Stockholms universitet

Investigating nowcasting of COVID-19 infected using linear regression

Lukas Fredriksson

Kandidatuppsats i matematisk statistik
Bachelor Thesis in Mathematical Statistics

Kandidatuppsats 2021:20
Matematisk statistik
September 2021

www.math.su.se

Matematisk statistik
Matematiska institutionen
Stockholms universitet
106 91 Stockholm

Stockholm University

Mathematical Statistics
Stockholm University
Bachelor Thesis **2021:20**
http://www.math.su.se

Investigating nowcasting of COVID-19 infected using linear regression

Lukas Fredriksson*

September 2021

Abstract

This thesis aims to investigate the possibility to perform nowcasting of the number individuals infectious with COVID-19. In order to do this a discrete time SIR (Susceptible, Infectious, Removed) with the added Diagnosed state is used to simulate epidemics. This model has three parameters, q which denotes the probability that a susceptible individual becomes infected by a given infectious individual, r which denotes the probability that an infectious individual stops being infectious and p which denotes the probability that an individual who stops being infectious become diagnosed. Four different sets of parameters where q, r and p either are fixed or uniformly randomised are used to simulate four sets of training data. These sets of data are then used to train a linear regression model where the inputs are the number of new daily diagnosed individuals of current time step and four steps before. Only diagnosed individuals are used in the linear regression because that is what we observe in reality.

The theory behind the model used for simulation show that the parameter p is not observed in the available data, while q and r are. This suggests that the cases where p is randomised during simulation should provide less accurate nowcasting, which is also suggested in linear regression. The regression model does however violate some of the assumptions made during linear regression. For this reason, further studies of nowcasting should investigate other regression models.

*Postal address: Mathematical Statistics, Stockholm University, SE-106 91, Sweden. E-mail: lrfson@gmail.com. Supervisor: Pieter Trapman.

Acknowledgments

Many thanks to Pieter Trapman for his endless support and help throughout my work with this thesis. Me being able to complete this thesis is much thanks to him. I would also like to thank Dmitry Otryakhin for his help in coming up with the subject.

Contents

1 Introduction

The coronavirus disease (COVID-19) caused by the virus SARS-Cov2 hit the world and has had a huge impact on our society. The disease is highly contagious and some of the symptoms like coughing, sore throat and headache fit in with other diseases. These symptoms come in varying degrees, mostly mild to moderate, and if they are mild should be treated at home if the person is otherwise healthy according to the World Health Organization (WHO) [10]. They also mention that it takes 5-6 days on average before symptoms show after being infected, but that it can take up to 14 days.

There are different ways to get tested for COVID-19 in Sweden, but the major way is to self-report that you are sick and then perform a test [6]. The available data for COVID-19 shows the number of confirmed infected individuals, the people who get the diagnose [7]. Since it take some time before symptoms show after getting infected and also a slight delay between taking the test and your case being added to the data set, the available data would not show the real number of infected individuals every day even if everybody self-reported immediately when they show symptoms. A study made by Folkhälsomyndigheten reports that, in Stockholm, for every confirmed case of COVID-19 there are 44.4 more cases, with a confidence interval that varies from lowest to highest with a factor of almost 3 [8]. This means that the dark numbers for confirmed cases could potentially be large.

Nowcasting is a term mainly used within economics and meteorology which describes the act of predicting withing the present or near future. This thesis aims to explore the possibilities of nowcasting the number of infectious individuals using the available data, in this case the number of diagnosed individuals. In the middle of an ongoing and deadly epidemic, delay in the data could mean that wrong decisions are made. With the help of a modified SIR (Susceptible, Infectious, Removed) model to simulate an epidemic this thesis tries to perform nowcasting on the daily number of infected individuals using a linear regression model with the number of diagnosed individuals from the current day and up to four days ago as inputs. We will see in the thesis that somewhat accurate nowcasting can be made, but that the model used for the nowcasting violates some of the assumptions made when performing linear regression. We also learn about some limitations in the currently available data

The thesis begins by explaining the theory behind the modified SIR model, followed by a section about linear regression. Then the application of the model to simulate a pandemic together with using a linear regression model to perform nowcasting are explained. Lastly the results are presented followed by a discussion.

2 Theory

In this thesis we are looking to perform nowcasting on the number of infectious individuals with the aid of linear regression. We are interested in how many

individuals are infected each day, and how it relates to our available data which is the number of diagnosed individuals. The data used for training the linear regression model comes from simulation. The model used for simulation in this thesis is a model called SIR (Susceptible, Infectious, Removed) which is widely used to describe infectious diseases. This model is described in section 2.1 and 2.2. The method used for nowcasting, linear regression, is described in section 2.3.

2.1 Continous Time SIRD epidemic

SIR (Susceptible, Infectious, Removed) is a model for infectious diseases and is explained in detail by Andersson and Britton [1]. The model has each individual in the three states susceptible, infectious and removed. In addition to these three states we have added a fourth state D, diagnosed. For this thesis the population is assumed closed and every person is treated the same, there is no difference between age groups or other factors that might increase the risk of getting infected or how long you remain infectious. The reality is more complicated than this but for the purpose of this thesis it is sufficient to make these assumptions since we are looking to explore the possibilities, not make actual nowcasting.

As previously mentioned there are four states in this model in which a person can be in at any given time. These states are susceptible, infectious, removed and diagnosed. Over time, each individual either change state or stay in their current state. The relationship between the states are shown in figure 1. A susceptible person can either stay susceptible or become infected (and therefore infectious) by the result of contact with an infectious individual.

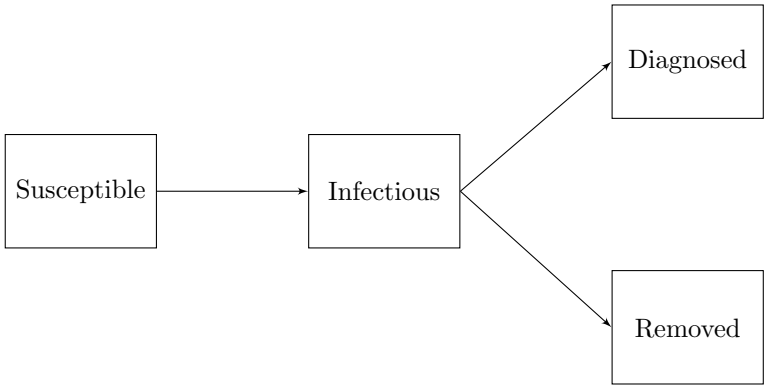

Figure 1: Flow chart of the SIRD model showing in which directions individuals transition.

From the infectious state each individual can either remain infectious, get diagnosed and enter the diagnosed state, or enter the removed state either by recovering from the infection or by deceasing. If an individual enters the removed

state it means that they are no longer infectious and can no longer infect any susceptible individuals, but not diagnosed and they never will be diagnosed. If an individual deceases and and it is found out that they had COVID-19 as cause of death, they will enter the diagnosed state. It is also assumed that a diagnosed individual is quarantined and therefore not considered infectious anymore. The removed state only includes individuals who has not been diagnosed.

Once an individual has entered the removed or diagnosed state they can not change state and will stay there until the end of the epidemic. Individuals transition between these states at different rates and the model is Markov, meaning that only the current time decides how the model behaves, not dependent on how it got there. The epidemic is considered over once the infectious state is empty. An important part of modelling epidemics is a number called the basic reproduction number (R_0) which is the expected number of transmissions of the disease made by the typical infected individual during the early stages of an epidemic which is usually between 2.5 and 5.0 [3]. A variant of the virus has evolved which have been starting to spread, the Delta variant. This Delta variant have even higher R_0, up to 60% higher [5].

We assume that initially there are n susceptible individuals and m infectious, with n being a large number and m being relatively small compared to n. Let $X(t)$, $Y(t)$ and $Z(t)$ denote the number of susceptible, infectious and diagnosed individuals at time t. As explained by Andersson and Britton [1], the time points at which an infectious individual makes contact with another individual is given by a time homogenous Poisson process with intensity $\frac{\lambda}{n}$ and if that individual is still susceptible they will now become infectious. The time that an individual stays infectious is exponentially distributed with intensity γ. This means that the rate at which new infections occur is $\frac{\lambda}{n}X(t)Y(t)$ and the rate at which people stop being infectious is $\gamma Y(t)$. The relationship between these intensities and the basic reproduction number is $R_0 = \frac{\lambda}{\gamma}$. Since we are also interested in the number of diagnosed people, let $p \in [0, 1]$ denote the fraction of no longer infectious individuals who become diagnosed. The rate at which people become diagnosed is then $\gamma Y(t)p$.

Now let $x(t)$, $y(t)$ and $z(t)$ denote the proportions of the population within each state and μ be the initial proportion of infectives. Since $X'(t)$ is the rate of change for the number of susceptible individuals, which is the same as the rate at which new infections occur, we get $X'(t) = -\frac{\lambda}{n}X(t)Y(t)$, since the newly infected leaves the susceptible state. With $x(t) = \frac{X(t)}{n}$ we get $\frac{X'(t)}{n} = -\frac{1}{n}\frac{\lambda}{n}X(t)Y(t)$ which gives $x'(t) = -\lambda x(t)y(t)$. Whenever an individual leaves the susceptible state they enter the infectious state and when an individual stops being infectious they leave the state. This, with the previous calculations leads to $y'(t) = \lambda x(t)y(t) - \gamma y(t)$. And since only a fraction p of the individuals leaving the infectious state enters the diagnosed state we get $z'(t) = \gamma p y(t)$.

