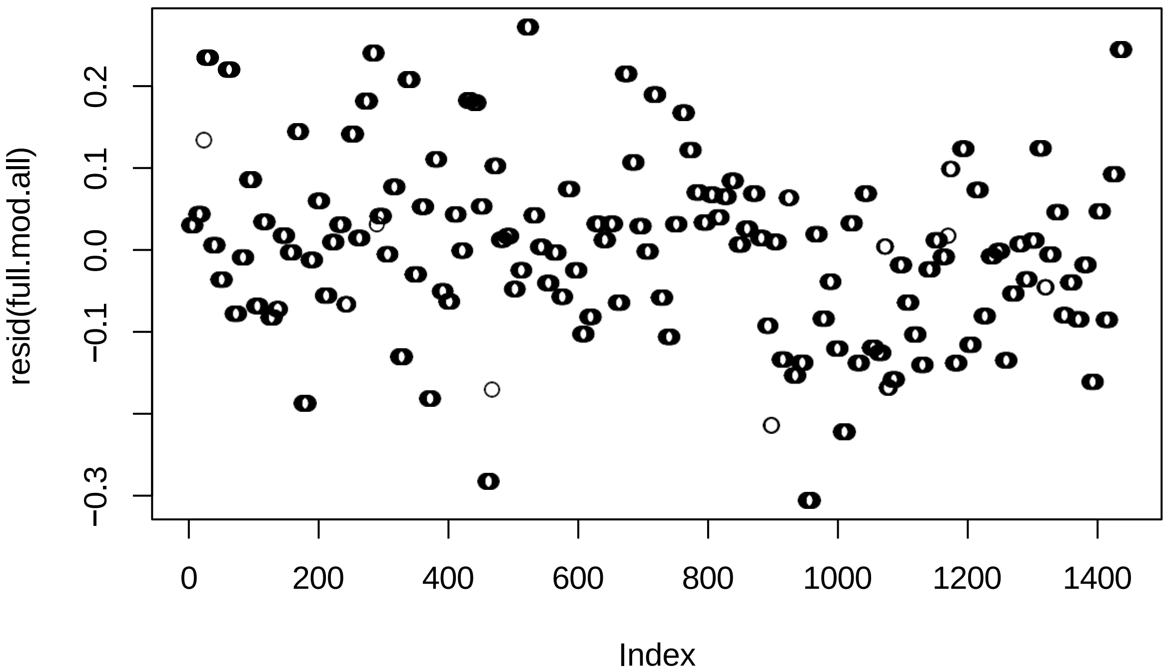
```
plot(resid(null.mod.all))
```



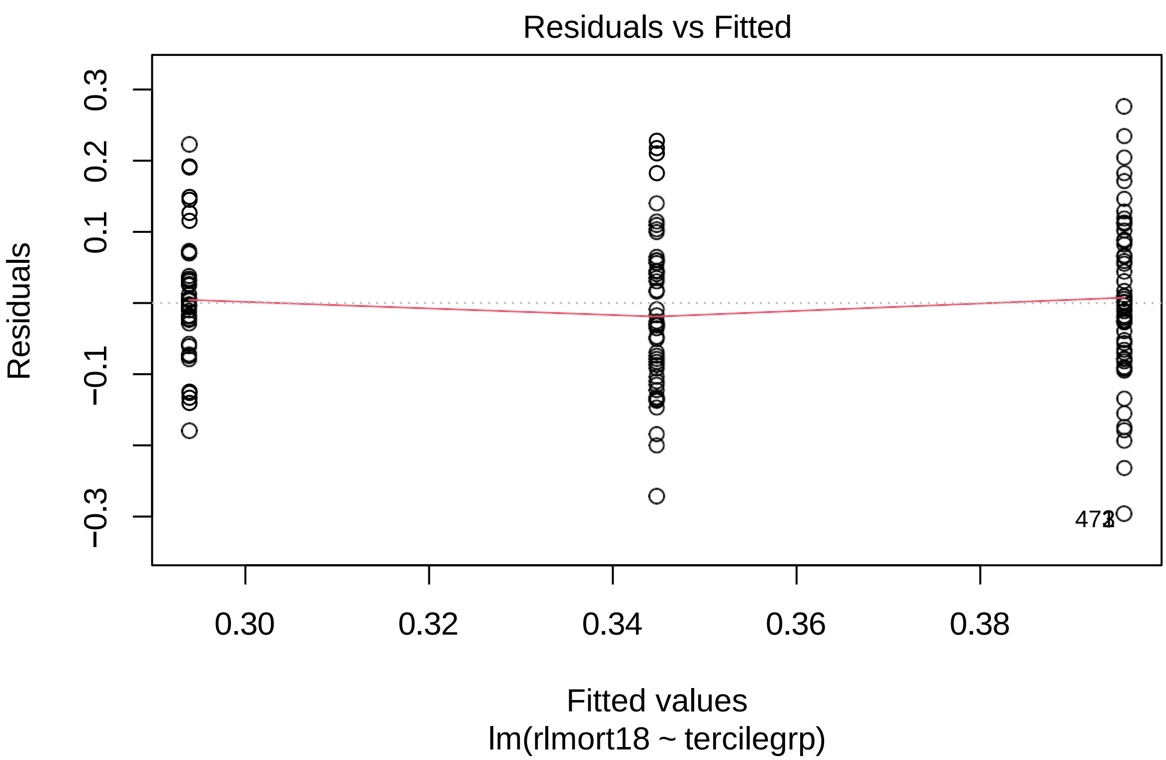

```
plot(resid(alt.mod.all))
```

