COMPARING COVID-19 MEASURES IN THE NETHERLANDS USING DYNAMICAL MODELLING

Monica R. Panigrahy   Dmitriy "Pike" Knyazhitskiy

May 2021

Word Count: 4395

Abstract

This paper compares the effectiveness of different government measures in the Netherlands to contain Covid-19. We suggest an innovative approach based on modifying the existing constant parameter SIR model to make it dynamic. After validating our model with optimisation in Python, we use the validated model to measure the overall effectiveness of government measures based on flattening waves of Covid-19 cases, and subsequently estimate the contribution of individual government measures in flattening the curve using Tikhonov regularisation method.

Keywords: Covid-19, SIR, flattening the curve, corona measures, BFGS, Tikhonov regularisation

# Contents

1 Introduction

1.1 Covid-19

Unless you've been living under a rock for the past year, you are most likely deeply affected by the ongoing Covid-19 pandemic. Coronavirus disease 2019 or Covid-19 is caused by the SARS-CoV-2 virus, commonly known as the coronavirus. Globally, there are over 168 million cases, and almost 3.5 million deaths at the time of writing. In the Netherlands, over 1.6 million people have been infected, with over 17,000 deaths (WHO Coronavirus Dashboard, 2021). Once infected, a person remains infectious for up to 14 days.

The high volume of cases can be explained by factors behind the transmission of the virus. Covid-19 is a largely airborne disease, and particles spread mainly when people breathe, cough or sneeze, or touch infected surfaces. This spread is particularly amplified indoors when people are in close proximity, hence the employed measures aim to minimise social contact (Morawska et al., 2020).

1.2 Measures to flatten the curve

Since the first onslaught of the virus, governments have focused on "flattening the curve" of Covid-19 waves. This refers to containing the number of cases below a certain threshold which is the maximum treatment capacity of the healthcare system.

Figure 0: Flattening the curve
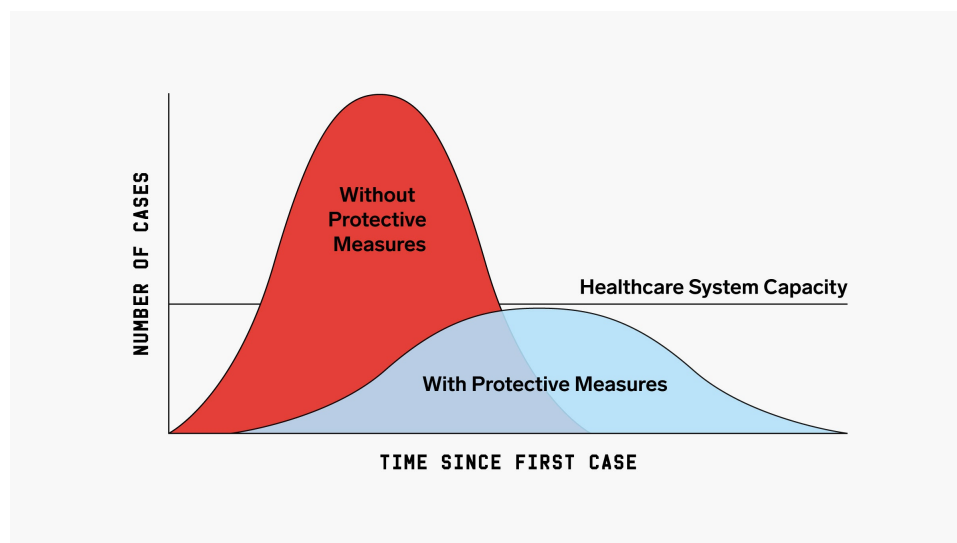
In this paper, measures are as defined by the European Centre for Disease Prevention and Control and assumed to be actions which diminish infection rates. A host of different measures have been implemented with varying levels of success. Such measures include travel restrictions, mandates to work from home, closure of education and commercial establishments, and curfews to name a few. We do not consider vaccines as a measure as they only grant immunity but do not prevent the spread of the virus (van Riel & de Wit, 2020).

Many papers have been published over the last year which compare the effectiveness of different government measures. The most popular methodologies are cross-country studies using straightforward statistical techniques which are largely theoretical and too static for the rapid nature of the virus (Brauner et al., 2021). Besides this, some studies also employ Bayesian hierarchical or mechanistic models which are not very approachable and tend to overstate the significance of the most recent measure (Haug, Geyrhofer, & Londei, 2020). We introduce an innovative approach to measuring effectiveness which is easily scalable and can be adopted in any national framework, and hence is an interesting contribution to this domain of research. By building a robust model, our research aims to provide an additional policy aid for governments.