This gives a set of differential equations

$$\begin{aligned} x'(t) &= -\lambda x(t)y(t), \quad x(0) = 1, \\ y'(t) &= \lambda x(t)y(t) - \gamma y(t), \quad y(0) = \mu \\ z'(t) &= \gamma p y(t), \quad z(0) = 0 \end{aligned} \tag{1}$$

which are the rates of transition for individuals between the states.

2.2 Discrete Time SIRD epidemic

The available data for the number of diagnosed individuals is not in continuous time, the reports come in at discrete time. For this reason the simulations are made with a discrete time Markov chain. The chain being Markov means that the conditional probabilities for the state of the chain in the next time step only depends on the current state of the chain, not how it got there. First let X_j, Y_j and Z_j denote the number of susceptible, infectious and diagnosed individuals at time step j. Further let

$$\begin{aligned} A_{j+1} &= X_j - X_{j+1} \\ B_{j+1} &= Z_{j+1} - Z_j \\ C_{j+1} &= X_j - X_{j+1} + Y_j - Y_{j+1} + Z_j - Z_{j+1} \end{aligned} \tag{2}$$

denote the number of individuals who transition from susceptible to infectious, infectious to diagnosed and infectious to removed respectively. A_{j+1} only depends on X_j and Y_j and is independent of B_{j+1} and C_{j+1} while B_{j+1} and C_{j+1} are not independent of each other and further only depends on Y_j. The conditional probabilities for the model are

$$\begin{aligned} &P(A_{j+1} = a_{j+1}, B_{j+1} = b_{j+1}, C_{j+1} = c_{j+1} | X_j = x_j, Y_j = y_j, Z_j = z_j) \\ &\qquad = \binom{x_j}{a_{j+1}} (1 - q^{y_j})^{a_{j+1}} q^{y_j(x_j - a_{j+1})} \\ &\frac{y_j!}{b_{j+1}! c_{j+1}! (y_j - (c_{j+1} + b_{j+1}))!} (1 - r)^{y_j - (c_{j+1} + b_{j+1})} (rp)^{b_{j+1}} (r(1-p))^{c_{j+1}}. \end{aligned} \tag{3}$$

There are three parameters in this model which are the probabilities of a person transitioning to a different state between time j to time $j + 1$. The first one, q, is the probability that a susceptible individual avoids being infected by an infectious individual. Since there are $Y_j = y_j$ infectious individuals at time j the susceptible one has to avoid being infected by y_j people and the probability for this is q^{y_j} which means that the probability to become infected is $1 - q^{y_j}$. The second one, r, is the probability that an infectious individual stops being infectious and the probability to remain infectious is $1 - r$. Lastly p denotes the probability that an individual no longer infectious is diagnosed. This means that the probability that an infectious individual becomes diagnosed is rp and that an infectious individual has stops being infectious without being diagnosed is $r(1 - p)$.

The time an individual spends being infectious in the continuous case is exponentially distributed with intensity γ as discussed in the previous section 2.1. This means that the expected time until an individual leaves the infectious state is $\frac{1}{\gamma}$, while it is $\frac{1}{r}$ in the discrete case. We want to choose parameters q and r of model (3) to make a good approximation of the continuous model and therefore choose $r = \gamma$. We also have that the probability of a contact occurring between two given individuals in the continuous case is $1 - e^{\frac{-\lambda}{n}}$ [1] which can be approximated by $\frac{\lambda}{n}$. This probability in the discrete case is $1 - q$, and to approximate the continuous model as good as possible we choose $q = 1 - \frac{\lambda}{n}$. Looking back at equation (1) we have $\frac{y'(t)}{y(t)} = \lambda x(t) - \gamma$. At the beginning of the epidemic $x(t) \approx 1$ which gives $y(t) = y(0)e^{(\lambda-\gamma)t}$ and towards the end we have $x(t) \approx x(\infty)$ which gives $y(t) = y(0)e^{(\lambda x(\infty)-\gamma)t}$ where $x(\infty)$ is the fraction of individuals still susceptible after the epidemic has ended. Let α denote the exponential growth rate of $y(t)$ and use the discrete notations $\lambda = (1-q)n$ and $\gamma = r$ to get two different α,

$$\begin{aligned} \alpha_1 &= (1-q)n - r, \\ \alpha_2 &= (1-q)nx(\infty) - r. \end{aligned} \tag{4}$$

This approximately gives growth rate α_1 for the first part of the epidemic and rate α_2 for the second part. Note that two of three parameters from model (3), q and r, appear in equation (4), while parameter p does not.

The fraction of individuals still susceptible after the epidemic has ended, $x(\infty)$, can be derived using a method described by Britton [4]. We assume that as n goes to infinity, $x(\infty)$ converges and thus is not random. This fraction is then approximately equal to the probability of not getting infected, which is approximately equal to escaping infection from each infectious individual through the epidemic. This is equal to $e^{-R_0(1-x(\infty))}$ where $1-x(\infty)$ is the fraction of all individuals who got infected during the epidemic. We then have the final size equation

$$x(\infty) = e^{-R_0(1-x(\infty))}.$$

Multiplying by $R_0 e^{-x(\infty)R_0}$ gives

$$x(\infty)R_0 e^{-x(\infty)R_0} = R_0 e^{-R_0}$$

which are two realisations of $f(x) = xe^{-x}$. The maximum of $f(x)$ is in $x = 1$ which is the only local extreme point, and since $f(R_0)$ and $f(x(\infty)R_0)$ has the same value with $R_0 \neq x(\infty)R_0$, one of the inputs has to be larger than 1 and the other has to be less than 1. We know that $R_0 > 1$, and thus $x(\infty)R_0 < 1$. Since $R_0 = \frac{\gamma}{\lambda}$ with $\gamma = (1-q)n$ and $\lambda = r$ we can use $r = \frac{(1-q)n}{R_0}$ to rewrite α_2 in (2.2) as

$$R_0\alpha_2 = (1-q)n[x(\infty)R_0 - 1].$$

We previously showed that $x(\infty)R_0 < 1$, meaning that the right hand side of the equation above is negative. On the left hand, we know that R_0 is positive, and thus α_2 has to be negative. This means that the number of new daily infected individuals decreases in the second part of the epidemic.

2.3 Linear regression

Regression is a method to predict values of a target variable given some input values. Linear regression in its simplest form is a linear function of the input values. Let t be our target value and x be a vector of explanatory input values that corresponds to this target value. We can say that we try to model the distribution of $p(t|\boldsymbol{x})$, that is the probability of t given the input vector $\boldsymbol{x}$ [2]. When we have one input value x and one output value our linear regression model is $y(x) = \beta_0 + \beta_1 x$ where β_k are the model parameters which are estimated using a maximum likelihood method. With this, $y(x)$ is a deterministic function and $t = y(x) + \epsilon$ where ϵ is a Gaussian random variable with mean zero.

Using only a linear function of the input variables has limitations on the model, and we can instead use nonlinear transformations of the input variables and combine them linearly [2]. For example, with one input variable we could have the model $y(x) = \beta_0 + \beta_1 log(x)$. More generalised we have $y(\boldsymbol{x}, \boldsymbol{\beta}) = \beta_0 + \sum_{k=1}^{N-1} \beta_k \phi_k(\boldsymbol{x})$, which is a model with N parameters and $\boldsymbol{x}$ as the input vector and $\boldsymbol{\beta}$ as the parameter vector, with $\phi_k(\boldsymbol{x})$ possibly nonlinear as long as $y(\boldsymbol{x}, \boldsymbol{\beta})$ is a linear function of $\boldsymbol{\beta}$ [2]. An important part for investigating the validity of a linear regression model is the residuals, which is the distance between the observed value and the regression line.

When performing linear regression there are some assumptions that are made which are explained by Knief and Forstmeier [9]. These are validity, independence, linearity, homoscedasticity of the errors and normality of the errors. Validity means that the regression should help answer the scientific question at hand. Independence means that the explanatory input values should be independent. Linearity means that the output variable should be linearly dependent on the input values through the parameters $\boldsymbol{\beta}$. Homoscedasticity means that the errors, ϵ, should be constant across all $\boldsymbol{x}$ and lastly normality means that the errors should be normally distributed. Homoscedasticity and normality of the errors are both checked by looking at the residuals. In the case of normality a Q-Q plot is often used, which plots the quantiles of the distribution of the residuals against the quantiles of a normal distribution.