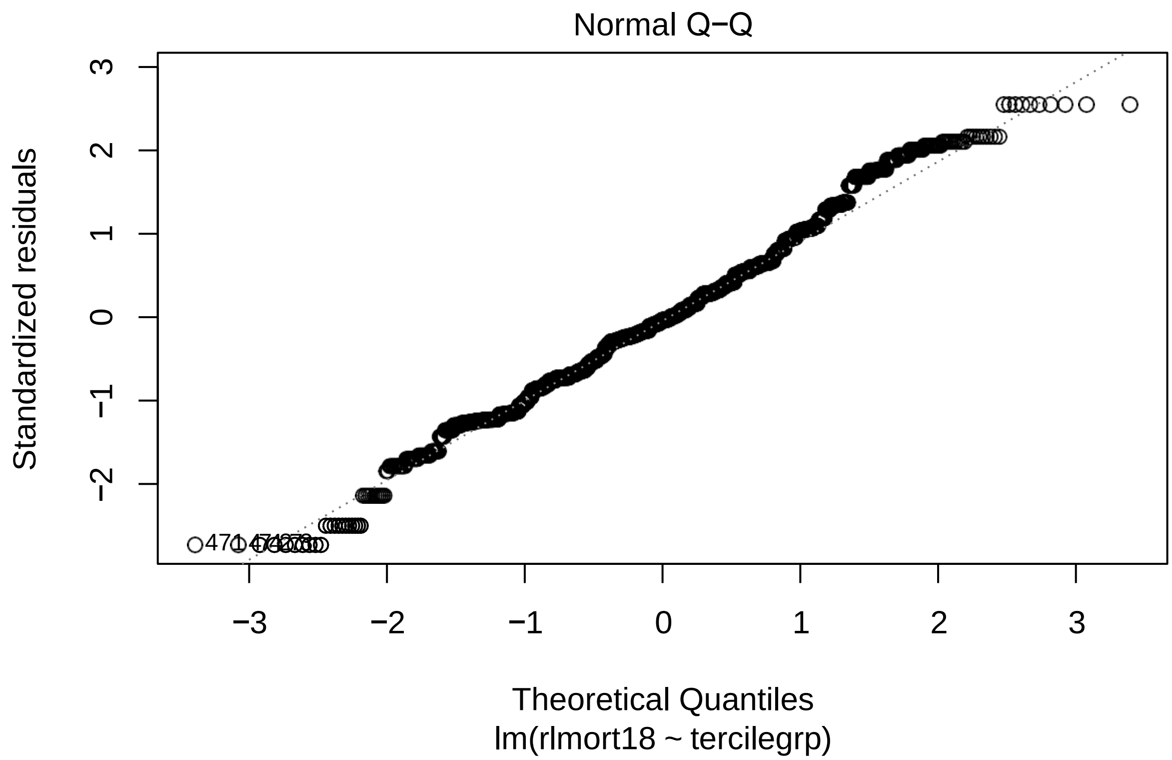


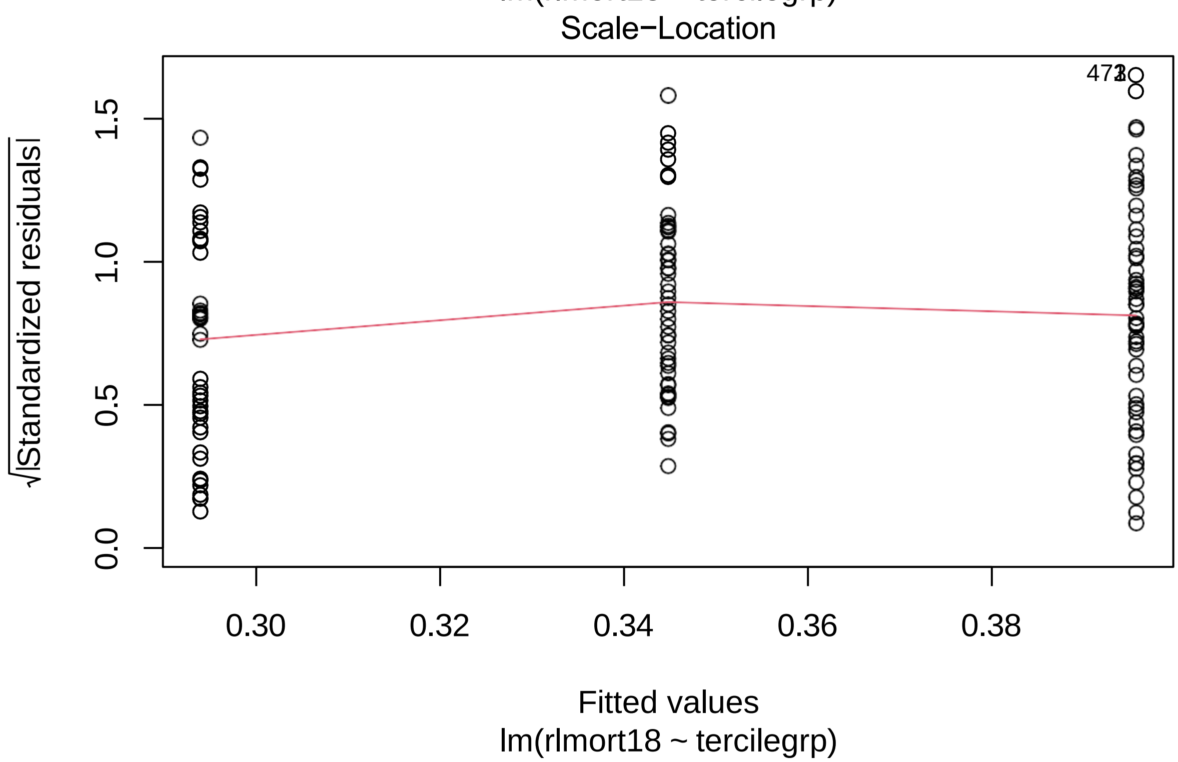

```
plot(resid(full.mod.all))
```

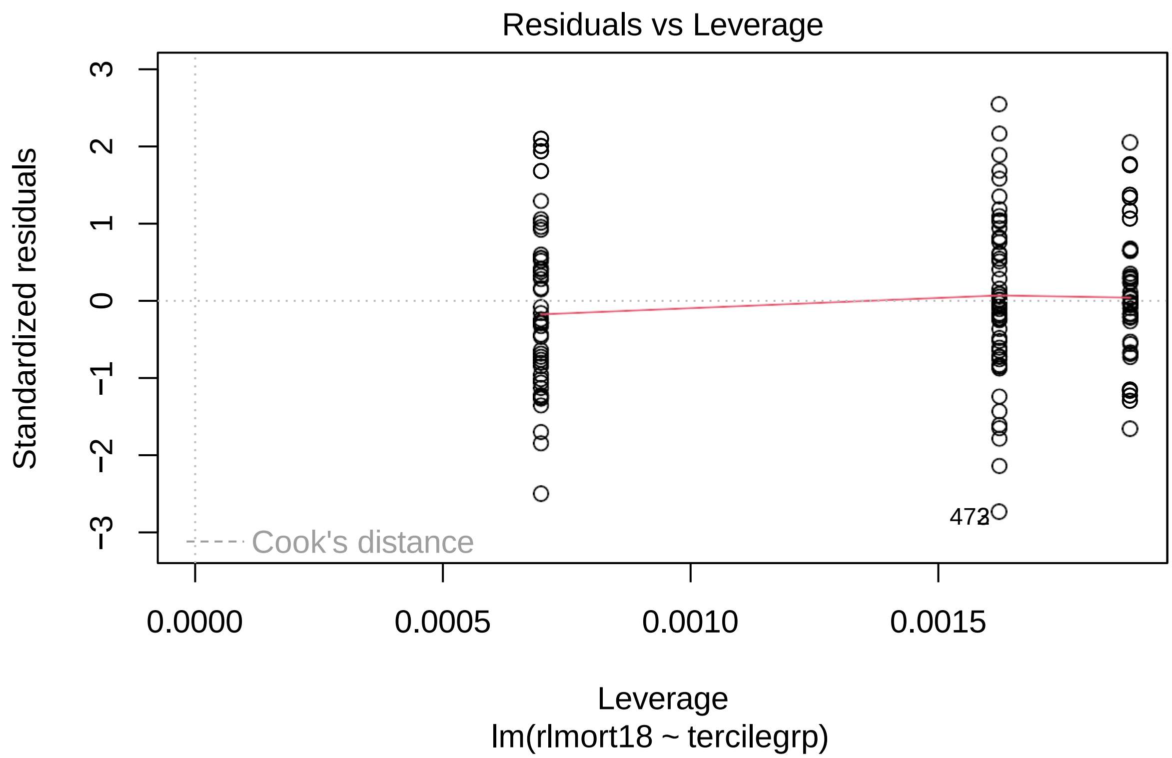