There is heated debate in the Netherlands about which of the economically-damaging measures such as curfews have played a significant role in the declining trend of cases or if they were redundantly implemented. Such issues can be addressed by our research question:

Which government measures have the largest impact on decreasing Covid-19 cases?

1.3   Outline

This paper is organised as follows: first, we explore existing literature and methods on comparing effectiveness of government measures in containing Covid-19 cases. We then introduce the basis of our model: the SIR model in compartmental epidemiology which is based on ordinary differential equations. We then introduce its dynamic version and establish its validity using Euler's method and limited-memory BFGS method for bounded optimisation, after which we test the model externally with Swedish and Italian data. Finally, we apply Tikhonov regularisation on our validated model to estimate how much different government measures have "flattened" the curve in the Netherlands, allowing us to identify the most impactful measures.

2 Theoretical Framework

2.1 Overview of existing literature on Covid-19 measures

Cross-country studies often evaluate existing measures in countries and divide them into definable units (Abir et al., 2021). Measures must be independent such that correlation between them is minimised, for example travel restrictions and evening curfews are sufficiently different to be addressed as two separate measures. More extensive studies based on Bayesian methods hierarchically code measures into separate categories (Haug et al., 2020). On the basis of these studies, we select a set of most stringent measures in the Netherlands to focus on, based on the chronological order they were introduced in:

Table 1: Dutch measures and the time they were implemented in

Measure	Date
1) Mandate to work from home	12-03-2020
2) Intelligent lockdown: closure of education and entertainment establishments	15-03-2020
3) Closure of EU border	18-03-2020
4) Restriction on number of visitors at home	18-08-2020
5) Face mask mandatory indoors	01-12-2020
6) Hard Lockdown	14-12-2020
7) Curfew-1 (21:00 - 04:30)	21-12-2020
8) Curfew-2 (22:00 - 04:30)	31-03-2021

Obtained from NOS

As you can notice, certain measures such as lockdowns are hybrids of other measures, which we pool together so as to keep our model computationally efficient and avoid the complications arising from finding data on micro-measures such as banning hand shakes. We consider only the impact of government measures on the *incidence* of the virus, so we ignore death rates and only focus on the dynamics of cases. Haug et al (2020) conclude that overall, government measures reduce the number of infections, hence flattening the curve. They also demonstrate that measures have heterogenous impacts based on the time of implementation, hence we make a weighting system for measures, which is described in the later sections.

2.2 Basic SIR model

We base our model on SIR, an existing compartmental model in epidemiology first introduced by Kermack and McKendrick (1927). SIR stands for **S**usceptible, **I**nfected, and **R**emoved. The model involves a system with a

fixed population where the total population of individuals (N) can be classified into three different compartments. Susceptible individuals (S) are all those who have not contacted the virus yet, Infected individuals (I) are currently infected and can pass it to susceptible individuals and Removed (R) is all those who have either recovered from the disease and are immune or those who have unfortunately died and hence are "removed" from the system. These state variables represent the number of people in each category, which varies over time. The time-varying versions of these state variables are denoted as $\dot{S}$, $\dot{I}$ and $\dot{R}$. Also, γ is the rate at which individuals are removed from the system (based on rate at which they recover) and β is the average number of contacts of each individual multiplied with the probability of virus transmission, hereafter referred to as the contact parameter.

Several assumptions are made in this model, both by the creators and us. The model assumes constant γ and β so the recovery period is the same throughout this paper, and each individual is assumed to have equal probability of contacting the disease and equal probability of interacting with another individual. We assume the number of Covid-19 re-infections is negligible and hence there is no movement from R to S or I to S classes (Okhuese, 2020). We also assume that immunity granted by vaccines prevents the movement from R to S class (van Riel & de Wit, 2020) so there is restricted one-way movement between classes. For convenience we assume during the pandemic, people give birth at the same rate at which they die, so the population N is constant, thus death rate equals birth rate.

We consider the rate at which susceptible individuals become infected (hence move from S to I class) to be dependent on the contact parameter, as individuals spread the virus more when they interact with more people. Thus the (decreasing) change in the number of susceptible individuals over time is $\frac{\beta I}{N}$. The number of removed individuals over time is just the rate at which infectious individuals are removed from the system, which means the I class includes all the remaining individuals.