Later in the thesis in section 3.1 we will see that the the logarithm of daily new infectious and diagnosed individuals is somewhat linear in the first (and the second) stage of the epidemic, and thus t is the logarithm of daily new infectious individuals. Every time step has a pair of daily number of infected and diagnosed individuals, but in order to have more information to work with the number of daily diagnosed individuals four days leading up to that time are also taken into account. This means that $\boldsymbol{x}$ is a vector with five elements. Since we are considering the logarithm of these, we have that $\phi_k(\boldsymbol{x})$ is the logarithm of the k-th element of $\boldsymbol{x}$.

3 Simulations and application of linear regression

This section of the thesis describes how the theory of section 2.2 can be used to perform simulations. The simulations generate the training- and test data required to estimate the parameters of the linear regression model used to perform nowcasting. Section 3.1 describes how the simulations are performed and section 3.2 describes how the simulated data is used to estimate the parameters of the model. All code is written in R and can be found in the appendix, section 6.4.

3.1 Simulations

The model (3) allows for a simulation where at each time step the number of individuals in each of the four states S, I, R and D are recorded. To get the number of individuals who transitioned from susceptible to infectious between time steps j and $j+1$ a binomial random function with the n parameter set to the number of susceptible individuals at time step time step j (X_j) and the probability parameter set to $1-q^{Y_j}$. This gives A_{j+1} in equation (2). With the help of a multinomial function with n set to the number of infectious individuals at time step j (Y_j) and the probabilities set to $1-r$, rp and $r(1-p)$ as discussed in section 2.1, we get B_{j+1} and C_{j+1} from equation (2). These functions use the first factor and the second factor respectively in model (3). These makes it trivial to get the number of susceptible, infectious and diagnosed individuals in the next time step.

This results in a matrix where each row contained the current time step, the total number of individuals in each state and the number of people who became infectious and diagnosed during that time step. Let $i(t)$ and $d(t)$ denote how many individuals transitioned to infectious and diagnosed respectively at time t. To investigate the relationship between the number of diagnosed and infectious individuals two new columns were created in the matrix, one with $\frac{i(t+1)}{i(t)}$ and one with $\frac{d(t+s)}{d(t)}$. The purpose for these columns is to compare the growth rate of these states.

As mentioned in section 2.1, R_0 is an important number when modeling infectious diseases. This number is important to bear in mind when choosing the parameters in the model, q and r in particular. They should result in R_0 being between 2.5 and 5.0 as mentioned in section 2.1. To accomplish this r was chosen to be $\frac{1}{4}$ meaning that an individual is infectious on average four days. Since q is the probability that an individual avoids getting infected by a given infectious individual, $1-q$ is the probability that they do get infected. We want an infectious person to infect one person per day on average to achieve $R_0 \approx 4$ and set $1-q = \frac{1}{X_0}$. This will result in an average of one infectious contact per day which is what we wanted. The parameter p is harder to estimate. There is no R_0 or similar to use to estimate it and therefore it has to be guessed. For this first simulation $p = \frac{1}{3}$ is used.

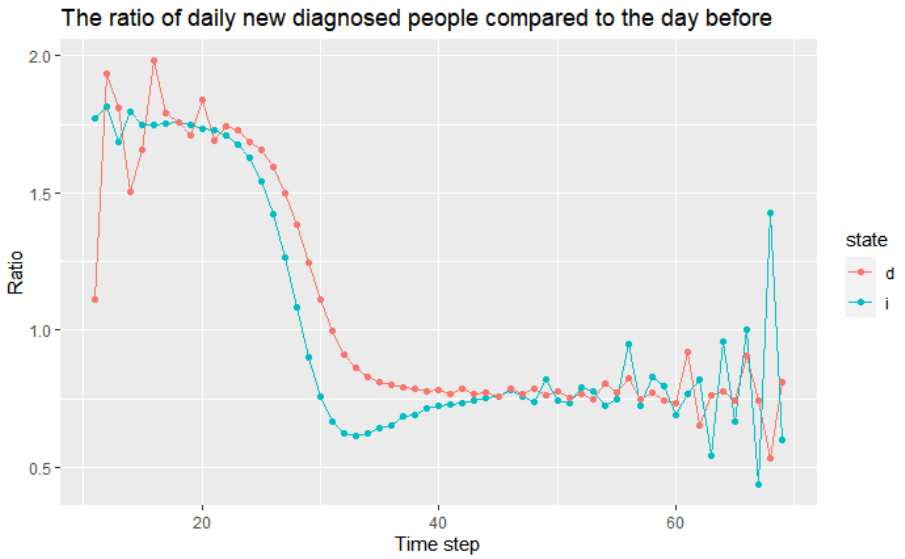

Figure 2: The ratio between daily new individuals of current time step and the time step before of both infected and diagnosed individuals.

At first one epidemic was simulated using these parameters. Figure 2 plots the ratio between daily new individuals of the current time step and the time step before of both diagnosed and infected individuals against t which shows that after the growth has been stabilised the growth rate of infectious and diagnosed individuals are similar. Then, since the growth rate of the number of infectious individuals is exponential the pool of susceptible individuals quickly depletes which in turn means that the growth of infectives slows down. There is a delay before the growth of diagnosed individuals slows down after, and then they converge to a similar rate.

This similarity between the growth rate of the two states is further shown in figure 3. It shows the logarithm of $d(t)$ and $i(t)$ against t and as in figure 2 there is a delay between the two, but they both have a similar growth rate after the epidemic has started spreading. After a while the pool of susceptibles starts to deplete and there is a decline in the growth. Looking back at equation (4) we can see that during the first stage of the epidemic the number of infectious has a growth rate of α_1 and during the second stage the growth rate is α_2 with the second rate being negative.

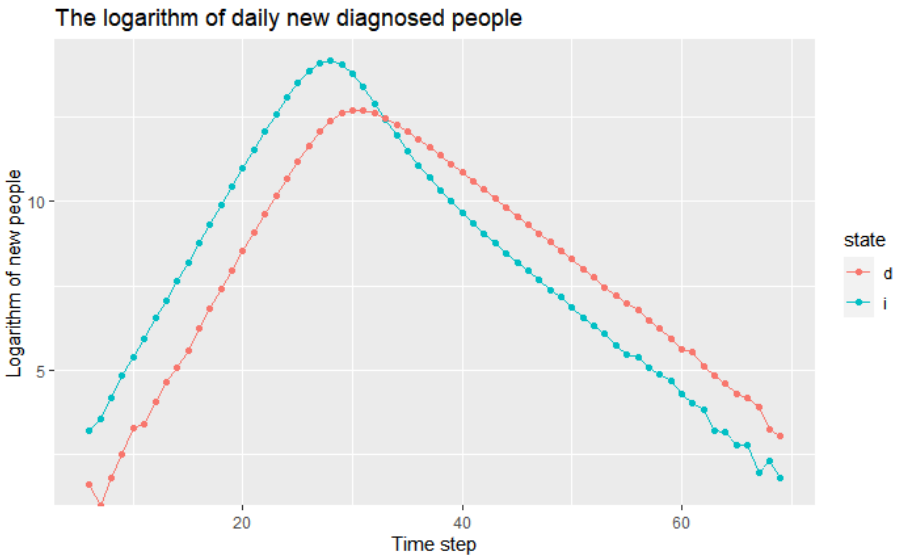

Figure 3: The logarithm of daily new infectious and diagnosed individuals.

3.2 Application of linear regression

As we saw in figure 3 the growth rate of the number of diagnosed and infectious individuals stays roughly equal throughout the epidemic. For now we will focus on the first stage of the epidemic, but the same methods can be used on the second stage as well. The nowcast requires that enough time has elapsed for us to be able to estimate α_1 in equation (4) with the help of linear regression. To get more training data 1001 simulations were made using the discrete time model (3) with different sets of parameters. Since equation (4) includes both r and q, but not p, our available data do not contain any information about p. For this reason the sets of parameters are chosen particularly to test if nowcasting is possible with, or without p. Four sets of parameters are used to simulate four sets of training data. The first two sets has q and r as in the previous section 3.1 with one of them having uniformly randomised p between $(\frac{1}{5}, \frac{1}{2})$. We keep q and r fixed in these sets with p fixed in one and randomised in the other in order to investigate if a variation in p makes a significant difference in the linear regression. The other two sets has r uniformly randomised between $(\frac{1}{6}, \frac{1}{3})$ and q uniformly randomised between $(1 - \frac{2}{3n}, 1 - \frac{1}{2n})$ (n being the number of initially susceptible), with p as in the first two sets. These sets of parameters are chosen to further investigate whether a variation in p has a bigger impact on the regression than a variation than q and r. Whenever a randomised parameter is used a new set of parameters was generated for each simulation.

These data sets are used to perform linear regression as described in section 2.3. Let i_j denote the number of new infectious individuals and d_j the number

of new diagnosed individuals at time j. The linear regression model used is

$$log(i_j) = \beta_0 + \beta_1 log(d_j) + \beta_2 log(d_{j-1}) + \beta_3 log(d_{j-2}) + \beta_4 log(d_{j-3}) + \beta_5 log(d_{j-4}) \tag{5}$$

where β_k are the parameters of the model. Since the growth fluctuates at the beginning of the epidemic the early values are filtered out. After a while when the growth starts to fall off since the pool of susceptible individuals is being depleted, the logarithm of the growth rate is no longer linear and therefore these values are also filtered out. This model is then used to predict the number of daily new infectious individuals given another simulated epidemic with the same set of parameters and the predicted values are compared to the actual test values.