```
plot(null.mod.all)
```

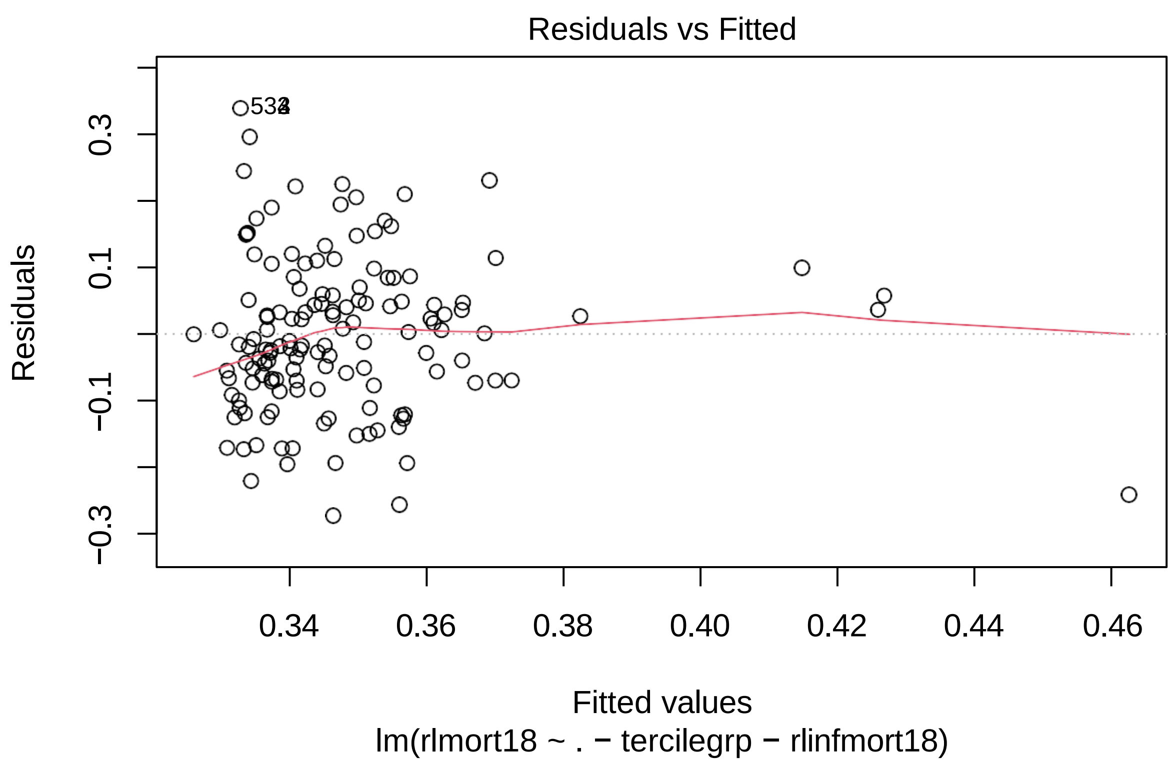


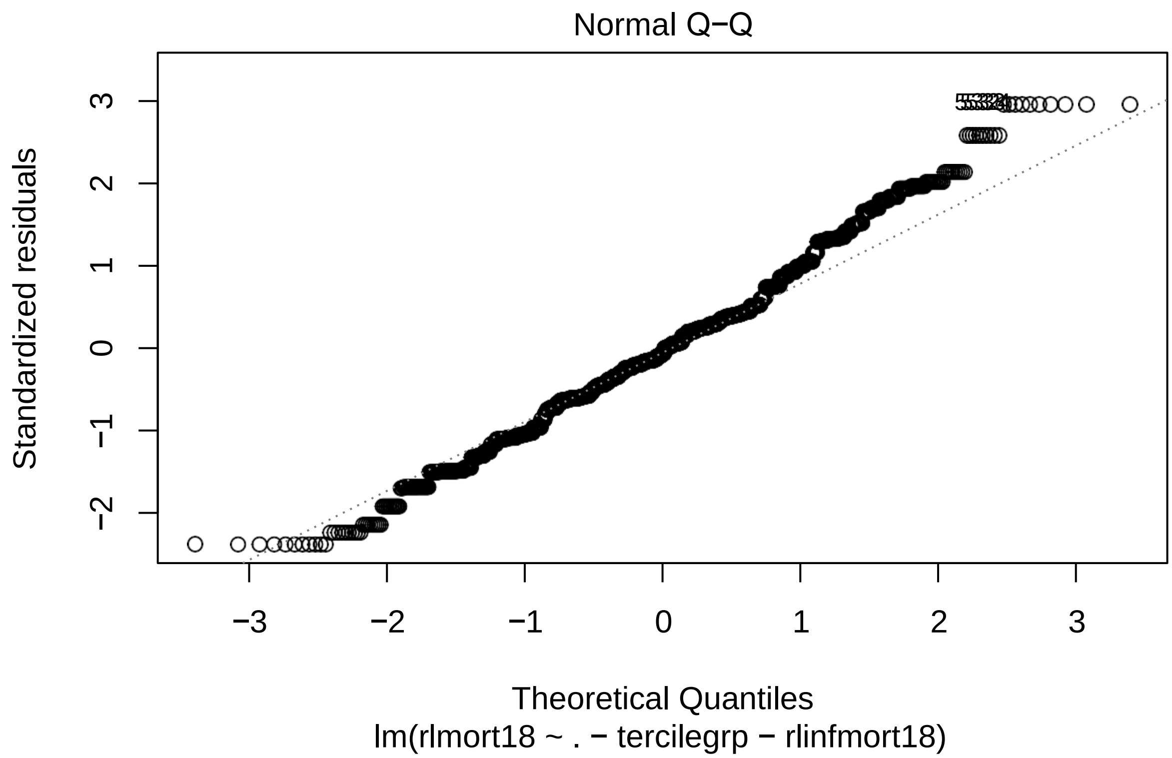

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(rlmort18 ~ tercilegrp)

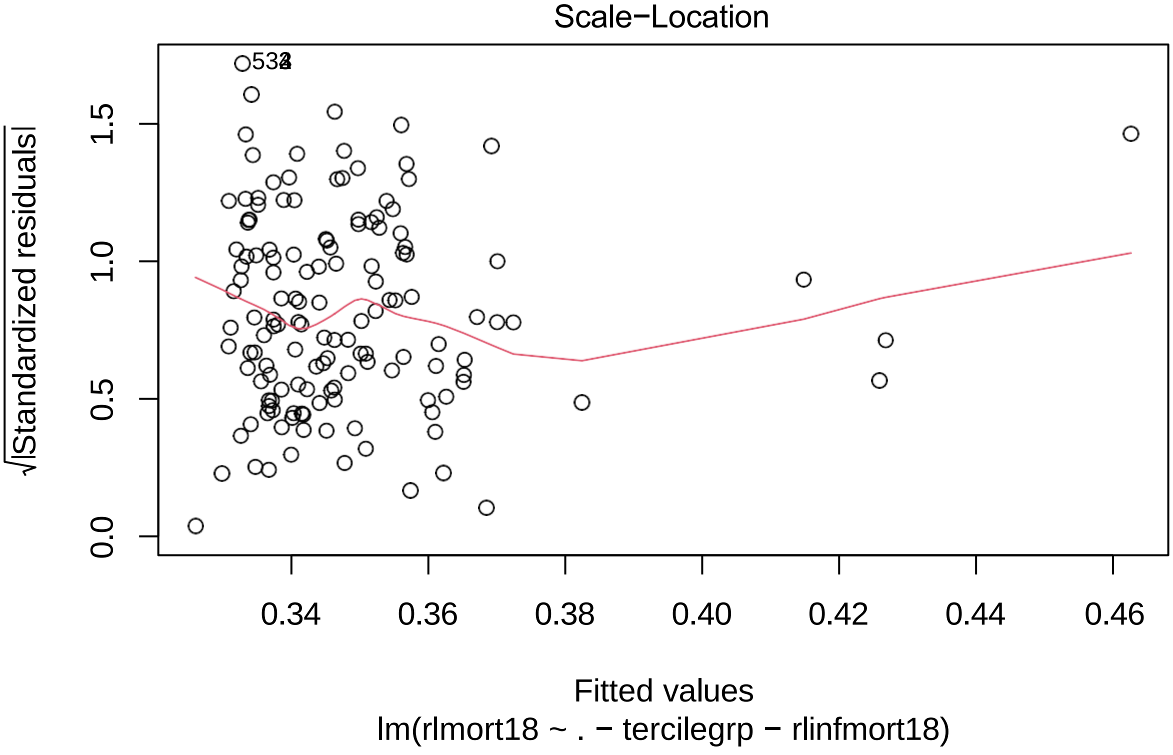

Scale−Location
√|Standardized residuals|
Fitted values
lm(rlmort18 ~ tercilegrp)