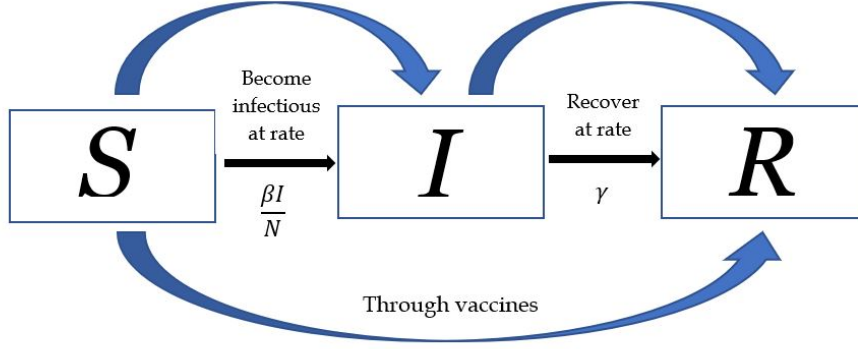Hence we introduce the basic model and parameters based on ordinary differential equations:

$$\dot{S} = \tfrac{dS}{dt} = -\tfrac{\beta IS}{N}$$

$$\dot{I} = \tfrac{dI}{dt} = \tfrac{\beta IS}{N} - \gamma I$$

$$\dot{R} = \tfrac{dR}{dt} = \gamma I$$

Clearly, $N = S(t) + I(t) + R(t) \; \forall t \geq 0$.

Figure 1: Basic SIR model

Now, the SIR model is fairly unrealistic, especially as it is a general epidemiological model but we use it for Covid-19. If government measures are effective, they must impact the contact parameter in some way and hence, it cannot remain constant. What is the motivation behind a dynamic β? Based on findings by Johnston and Pell (2020), we assume that when people observe more infections, they get more cautious, and hence β decreases. Furthermore, we assume that introduction of government measures will cause a downward "jump" in beta and its removal will lead to an increase in beta. For example, when curfew is introduced, we expect number of interactions and hence number of infections to drop, and once curfew is over, number of interactions will increase towards its previous level. The change in β is dependent on the measure implemented. β differs between different measures so evaluating the change in β allows us to investigate infection rates under different measures. A central hypothesis is thus:

Hypothesis H1: *The contact parameter is not constant and varies with the measures implemented.*

3 Methods

Firstly, we make some much needed alterations to the basic SIR model to make it more suited to the dynamic nature of virus transmission. Instead of a fixed β, we estimate a varying $\beta(t)$. The next sections are devoted to defining our model, validating it externally and applying it to obtain what we call the *"flattedness"* parameter, i.e., by how much (each) government measures flatten the curve, in order to estimate their effectiveness. Let

us first describe what kind of data we use to perform the aforementioned steps.

3.1 Data

3.1.1 Model parameters

To estimate the contact parameter in our model, we need data on N, I and γ (data on S and R is optional). Data on the first four variables except S is obtained from census (N), (reported) number of active cases of Covid-19 (I), number of dead and recovered (R) and the remaining population is classified as S. We use a moving average as daily data is too noisy due to data collection issues. We assume $\gamma = 1/14$, so the average time for a person to no longer be infectious is 14 days (Zhai et al., 2020). We use data provided by WHO, as it contains daily Covid-19 cases information from start of pandemic for almost all countries and is a reliable source.

3.1.2 Covid-19 measures

We use the definition of measures and timeline described in the theoretical framework and information from the Dutch government to define observable waves. We then group these points in time with reference to their position in every "wave" of Covid-19. In this way, we assign a weight to each measure in a wave based on the proportion of time in a wave it was implemented for.
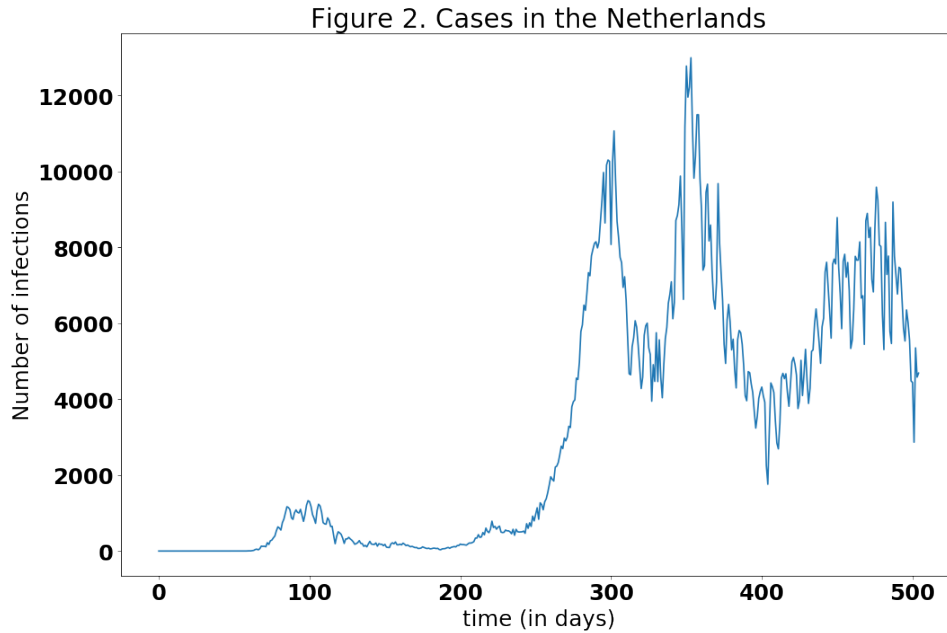