4 Results

The method described in section 3.2 results in plots of the comparison between the predicted values and the actual test values for each set of parameters used when simulating the data. There were also model parameter estimates, Q-Q plots and pair plots.

The Q-Q plot all show that the residuals are non-normal except for the case of all parameters being randomised which can be seen in figure 4 shows the case when all parameters in the simulations are fixed. If the residuals are normal the dots should align to the line in the plot, and in this case the dots are reasonable aligned to the line. This means that for the other three cases the residuals are not normal and did not fulfill that assumption described in section 3.2. Plots for the other cases can be found in appendix section 6.1.

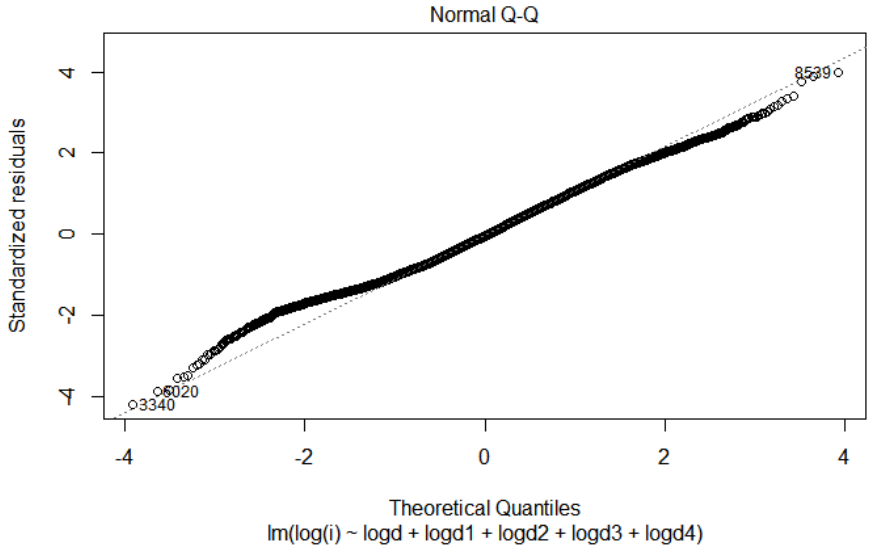

Figure 4: Q-Q plot for the linear regression model when all parameters are randomised.

There is also a clear evidence of dependence between the variables which can be shown in figure 5, which is the case where all parameters are fixed during simulation. The figure plots every variable in the data set used to train the linear regression model against each other in pairs and it shows that there are strong correlations between each of the explanatory variables meaning that they depend on each other. This violates the assumption of dependency described in section 3.2. The rest of the pair plots can be found in appendix section 6.3 and show more variance, but still strong correlation between the variables.

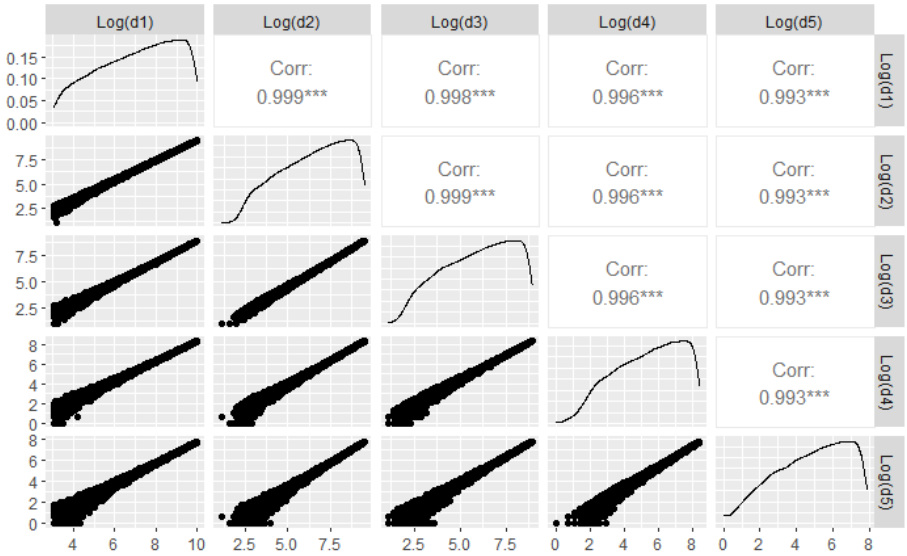

Figure 5: Pair plot for the linear regression model when all parameters are fixed.

The last assumption that will be checked is the one about homoscedasticity. Figure 6 shows a plot of the residuals over the predicted values of the model where all parameters are fixed. We can see that the residuals do not have constant variance, as it is larger in the beginning but become smaller after a while. This means that the model with all parameters fixed during simulation violates this assumption. However, the variance become more even the more randomised parameters there are in the model. For the model with all three parameters randomised the variance seems to be closer to constant, as seen in figure 7. There are some outliers however, and they mostly appear at the beginning of a simulated epidemic. The other plots can be found in appendix section 6.2.

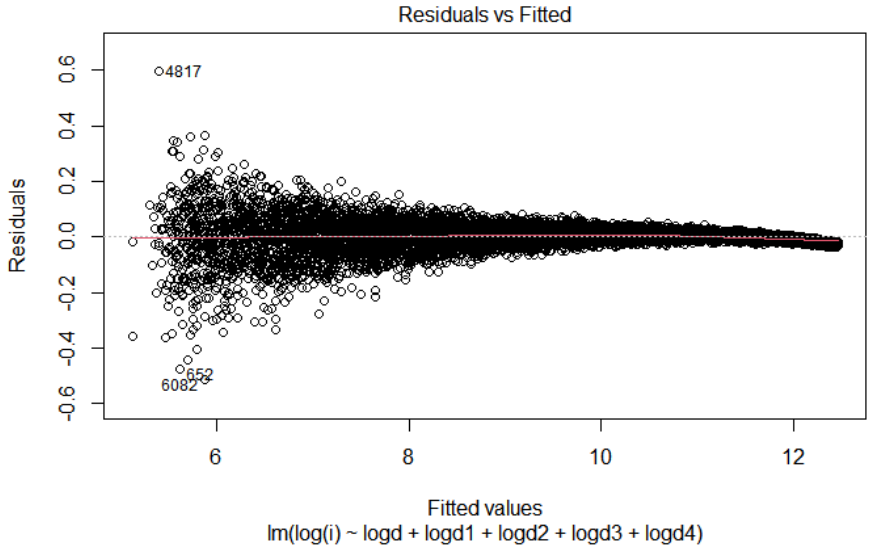

Figure 6: Residual plot for the linear regression model when q and r are fixed, p randomised.

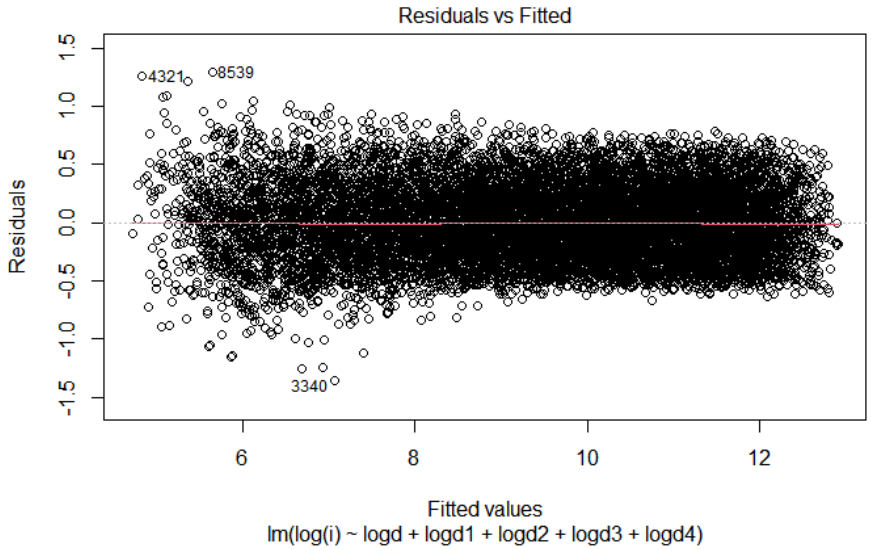

Figure 7: Residual plot for the linear regression model when all parameters are randomised.

When it comes to the actual predictions we first look at the cases where q

and r are fixed during simulation. Figure 8 shows a comparison between the cases with the left plot having p fixed and the right plot having p randomised during simulation. The figure plots the actual number of daily new infectious individuals against the predicted number of individuals with a line indicating them being equal. In table 1 we can see the estimated parameters of model (5) when q and r are fixed during simulation.