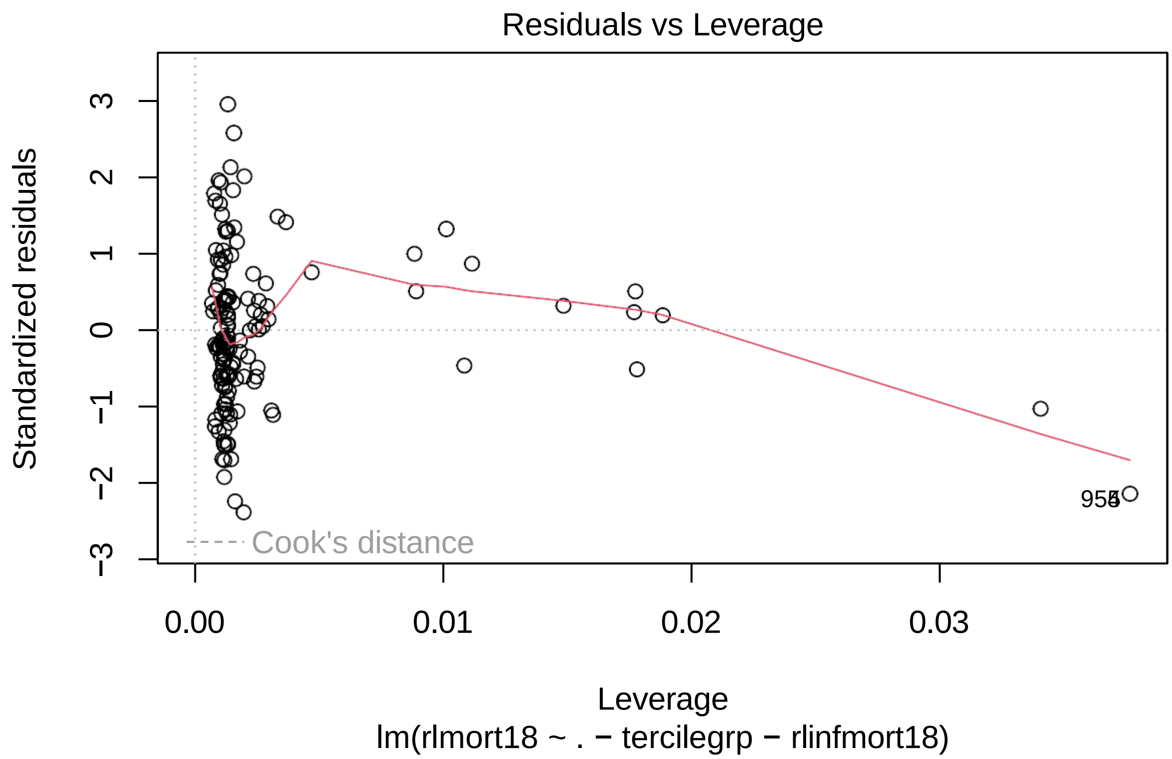



```
plot(alt.mod.all)
```

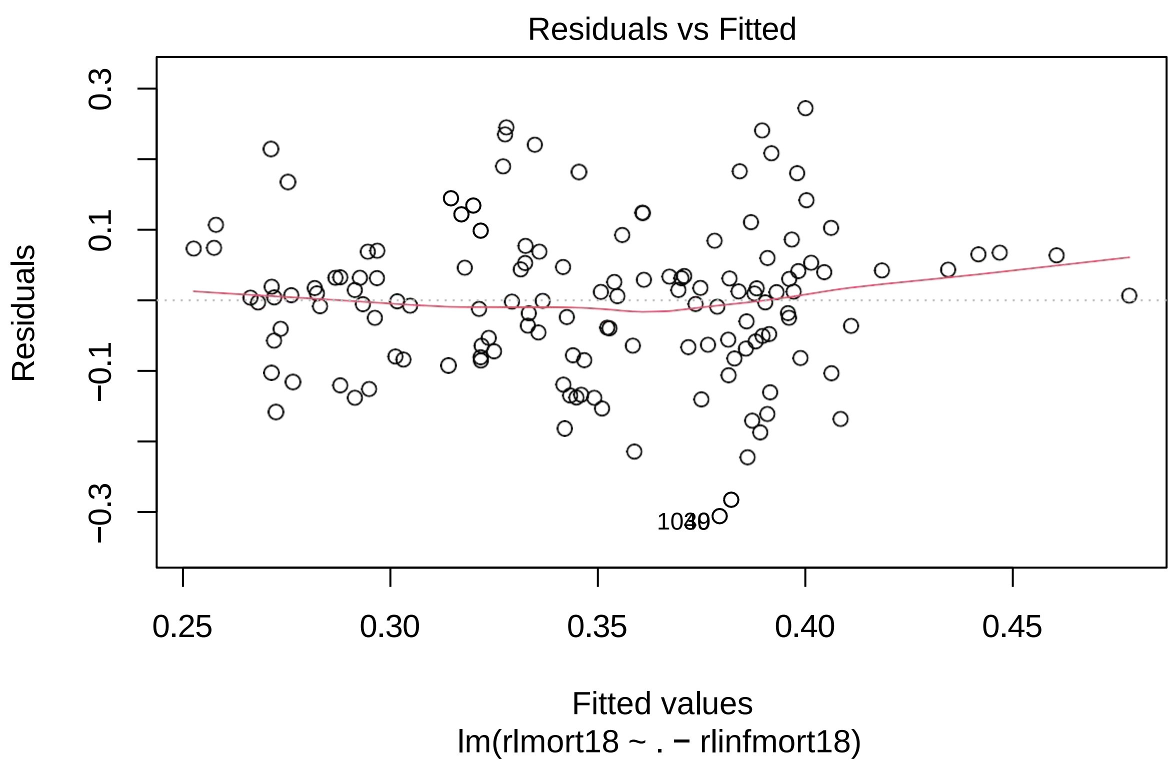


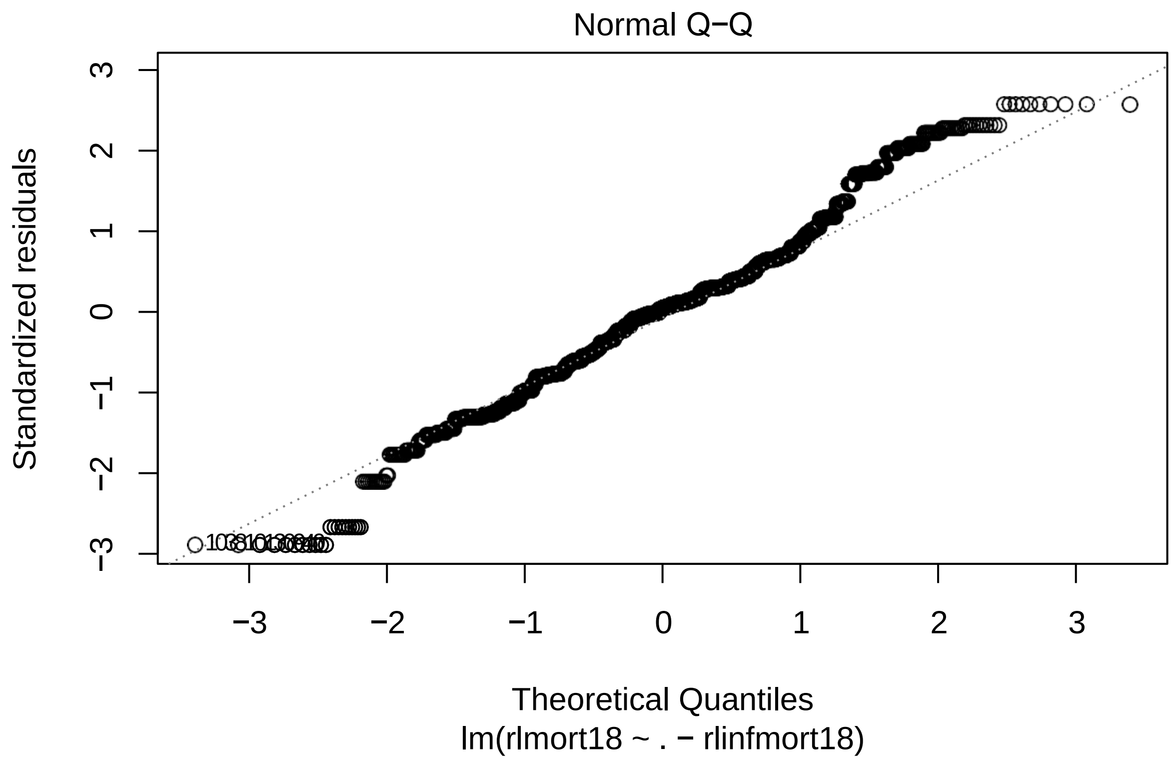

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(rlmort18 ~ . − tercilegrp − rlinfmort18)

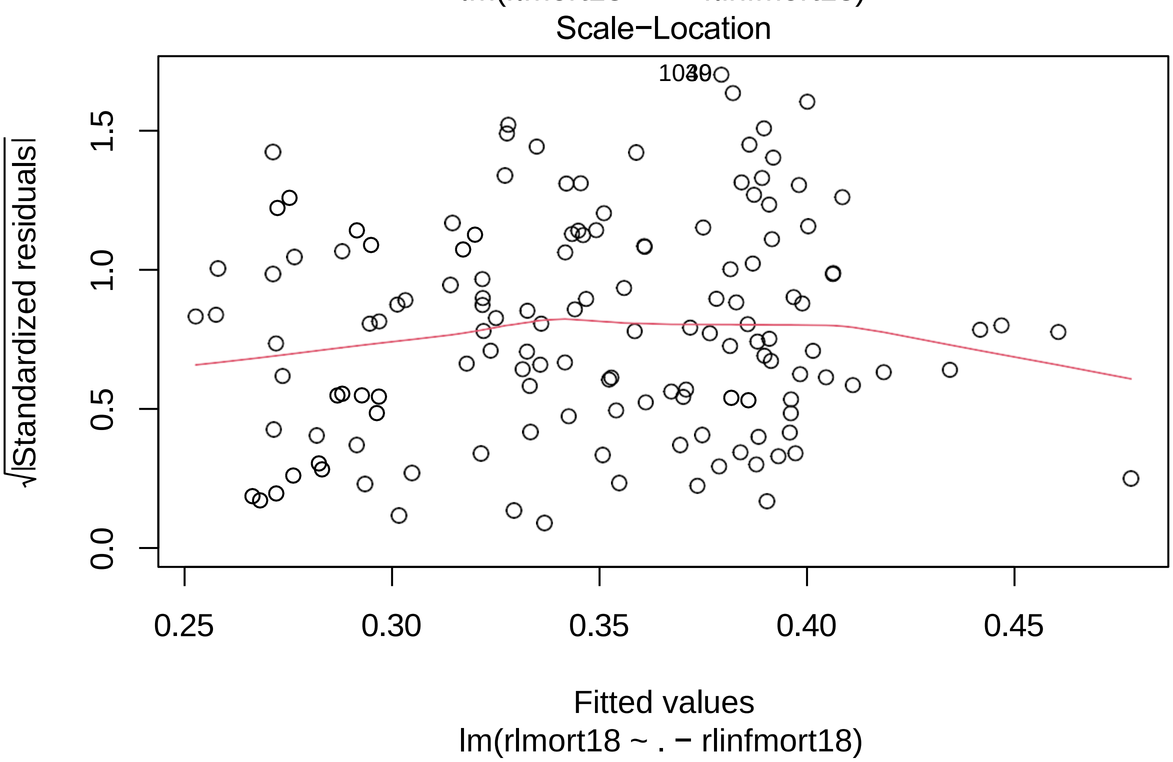

Scale−Location