Figure 2. Cases in the Netherlands

Table 2: Periods considered as waves

Wave	Timeline
1	06-03-2020 - 11-05-2020
2	11-08-2020 - 16-11-2020
3	30-11-2020 - 07-02-2021
4	07-02-2021 - 19-05-2021

3.2 Upgraded d-SIR model

In the original model β is assumed to be constant. However, we are cool so we make our beta dynamic and introduce our own variation of the SIR model: the d-SIR, or dynamic-SIR model.

We consider the dynamics of infection as a function of β, hereon referred to as $f(\beta)$. For further purposes we add 2 parameters: duration and scale. Duration (number of days) is the determinant of how many points we generate in this model whereas scale is a multiplier of infections. Besides the original equations of the SIR model, our model includes:

$$y = f(\beta, scale)_{duration} + e$$

Where y is the daily number of infections. All original assumptions from the SIR model except a constant β hold in the d-SIR.

It is important to keep in mind that:

$$f(\beta, scale)_{duration} : \mathbb{R}^2 \longrightarrow \mathbb{R}^{duration}.$$

This function perhaps looks confusing, so let us elaborate on it. SIR model can be treated as a mathematical function where we plug in values of β and generate a stream of infection rates for all future dates. So one number determines an infinite series of output consisting of daily infection rates. However, the main weakness of the constant β SIR model is that this output would always have a similar structure: first an increase in infectious rate and then eventually convergence to 0, which is why we are interested in only a finite number of points where the infection rate significantly changes (usually around 70-200 days for a wave, based on observable data). This is why we don't consider *duration* as a variable, but rather as a parameter. Also, *scale* is simply a constant which we multiply with the output of the model to match it with actual (or scaled) infections number.

3.3   Establishing validity of d-SIR

Our main model builds on SIR, which is one of the most popular compartmental epidemiological models used for depicting the spread of Covid-19. This significantly improves the validity and feasibility of research as a lot of the groundwork has already been laid. By running simulations with data from different countries, we test the robustness of our model under the aforementioned assumptions and obtain an error margin of our results.

We base our extended model on ordinary differential equations as it is the only feasible way to efficiently obtain a numerical solution, and is extensively used in epidemiological modelling.

We start with the numerical solution of the SIR model (without any adjustments), and for this we use the Euler method for solving this system. It is a fairly straightforward method where the next iterate is related to the previous via $y_{n+1} = y_n + hf(t_n, y_n)$, where h is a step size (0,01 in our case, so we perform 100 iterations per day).

Figure 3. Example of solution for beta = 0.3
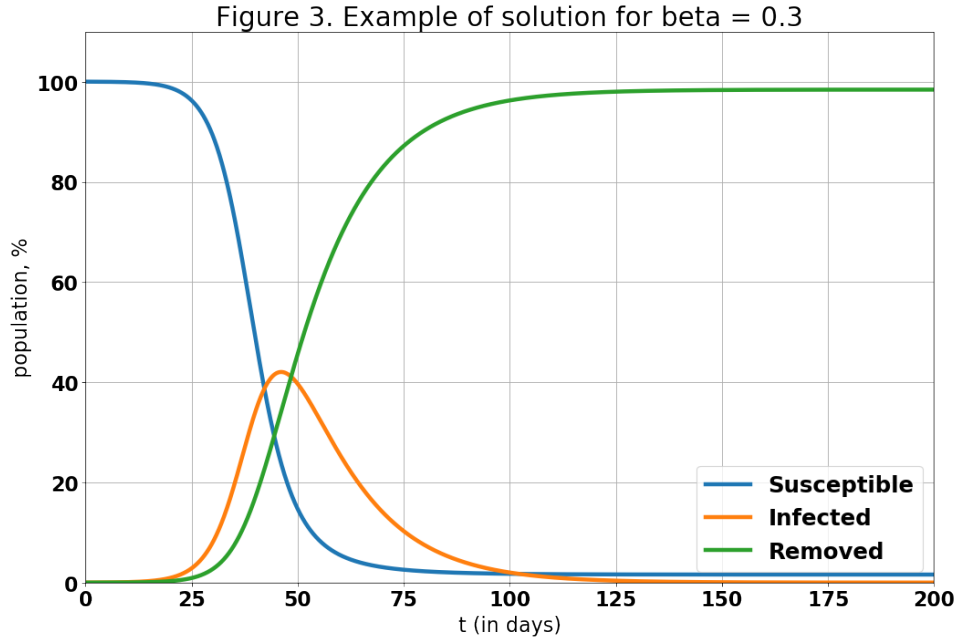
Let us consider a set of solutions of the SIR model as a function of β, or the inverse of the contact rate as the parameter, and γ, or the inverse of the infection period (fixed to be 1/14). We start our simulation where a proportion of 0.0001 of people are infected. Variation of this number will not change the dynamics of infection spread as it is just chosen to be an initial point (see Figure 3).