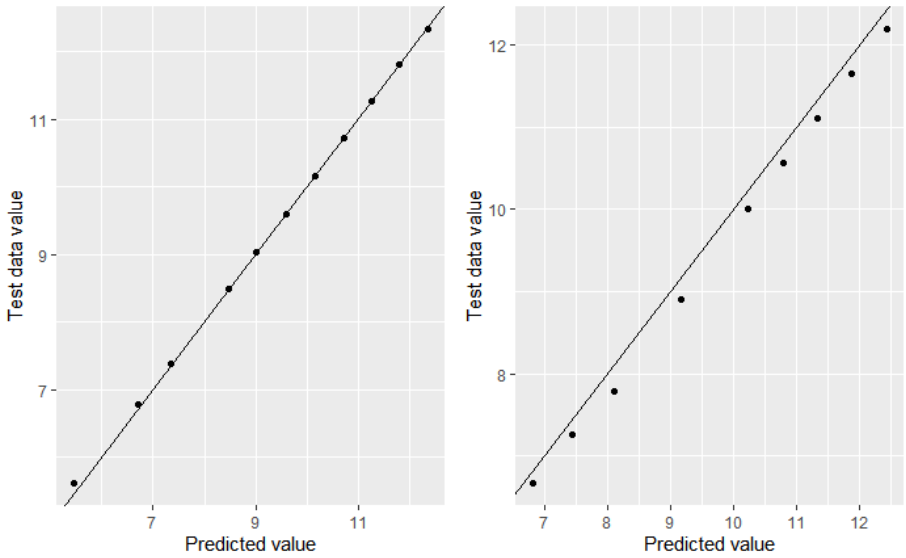

Figure 8: Plots showing a comparison between the predicted number of infectious and actual number of infectious individuals when q and r are fixed during simulation. Left plot has p fixed and right plot has p randomised

Model parameters with q and r fixed		
Parameter	p fixed	p randomised
β_0	2.97	3.05
β_1	0.53	0.48
β_2	0.26	0.25
β_3	0.12	0.13
β_4	0.06	0.09
β_5	0.03	0.04

Table 1: Table showing the estimated parameters of the linear regression models when q and r are fixed.

For the cases where q and r are both randomised during simulation we have figure 9 comparing the predicted and actual values and like previously, table 2 lists the estimated parameters of model (5), but with q and r being randomised.

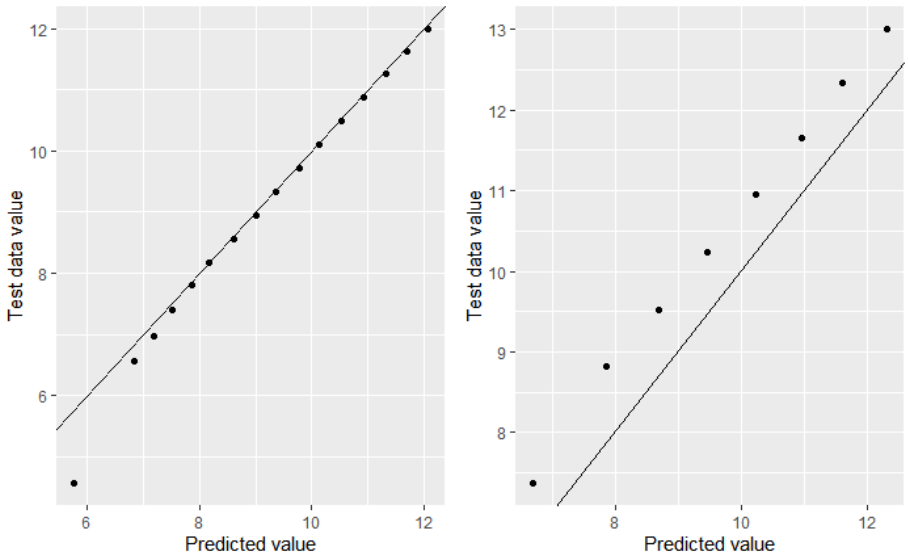

Figure 9: Plots showing a comparison between the predicted number of infectious and actual number of infectious individuals when q and r are randomised. Left plot has p fixed and right plot has p randomised

Model parameters with q and r randomised		
Parameter	p fixed	p randomised
β_0	1.38	1.35
β_1	1.44	1.40
β_2	0.33	0.41
β_3	-0.28	-0.28
β_4	-0.24	-0.30
β_5	-0.24	-0.23

Table 2: Table showing the estimated parameters of the linear regression models when q and r are randomised.

Looking at figure 8 and 9 we can see that the cases where p of model (3) is fixed seems to give better predictive results than the cases where p is randomised.

5 Discussion and Conclusion

Having residuals that are not normal as shown in the appendix section 6.1 could mean that there are information in the data that the linear regression model is not taking into account for. This is something that, if there were to be further work with this method of nowcasting the number of infectious individuals, would

have to be looked into. Taking another look at 6 and investigating the outliers by running the simulations found in appendix 6.4 and looking at the data shows that they mostly appear in the beginning of an epidemic. Looking at the other cases with different model parameters show that they also mostly appear in the beginning. This could be because the beginning of the simulated epidemics are volatile and tweaking the filtering done in section 3.2 could improve the predictions.

The model used for the simulations is also simplified. In reality there is an incubation time and there are different groups of people. But for now we make the assumption that this is how the reality looks and we interpret the results. The data available in Sweden, which is the number of diagnosed individuals, allows us to observe (4). Under the assumption that the model used in the thesis holds, we can only observe q and r, while p remains unobserved. Knowing this, we can expect that the biggest problem with the nowcasting would be dealing with p, as p gives the relationship between the number of diagnosed (observed) individuals and the actual number of infected. From figure 8 and 9 we can see that the cases where p is fixed during simulation seems to give better predictive results than the cases where p is randomised. However, figure 4 shows the residuals of the model where all parameters are randomised are close to normally distributed and 7 shows variance in the residuals that is much closer to constant than 6.

Multiple of the assumptions made in linear regression as described in section 3.2 are violated which means that the results provided by model (5) could potentially be misleading. The main issue is that the input variables of the regression model are very correlated, as shown in figure 5. Further work with the model could involve the removal of some of variables, possibly using only the current time as input instead of the series of time steps leading up to the current. This might however not improve the non normal distribution of the residuals, meaning that it could be useful to add more complexity to the model or trying another method of regression. Looking at the theory of section 2 indicates that the problem of not being able to observe parameter p, the probability that an infected individual gets diagnosed, has to be dealt with to be able to perform nowcasting. This is also somewhat reflected in 8 and 9 where we can see that having p randomised rather than fixed during simulations has a negative impact on the predictions.

As a summary of the thesis we can say that the model used in this thesis violates multiple of the assumptions made when performing linear regression, and the results in section 4 might not be reliable. However, we showed that the available data only observes parameters q and r of model (3), and not p, and that this is a potential problem when performing nowcasting of infectious individuals when we can only observe the number of diagnosed individuals.

6 Appendix

6.1 Q-Q plots

Here are the rest of the Q-Q plots from section 4 presented. Figure 10 shows the Q-Q plot of the case when parameters q and r are fixed while p is randomised during simulation. Figure 11 shows Q-Q plot of the case where q and r are randomised and p fixed during simulation. Lastly figure 12 shows the Q-Q plot of the case when all parameters are fixed during simulation.

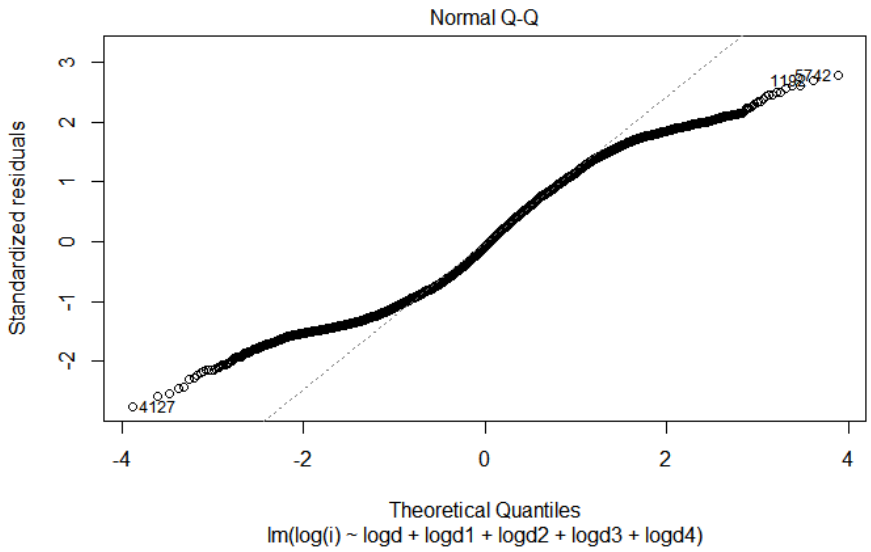

Figure 10: Q-Q plot for the linear regression model when q and r are fixed, p randomised.