√|Standardized residuals|
Fitted values
lm(rlmort18 ~ . − tercilegrp − rlinfmort18)

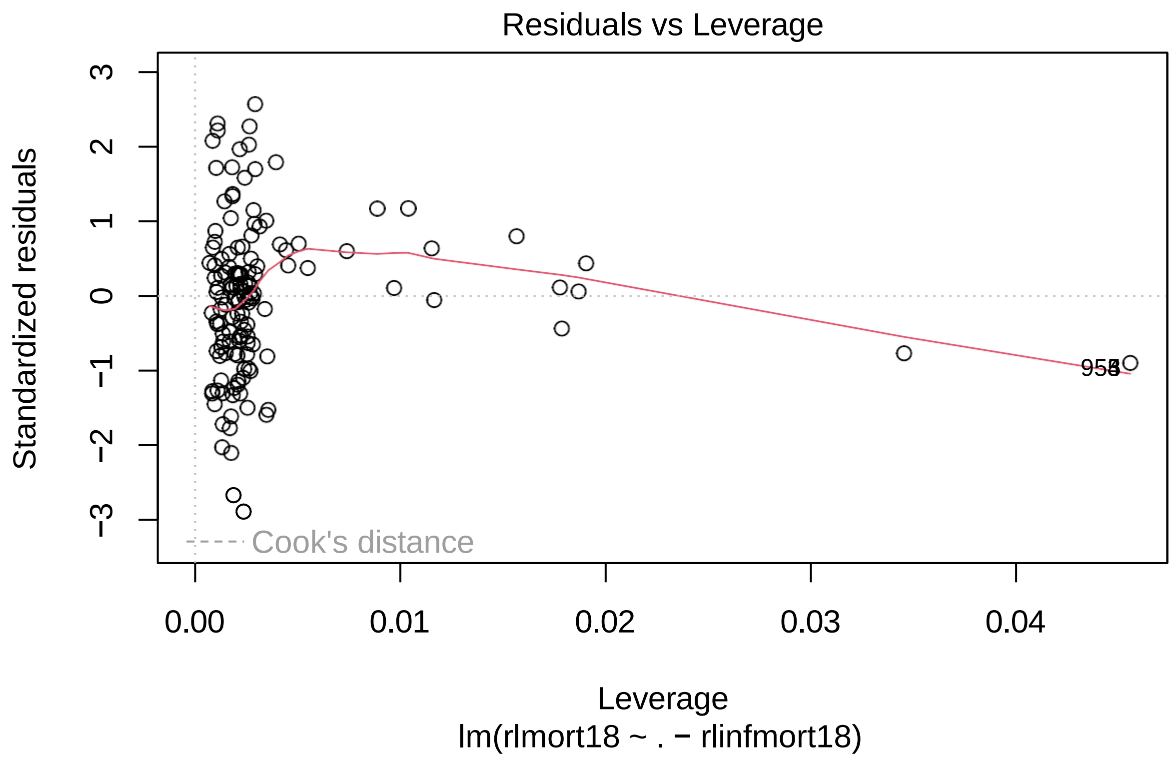



```
plot(full.mod.all)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(rlmort18 ~ . − rlinfmort18)


Scale−Location
√|Standardized residuals|
Fitted values
lm(rlmort18 ~ . − rlinfmort18)