We perform 2 main steps. Firstly, we verify that in the absence of government measures, we can describe the number of infections by the SIR model. Secondly, we consider periods where measures were introduced and we show that the model estimated for the period without measures cannot be used to accurately forecast the period with measures, hence verifying the need for the d-SIR model. To validate our model with external data from another country and observe the dynamics of the virus under no measures, we shall consider a country different than the Netherlands. We chose Sweden because it was one of the last European countries to introduce any measures so there are more data points available for which we can consider the model without government measures.

To check if the model fits well enough, we use the waves defined by observable data and the timeline of measures we constructed. We select the dates ourselves as leaving it up to the algorithm completely is not a good solution because it will proudly conclude that the best fit is no forecasted infections for the period without actual infections. Imposing constraints that would discourage such behavior is rather challenging and computationally demanding. That is why we introduce variables for start and end date and bound their values. This way we solve a problem of selecting an undesirable optimum point. In other words, we use heuristical inferences and official data on start and end of wave, set this as $start$ and end parameters and allow them to move within a small (in our case, 10 days) interval in any direction. So, the optimisation problem then changes into:

$$\min_{\beta, scale, start, end} \epsilon' \epsilon$$

s.t.

$$start \in (start^* - 5, start^* + 5), end \in (end^* - 5, end^* + 5)$$

$$\epsilon = (y[start:end] - f(\beta, scale)_{end-start})$$

Where $start*$ and $end*$ are defined by our timeline. We use a modification of the BFGS method to perform optimisation. The BFGS method is an iterative approach to solving unconstrained non-linear optimisation problems. You can see the result in figure 4.

Figure 4. Fitted curve and weekly moving average for Sweden
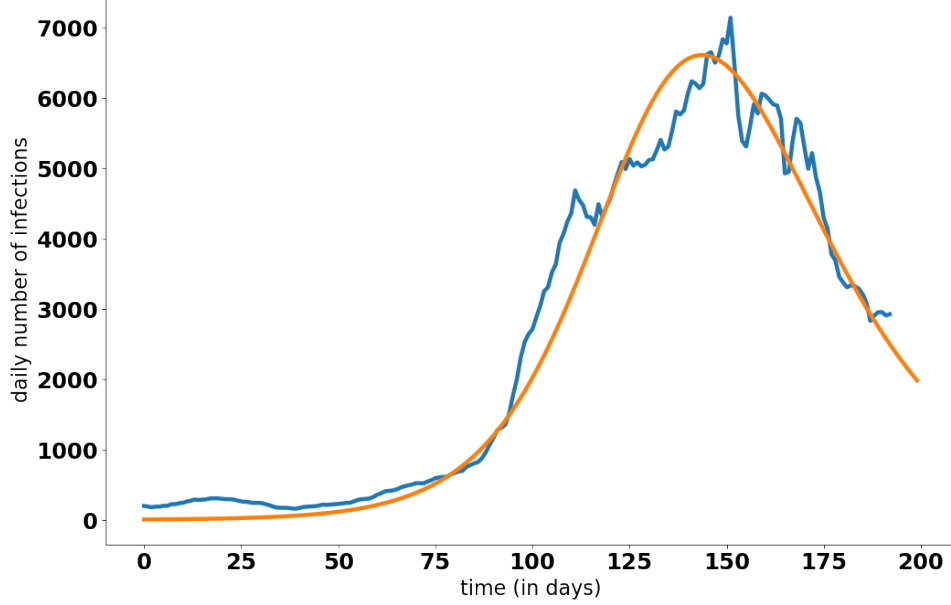
Now, we would like to validate if we indeed can use this model. Given that this is not a standard econometric problem, we cannot consider statistical properties of explanatory variables. We formulate our null hypothesis:

H_0: *SIR can accurately forecast dynamics of Covid-19 cases in the absence of government measures.*

The alternative hypothesis states the opposite. To test the significance of our variables, we use the G-test, which is an asymptotic likelihood ratio test and the test statistic is

$$G = 2 \sum_i O_i \cdot \ln\left(\frac{O_i}{E_i}\right) \sim \chi^2(3)$$

O_i is the observed daily number of infections, E_i is the number expected by our model. This test is an asymptotic χ^2 test. However, G-test is better for data with a large spread of predicted values. Here, some predicted values are close to 0, and small fluctuations in observed values cause significant deviations in output of χ^2 test. The test is however is more robust.