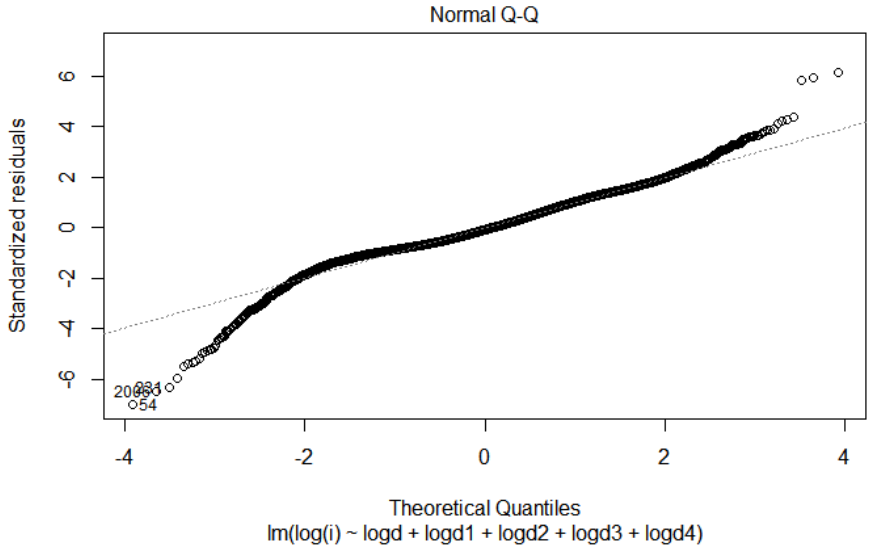

Figure 11: Q-Q plot for the linear regression model when q and r are randomised, p fixed.

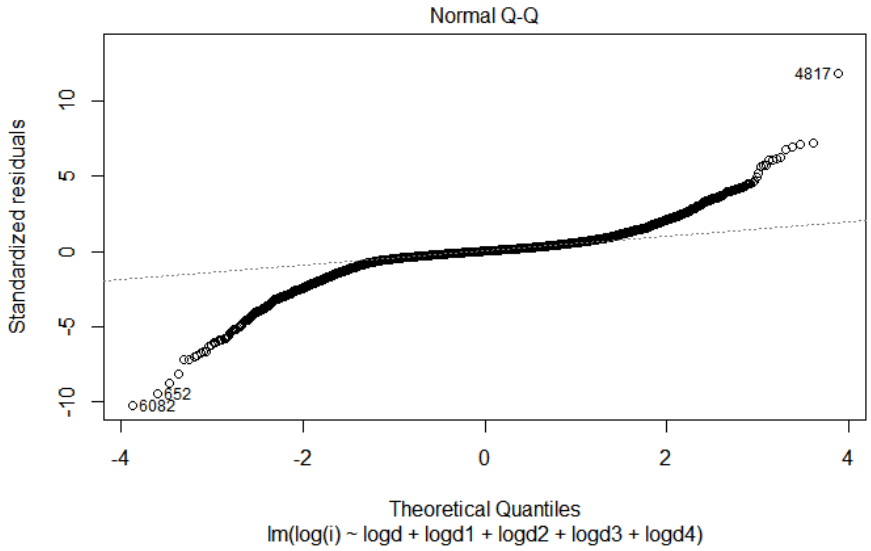

Figure 12: Q-Q plot for the linear regression model when all parameters in the simulation are fixed.

6.2 Residual plots

These are the residual plots from section 4. Figure 13 shows the residuals resulting from the regression model when q and r are randomised and p fixed during simulation, and 14 shows the residual plot from when q and r are fixed while p is randomised during simulation.

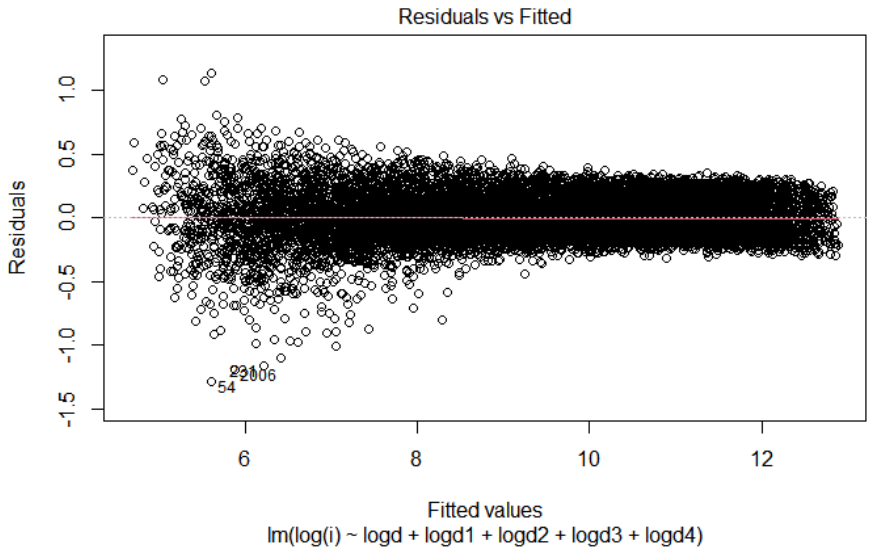

Figure 13: Residual plot for the linear regression model when q and r are randomised, p fixed.

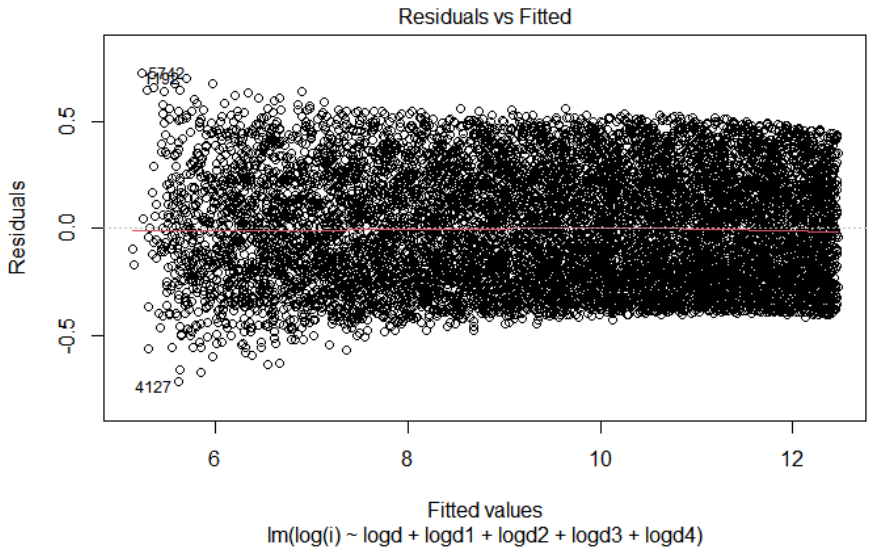

Figure 14: Residual plot for the linear regression model when q and r are fixed, p randomised.

6.3 Pair plots

The plots in this section are the pair plots left out of section 4. Figure 15 is the plot from the case when q and r are kept fixed and p is randomised during simulation, figure 16 from the case when q and r are randomised while p is kept fixed, and figure 17 is from the case when all parameters are randomised.

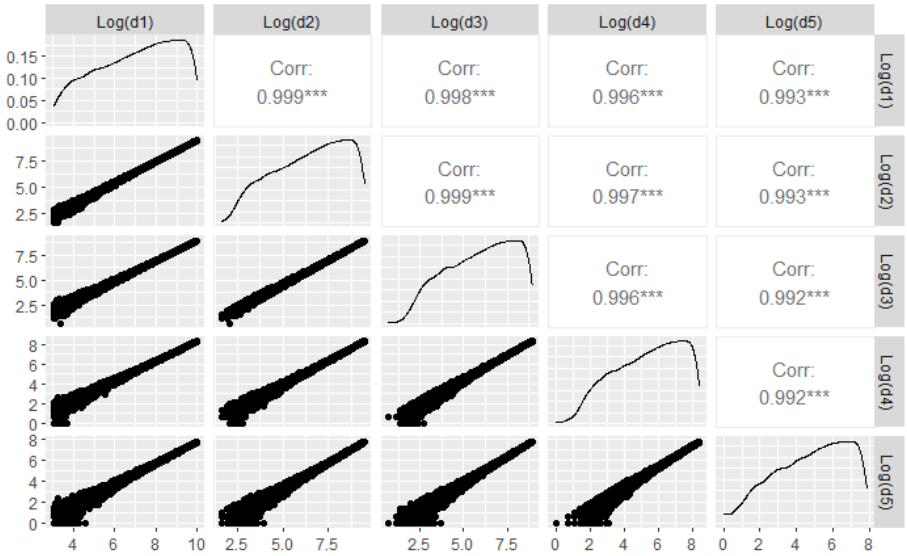

Figure 15: Pair plot for the linear regression model when q and r are fixed, p randomised.