```
plot(data.copy$tercilegrp, data.copy$rlmort18, main="Excess Mortality Rates vs. Coal Capacity Terciles"
abline(lm(data.copy$rlmort18 ~ data.copy$tercilegrp))
```

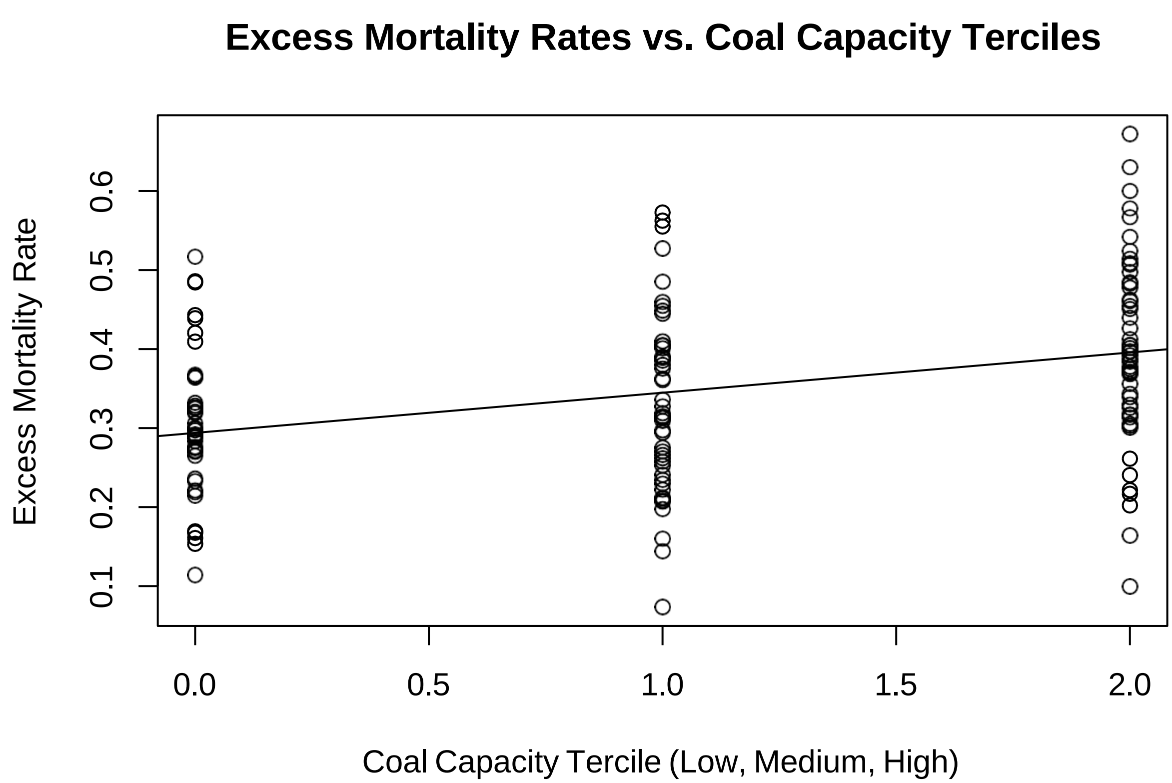


Figure 2

```
plot(data.copy$rlmort18 ~ data.copy$typhoidave, main="Excess Mortality Rates vs. Poor Water Quality", x
abline(lm(data.copy$rlmort18 ~ data.copy$typhoidave))
```

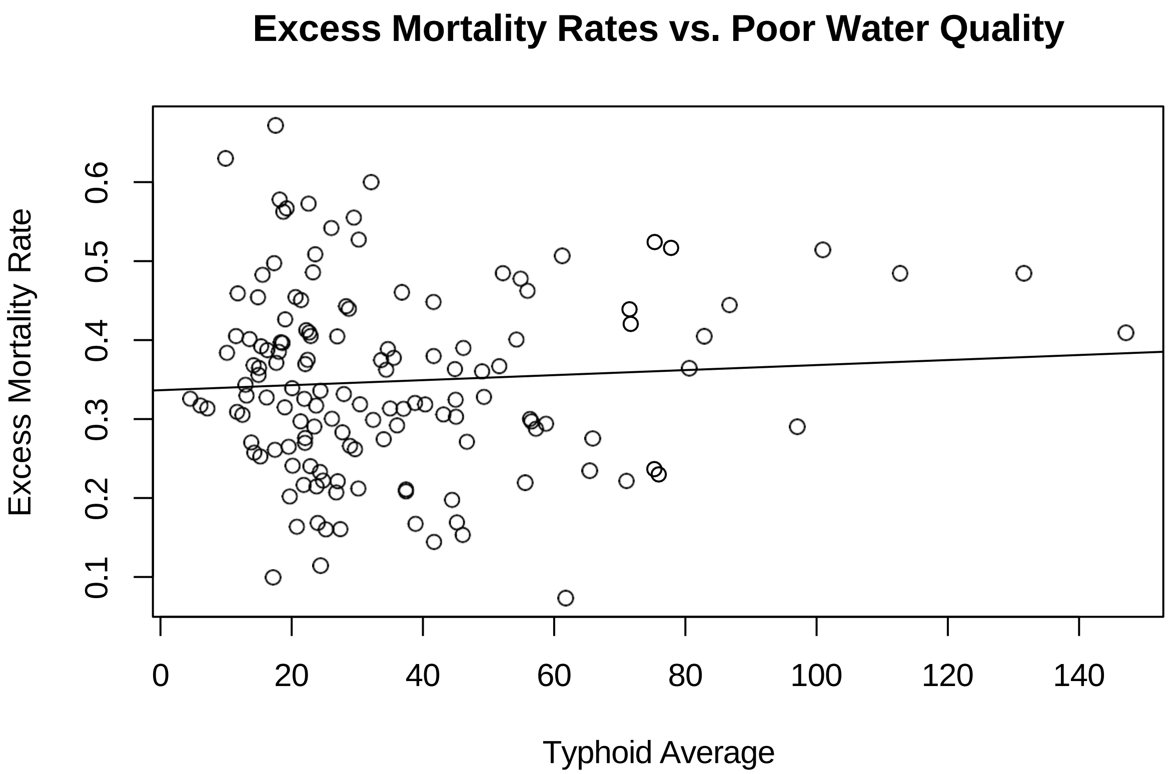


Figure 6

Results