The distribution of the test statistic has 3 degrees of freedom. The test statistic is 7.21 and the 95% critical value is 7.81, so we do not reject the null hypothesis ($p = 0.065$). We conclude that we can use SIR model for forecasting of Covid-19 waves. We also assume a similar structure of infection spread under no measures for any country similar to Sweden in size

12

and proximity. Due to these reasons, we consider the SIR model to be a 'natural' way of infection spread.

Next we want to test if the SIR model can be used to forecast a period with government measures. We consider the very first corona wave in Italy. This is the best period because measures were introduced rapidly and we have a clearly defined date of when this happened (15 March 2020 in case of Italy).

We are going to perform a Chow forecast test in order to see if SIR can forecast data with first measures using data without measures. Despite the fact that the Chow test is originally created for linear models, it is still a type of F-test, and hence (asymptotically) applicable for non-linear models as well, according to our holy Econometrics textbook. We consider predictions for the next 50 days. The null hypothesis for the Chow test:

H_0: *SIR model can accurately forecast data for the period with measures, based on data from period with no measures.*

Our alternative hypothesis H_1 is that SIR cannot, so the coefficient β is different for parts with and without measures. The test statistic for the Chow forecast test is as follows:

$$\frac{(RSS_0 - RSS_1)/n_2}{RRS_1/(n-k)} \sim F(n_2, n-k)$$

Where k is number of independent variables, which is 3; n is the number of days taken for no-measure part (around 90). n_2 is the number of days with measures, intended to be 50, but because our model 'selects' location parameters it might be slightly different.

The test statistic is 176.6, while the 95 % critical value is 1.44. We strongly reject H_0 (p-value = 0.0000). So, there exists a difference in β's for when measures are introduced or not, so we need the d-SIR model. **Hence, we do not reject H1.**

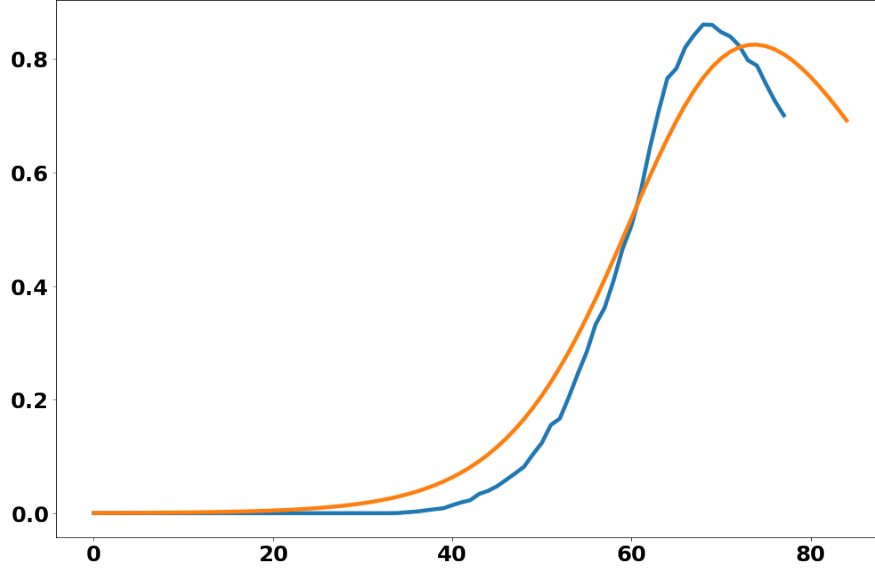Figure 5. Fitted curve and weekly moving average for Italy no measures
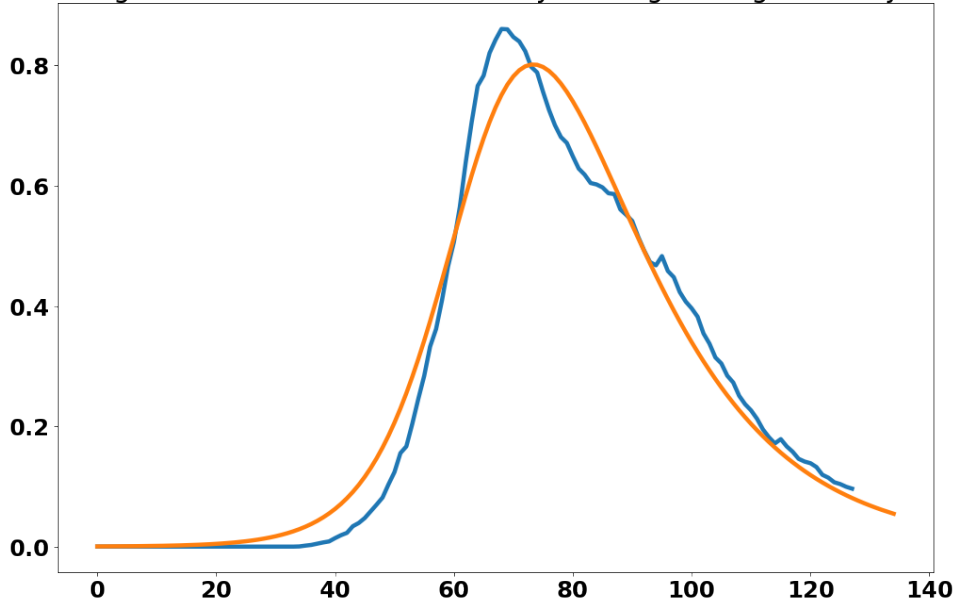