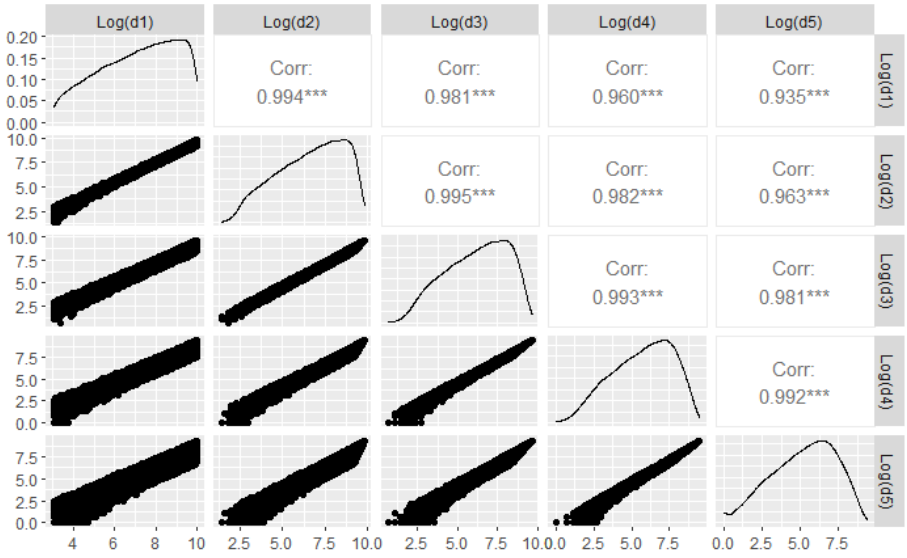

Figure 16: Pair plot for the linear regression model when q and r are randomised, p fixed.

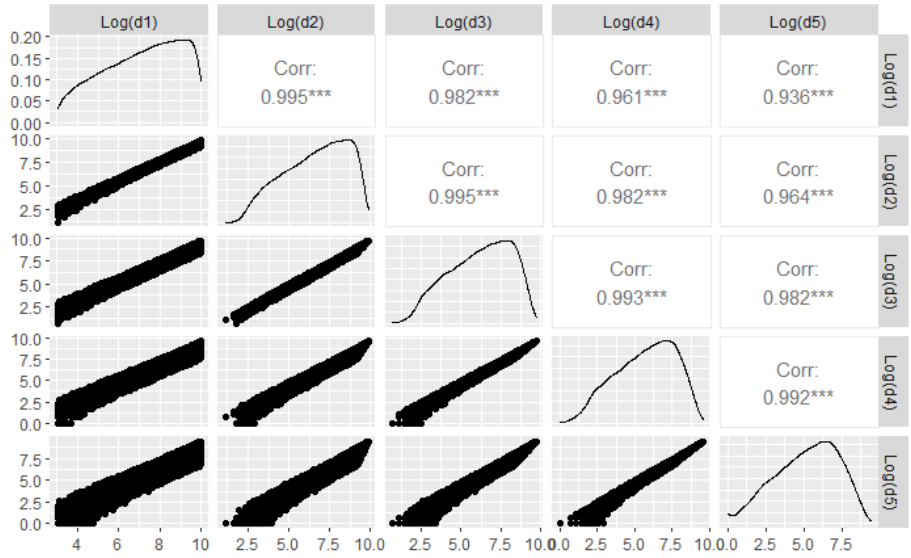

Figure 17: Pair plot for the linear regression model when all parameters are randomised.