```
# Resample bootstrap

B = 1000
null.mse = c()
alt.mse = c()
full.mse = c()

MSE_stat <- function(mod, bs) {
  mean((bs$rlinfmort18 - predict(mod, newdata=bs))^2)
}

for (i in 1:B) {
  boot.sample = sample_n(data.copy, nrow(data.copy), replace=TRUE)

  null.mod = lm(rlinfmort18 ~ tercilegrp, data=boot.sample)
  null.mse[i] = MSE_stat(null.mod, boot.sample)

  alt.mod = lm(rlinfmort18 ~ .-tercilegrp, data=boot.sample)
  alt.mse[i] = MSE_stat(alt.mod, boot.sample)
```

```
  full.mod = lm(rlinfmort18 ~ ., data=boot.sample)
  full.mse[i] = MSE_stat(full.mod, boot.sample)
}

(null.ci = quantile(null.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01423934 0.01679352
```

```
(alt.ci = quantile(alt.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01144480 0.01359612
```

```
(full.ci = quantile(full.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01102543 0.01313308
```

Null model is significantly worse.

```
#  Cross-validation  MSE
# CV implementation: p.27 of CV slides

set.seed(0)

MSE_stat <- function(mod, bs) {
  mean((bs$rlinfmort18 - predict(mod, newdata=bs))^2)
}

null.mse = c()
alt.mse = c()
full.mse = c()

for(i in 1:1000) {
  N <- nrow(data.copy)
  train_idx <- sample(seq(N), size=floor(0.7*N))
  train <- data.copy[train_idx,]
  test <- data.copy[-train_idx,]

  null.train.mod = lm(rlinfmort18 ~ tercilegrp, data=train)

  alt.train.mod = lm(rlinfmort18 ~ pop1910 + swhite1910 +typhoidave, data=train)

  full.train.mod = lm(rlinfmort18 ~ tercilegrp + pop1910 + swhite1910 + typhoidave, data=train)

  null.mse[i] = MSE_stat(null.train.mod, test)
  alt.mse[i] = MSE_stat(alt.train.mod, test)
  full.mse[i] = MSE_stat(full.train.mod, test)
}

(null.ci = quantile(null.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01371550 0.01752863
```

```r
(alt.ci = quantile(alt.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01531652 0.01959108
```

```r
(full.ci = quantile(full.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01354764 0.01731043
```

None of the MSEs have a statistically significant difference.

```r
# Resample bootstrap

B = 1000
null.mse = c()
alt.mse = c()
full.mse = c()

MSE_stat <- function(mod, bs) {
  mean((bs$rlmort18 - predict(mod, newdata=bs))^2)
}

for (i in 1:B) {
  boot.sample = sample_n(data.copy, nrow(data.copy), replace=TRUE)

  null.mod = lm(rlmort18 ~ tercilegrp, data=boot.sample)
  null.mse[i] = MSE_stat(null.mod.all, boot.sample)

  alt.mod = lm(rlmort18 ~ .-tercilegrp-rlinfmort18, data=boot.sample)
  alt.mse[i] = MSE_stat(alt.mod.all, boot.sample)

  full.mod = lm(rlmort18 ~ .-rlinfmort18, data=boot.sample)
  full.mse[i] = MSE_stat(full.mod.all, boot.sample)
}

(null.ci = quantile(null.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01086128 0.01264034
```

```r
(alt.ci = quantile(alt.mse, c(0.025, 0.975)))
```

```
##        2.5%      97.5%
## 0.01212983 0.01409194
```

```r
(full.ci = quantile(full.mse, c(0.025, 0.975)))
```

```
##       2.5%      97.5% 
## 0.01030236 0.01205975
```

Full model is significantly better.

```r
set.seed(0)

MSE_stat <- function(mod, bs) {
  mean((bs$rlmort18 - predict(mod, newdata=bs))^2)
}

null.mse = c()
alt.mse = c()
full.mse = c()

for(i in 1:1000) {
  N <- nrow(data.copy)
  train_idx <- sample(seq(N), size=floor(0.7*N))
  train <- data.copy[train_idx,]
  test <- data.copy[-train_idx,]

  null.train.mod = lm(rlmort18 ~ tercilegrp, data=train)

  alt.train.mod = lm(rlmort18 ~ pop1910 + swhite1910 + typhoidave, data=train)

  full.train.mod = lm(rlmort18 ~ tercilegrp + pop1910 + swhite1910 + typhoidave, data=train)

  null.mse[i] = MSE_stat(null.train.mod, test)
  alt.mse[i] = MSE_stat(alt.train.mod, test)
  full.mse[i] = MSE_stat(full.train.mod, test)
}

(null.ci = quantile(null.mse, c(0.025, 0.975)))
```

```
##       2.5%      97.5% 
## 0.01048301 0.01306514
```

```r
(alt.ci = quantile(alt.mse, c(0.025, 0.975)))
```

```
##       2.5%      97.5% 
## 0.01171637 0.01479805
```

```r
(full.ci = quantile(full.mse, c(0.025, 0.975)))
```

```
##        2.5%       97.5% 
## 0.009836687 0.012591176
```

None of the MSEs have a statistically significant difference.

Bootstrapping tended to favor the full model.

Cross-Validation did not favor any model. Cross-Validation tests on unseen data so I will put more weight towards that.