Figure 6. Fitted curve and weekly moving average for Italy

Note: All tests in our research asymptotic LR tests. Their accuracy on finite samples can be questioned, however, if such concerns arise, heuristical comparison of Sum of Squared Residuals can be made. And their values for similar settings would differ in around 10 times for Sweden and Italy waves.

3.4 Fitting d-SIR to Dutch data to measure overall effectiveness of government measures

If you made it this far into the paper, thank you! We reward your patience with the coolest part. After establishing that we need the d-SIR to fit Covid-19 waves with measures, we calculate the overall effectiveness of government measures. The main parameter we use is "flattedness" of the curve, or by how much the combination of government measures reduces the natural peak. It is the (scaled) difference in Covid-19 cases between the 'natural' peak that would be reached without any government interventions and the peak that is reached as a result of government measures. Below you can see pictures of cases with flattedness of 7.5 and 0.2 respectively.

Figure 7. Fitted curve and weekly moving average for Netherlands, flattedness = 0.21
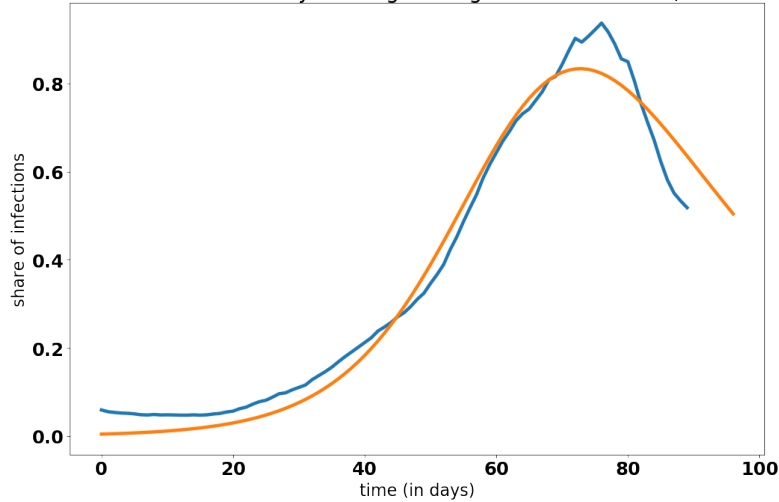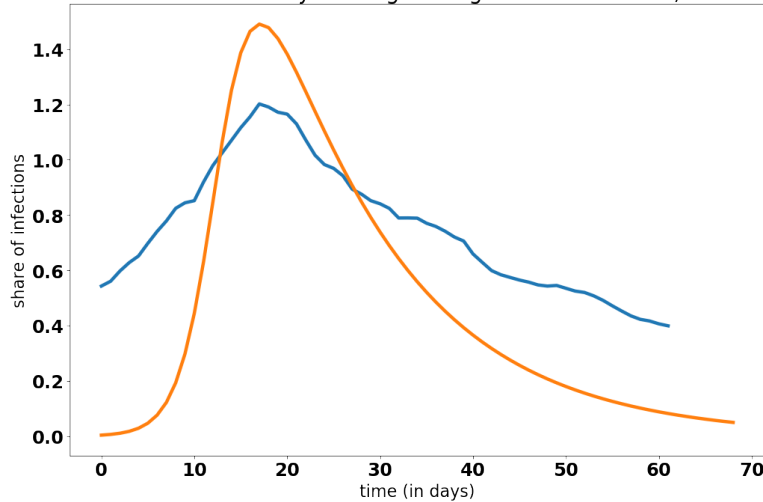