6.4 R code

```
# Library imports and global variables
#-----------------------------------
# tidyverse used to manipulate data frames
# and generate plots
library(tidyverse)

# Additional packages GGally and gridExtra
# has to be installed to generate plots

# Seed for randomisation
seed <- 123456
#-----------------------------------


# Functions
#-----------------------------------

# This function performs a single simulated epidemic.
# The default arguments are as described in the thesis
simulate_epidemic <- function (N = 10000000,
                               q = exp(-1/(N-1)),
                               r = 1/4,
```

```
                                p = 1/3) {
  valid_epidemic <- FALSE
  while (!valid_epidemic) {
    # This data frame keeps track of each state in
    # each time step of the epidemic.
    df <- data.frame(t = 0,
                     S = N-1,
                     I = 1,
                     R = 0,
                     D = 0,
                     d = 0,
                     i = 0)
    k <- 1
    # Loop while there are still infectious individuals
    while (df$I[k] > 0) {
      # How many susceptible individuals become infected
      # with parameters as described in the thesis
      a <- rbinom(1, df$S[k], (1-q^df$I[k]))

      # How many currently infectious individuals
      # stops being infectious, and how many of these gets diagnosed.
      # Parameters described in the thesis
      multi <- rmultinom(1, df$I[k], c(1-r, r*p, r*(1-p)))

      # Add next time step to the data frame using equation (2)
      # in the thesis to calculate.
      df[k+1,] <- c(k,
                    df$S[k]-a,
                    a+multi[1],
                    df$R[k]+multi[3],
                    df$D[k]+multi[2],
                    multi[2],
                    a)
      k <- k+1
    }
    # If the epidemic is too short it is not considered valid
    # because we only want to consider major outbreaks
    if (nrow(df) > 20) {
      valid_epidemic <- TRUE
    }
  }
  return (df)
}

# This function takes a data frame consisting of epidemic data and
# returns a data frame consisting of the ratio of change of
```

```r
# infectious and diagnosed individuals between each time step.
change_ratio <- function (df) {
  result <- data.frame(t = 0,
                       dS = 0,
                       dI = 0,
                       dR = 0,
                       dD = 0,
                       d = 0,
                       i =0)

  # Loops through the data and calculates the change ratio
  # defaulting to 0 if the denominator is 0
  for(k in 1:(nrow(df)-1)) {
    ds <- ifelse(df$S[k] != 0,
                 df$S[k+1] / df$S[k],
                 0)
    di <- ifelse(df$I[k] != 0,
                 df$I[k+1] / df$I[k],
                 0)
    dr <- ifelse(df$R[k] != 0,
                 df$R[k+1] / df$R[k],
                 0)
    dd <- ifelse(df$D[k] != 0,
                 df$D[k+1] / df$D[k],
                 0)
    d <- ifelse(df$d[k] != 0,
                 df$d[k+1] / df$d[k],
                 0)
    i <- ifelse(df$i[k] != 0,
                 df$i[k+1] / df$i[k],
                 0)
    result[k,] <- c(k,ds,di,dr,dd,d,i)
  }
  return (result)
}

# This function reformats a data frame consisting of
# epidemic data to better suit the linear regression
previous_days <- function (df) {
  df <- df %>%
    mutate(logd = log(d)) %>%
    select(t, i, d, logd) %>%
    filter(d != 0) %>%
    # Adds value from previous days to each time step
    # where d1 is one time day ago
    mutate(d1 = lag(d, n=1, default = 0),
```

```
            d2 = lag(d, n=2, default = 0),
            d3 = lag(d, n=3, default = 0),
            d4 = lag(d, n=4, default = 0),
            logd1 = lag(log(d), n=1, default = 0),
            logd2 = lag(log(d), n=2, default = 0),
            logd3 = lag(log(d), n=3, default = 0),
            logd4 = lag(log(d), n=4, default = 0)) %>%
    # This filters away the non increasing parts of the daya
    # to make sure we keep ourselves within the first part of the epidemic
    filter(d >= d1 &
             d1 >= d2 &
             d2 >= d3 &
             d3 >= d4 &
             # To make sure we are in the linear part, filter away
             # parts where the slope changes too much
             d/d1 > 0.9*d1/d2 &
             logd > 3 &
             logd < 10)
  return (df)
}

# Help function to generate new set of parameters as described in section 3.2
# If p, q or r set to fixed, return the fixed probability. Otherwise randomise
# between values described in the thesis.
random_probs <- function(q_fixed = FALSE, r_fixed = FALSE, p_fixed = FALSE) {
  p <- ifelse(p_fixed, 1/3, runif(1, 1/5, 1/2))
  q <- ifelse(q_fixed,
           exp(-1/9999999),
           runif(1, exp(-1.5/9999999), exp(-0.5/9999999)))
  r <- ifelse(r_fixed, 1/4, runif(1, 1/6, 1/3))

  return (c(p = p, q = q, r = r))
}

# Perform 1000 simulations using the chosen set of fixed or randomised parameters
sim_many_times <- function(q_fixed = TRUE,
                           r_fixed = TRUE,
                           p_fixed = TRUE) {
  # Generate one set of parameters, perform one simulation and
  # reformat data for regression
  probs <- random_probs(q_fixed, r_fixed, p_fixed)
  sims <- simulate_epidemic(q = probs["q"],
                           r = probs["r"],
                           p = probs["p"])
  sims <- previous_days(sims)
```

```r
  # Then do it 1000 more times
  for (i in 1:1000) {
    probs <- random_probs(q_fixed, r_fixed, p_fixed)

    sim <- simulate_epidemic(q = probs["q"],
                             r = probs["r"],
                             p = probs["p"])
    sim <- previous_days(sim)
    sims <- rbind(sims, sim)
  }
  return (sims)
}
#-----------------------------------


# Simulations
#-----------------------------------
# First we set the seed
set.seed(seed)

# Run simulation for figure 2 and figure 3
test <- simulate_epidemic()
ratio <- change_ratio(test)


# Run 1001 simulations using the default parameters
# which are the fixed probabilities
sims <- sim_many_times()

# Simulate one epidemic as test data
test_data <- simulate_epidemic()
test_data <- previous_days(test_data)

# Filter away where i = 0 since we take the logarithm of i and dont want errors
sims <- sims %>%
  filter(i != 0)
# Fit the regression model
model <- lm(log(i) ~ logd + logd1 + logd2 + logd3 + logd4,
            data = sims)

# Get predicted values using the model and simulated test data
preds <- predict(model, test_data)

# The procedure above is then performed for the different
# sets of parameters explained in section 3.2 of the thesis.
```

```r
# Run the above procedure but with p fixed, q and r randomised
# as described in section 3.2
# with the added step of generating probabilities.
sim_known_p <- sim_many_times(r_fixed = FALSE, q_fixed = FALSE)

# Generate probabilities when p is kept fixed
probs <- random_probs(p_fixed = TRUE)

test_known_p <- simulate_epidemic(q = probs["q"],
                          r = probs["r"],
                          p = probs["p"])
test_known_p <- previous_days(test_known_p)

sim_known_p <- sim_known_p %>%
  filter(i != 0)
known_p_model <- lm(log(i) ~ logd + logd1 + logd2 + logd3 + logd4,
                    data = sim_known_p)

known_p_pred <- predict(known_p_model, test_known_p)


# Same procedure but with q and r fixed, p randomised
# as described in section 3.2
sim_unknown_p <- sim_many_times(p_fixed = FALSE)

probs <- random_probs(q_fixed = TRUE, r_fixed = TRUE)

test_unknown_p <- simulate_epidemic(q = probs["q"],
                          r = probs["r"],
                          p = probs["p"])
test_unknown_p <- previous_days(test_unknown_p)

sim_unknown_p <- sim_unknown_p %>%
  filter(i != 0)
unknown_p_model <- lm(log(i) ~ logd + logd1 + logd2 + logd3 + logd4,
                      data = sim_unknown_p)

unknown_p_pred <- predict(unknown_p_model, test_unknown_p)


# Once again same procedure but every parameter
# randomised as described in section 3.2
sim_unknown_all <- sim_many_times(q_fixed = FALSE,
                                  r_fixed = FALSE,
                                  p_fixed = FALSE)
```

```r
probs <- random_probs()

test_unknown_all <- simulate_epidemic(q = probs["q"],
                                      r = probs["r"],
                                      p = probs["p"])
test_unknown_all <- previous_days(test_unknown_all)

sim_unknown_all <- sim_unknown_all %>%
  filter(i != 0)
unknown_all_model <- lm(log(i) ~ logd + logd1 + logd2 + logd3 + logd4,
                        data = sim_unknown_all)

unknown_all_pred <- predict(unknown_all_model, test_unknown_all)
#-----------------------------------

# Plotting
#-----------------------------------

# Plot figure 2
ratio %>%
  # First filter away the most unstable parts of the epidemic
  filter(t>10 & t<70) %>%
  # Create a new column called state to be able to have
  # two sets of points and lines
  pivot_longer(c(i,d), names_to = "state") %>%
  ggplot(aes(t, value, group = state)) +
    geom_line(aes(color=state)) +
    geom_point(aes(color=state)) +
    labs(title = "The number of daily new diagnosed people",
         x = "Time step",
         y="Number of people")

# Plot figure 3
test %>%
  mutate(i = log(i), d = log(d)) %>%
  filter(t>5 & t<70) %>%
  pivot_longer(c(i,d), names_to = "state") %>%
  ggplot(aes(t, value, group = state)) +
    geom_line(aes(color=state)) +
    geom_point(aes(color=state)) +
    labs(title = "The logarithm of daily new diagnosed people",
         x = "Time step",
         y="Logarithm of new people")

# Generates a plot of predicted values against
# test data values of the case when all parameters are fixed
```

```
plot1 <- ggplot(data = data.frame(prediction = preds,
                test_data = log(test_data$i)),
                aes(prediction, test_data)) +
            geom_point() +
            geom_abline(slope = 1, intercept = 0) +
            labs(x = "Predicted value", y="Test data value")

# Generates a plot of predicted values against
# test data values of the case when q and r are randomised, p fixed
plot2 <- ggplot(data = data.frame(prediction = known_p_pred,
                test_data = log(test_known_p$i)),
                aes(prediction, test_data)) +
            geom_point() +
            geom_abline(slope = 1, intercept = 0) +
            labs(x = "Predicted value", y="Test data value")

# Generates a plot of predicted values against
# test data values of the case when q and r are fixed, p randomised
plot3 <- ggplot(data = data.frame(prediction = unknown_p_pred,
                test_data = log(test_unknown_p$i)),
                aes(prediction, test_data)) +
            geom_point() +
            geom_abline(slope = 1, intercept = 0) +
            labs(x = "Predicted value", y="Test data value")

# Generates a plot of predicted values against
# test data values of the case when all parameters are randomised
plot4 <- ggplot(data = data.frame(prediction = unknown_all_pred,
                test_data = log(test_unknown_all$i)),
                aes(prediction, test_data)) +
            geom_point() +
            geom_abline(slope = 1, intercept = 0) +
            labs(x = "Predicted value", y="Test data value")

# Will provide useful plots and information regarding estimates of
# the case when all parameters are fixed during simulation
plot(model)
summary(model)

# Will provide useful plots and information regarding estimates of
# the case when q and r are randomised, p fixed during simulation
plot(known_p_model)
summary(known_p_model)

# Will provide useful plots and information regarding estimates of
# the case when q and r are fixed, p randomised during simulation
```

```
plot(unknown_p_model)
summary(unknown_p_model)

# Will provide useful plots and information regarding estimates of
# the case when all parameters are randomised during simulation
plot(unknown_all_model)
summary(unknown_all_model)

# Plot figure 8
gridExtra::grid.arrange(plot1, plot3, ncol=2)
# Plot figure 9
gridExtra::grid.arrange(plot2, plot4, ncol=2)

# Plot figure 5
sims %>%
  select(logd,logd1,logd2,logd3,logd4) %>%
  rename(`Log(d1)`=logd,
         `Log(d2)`=logd1,
         `Log(d3)`=logd2,
         `Log(d4)`=logd3,
         `Log(d5)`=logd4) %>%
  GGally::ggpair

# Plot figure 16
sim_known_p %>%
  select(logd,logd1,logd2,logd3,logd4) %>%
  rename(`Log(d1)`=logd,
         `Log(d2)`=logd1,
         `Log(d3)`=logd2,
         `Log(d4)`=logd3,
         `Log(d5)`=logd4) %>%
  GGally::ggpair

# Plot figure 15
sim_unknown_p %>%
  select(logd,logd1,logd2,logd3,logd4) %>%
  rename(`Log(d1)`=logd,
         `Log(d2)`=logd1,
         `Log(d3)`=logd2,
         `Log(d4)`=logd3,
         `Log(d5)`=logd4) %>%
  GGally::ggpair

# Plot figure 17
sim_unknown_all %>%
  select(logd,logd1,logd2,logd3,logd4) %>%
```

```
rename(`Log(d1)`=logd,
       `Log(d2)`=logd1,
       `Log(d3)`=logd2,
       `Log(d4)`=logd3,
       `Log(d5)`=logd4) %>%
GGally::ggpair
```

References

[1] Andersson, H. and Britton, T. *Stochastic Epidemic Models and Their Statistical Analysis*. Springer-Verlag New York, Inc. 2000. https://doi.org/10.1007/978-1-4612-1158-7

[2] Bishop Christopher M. *Pattern Recognition and Machine Learning* Springer-VerlagBerlin, Heidelberg, 2006.

[3] Britton, T., Trapman, P. and Ball F. *The risk for a new COVID-19 wave and how it depends on R0, the current immunity level and current restrictions*. R. Soc. open sci.8210386210386. http://doi.org/10.1098/rsos.210386.

[4] Britton, T. *Stochastic epidemic models: A survey.* Mathematical Biosciences 225, 1, 2010, 24-35. https://doi.org/10.1016/j.mbs.2010.01.006

[5] Callaway, E. *Delta coronavirus variant: scientists brace for impact*. Nature. 2021 Jul;595(7865):17-18. doi: 10.1038/d41586-021-01696-3.

[6] Folkhälsomyndigheten
`https://www.folkhalsomyndigheten.se/smittskydd-beredskap/utbrott/aktuella-utbrott/covid-19/testning-och-smittsparning/testa-dig-for-covid-19/`, May 2020
Accessed on 2021-08-12.

[7] Folkhälsomyndigheten
`https://www.folkhalsomyndigheten.se/smittskydd-beredskap/utbrott/aktuella-utbrott/covid-19/statistik-och-analyser/bekraftade-fall-i-sverige/`
Accessed on 2021-08-12

[8] Folkhälsomyndigheten
`https://www.folkhalsomyndigheten.se/contentassets/da0321b738ee4f0686d758e069e18caa/skattning-letalitet-covid-19-stockholms-lan.pdf`
Accessed on 2021-08-12

[9] Knief, U., Forstmeier, W. *Violating the normality assumption may be the lesser of two evils* Behav Res (2021). https://doi.org/10.3758/s13428-021-01587-5

[10] World Health Organization
`https://www.who.int/health-topics/coronavirus`
Accessed on 2021-08-12.