Figure 8. Fitted curve and weekly moving average for Netherlands, flattedness = 7.46

15

We denote the flattedness parameter of each wave in the Netherlands as c_i with $i \in \{1, 2, 3, 4\}$ and it is calculated as:

$$c_i = \frac{SSR_i}{n}$$

Where SSR_i is a sum of squared residuals for interval i and n is duration of wave. An attentive reader might wonder why we refer to this parameter as flattedness while in fact it's a measure of fit. Firstly we notice that this measure is similar to the objective function being minimised when fitting a curve and it would mean our parameter becomes 'less random'. Secondly, some might notice that this parameter doesn't necessary provides us with flattedness. But it does in this case, because for all periods, the shapes for the waves and the fitted curves are alike as on the picture above, so mean squared deviation is a good estimator for flattedness.

Table 3: Flattedness parameter of each wave, measured as (scaled) difference in maximum number of Covid-19 cases

Wave	Flattedness parameter
c_1	0.026
c_2	0.002
c_3	0.108
c_4	0.123

3.5 Estimating the impact of individual measures

After obtaining the overall effectiveness of government measures, we evaluate the contribution of each individual measure to flattening the curve. We assume that the impact of all measures is independent from each other and total impact in each period is a linear combination of measures. In other words, we can express measures as a system of linear equations.

3.5.1 Defining measures

We denote the impact of each measure as m_i and assume a set of possible measures $M = \{m_i\} \ \forall i \in \{1..n\}$. n refers to the total number of different measures. We consider 8 different measures as defined in the theoretical framework and denote them as:

m_1: Work from home
m_2: Intelligent lockdown
m_3: Closing EU borders
m_4: Restricting visitors at home
m_5: Face mask mandatory indoors

m_6: Hard Lockdown
m_7: Curfew 1
m_8: Curfew 2

3.5.2   System of equations with measures

We then assign weights to the measures in each wave based on the amount of time they were implemented for in a particular wave, denoted as the variable coefficients.

We obtain the following system of equations where each equation corresponds to the measures implemented during each wave i:

$$0.93m_1 + 0.91m_2 + 0.8m_3 \rightarrow c_1$$
$$m_1 + m_1 + m_3 + 0.92m_4 \rightarrow c_2$$
$$m_1 + m_2 + m_3 + m_4 + m_5 + 0.85m_6 + 0.68m_7 \rightarrow c_3$$
$$m_1 + m_2 + m_3 + m_4 + m_5 + 0.8m_6 + 0.5m_7 + 0.5m_8 \rightarrow c_4$$

But here comes a small complication - we have 4 periods and 8 measures, so the solution is not unique. In fact, the dimension of solution space is 4, while we want to end up with a definite answer. However, we have some additional information about the solution: firstly, we are interested in relative values (or just rankings) of solutions, and secondly, we want to consider the case where impact of measures is the smallest possible (similar to our null hypothesis).

We solve this issue with Tikhonov regularisation. This is a kind of regularisation endowed by L_2 norm. In fact, it is just a Lasso method with different norm and we use L_2 instead of L_1 for the sake of consistency with techniques above. Regularisation is a machine learning approach to deal with ill-positioned problems, but instead of going into technicalities here, let's explain the intuition behind this process. We are interested in relative impact of measures, not absolute values, while the solutions consist of linear space. The gradient of the solution space is constant, so the *shadow prices* of measures are constant. Then we can introduce objective function ($\|x\|_2^2$ in our case) and minimise it given system of equations as constraints. The vector of solutions will incorporate all requests above. If you are concerned with reliability of this approach, you can perform Lasso method (or methods based on different L_p-norms) - the results will appear to be similar. As we perform Tikhonov regularisation, we solve it as a constrained optimisation problem using Sequential Least Squares Quadratic

Programming. This method uses iterative linear approximations of objective (penalty function). As our objective function is quadratic this approach is perfect for our case.

4   Results

The following estimates for the effectiveness of each measure were obtained:

Table 4: Quantified impact of different government measures

Measure	Impact
Work from home	0.012
Intelligent lockdown	0.011
Closing EU borders	0.005
Restricting visitors at home	-0.030
Face mask mandatory indoors	0.061
Lockdown	0.054
Curfew 1	0.003
Curfew 2	0.028

We would also like to provide a bar plot with results:

Figure 9. Measure intesity graph
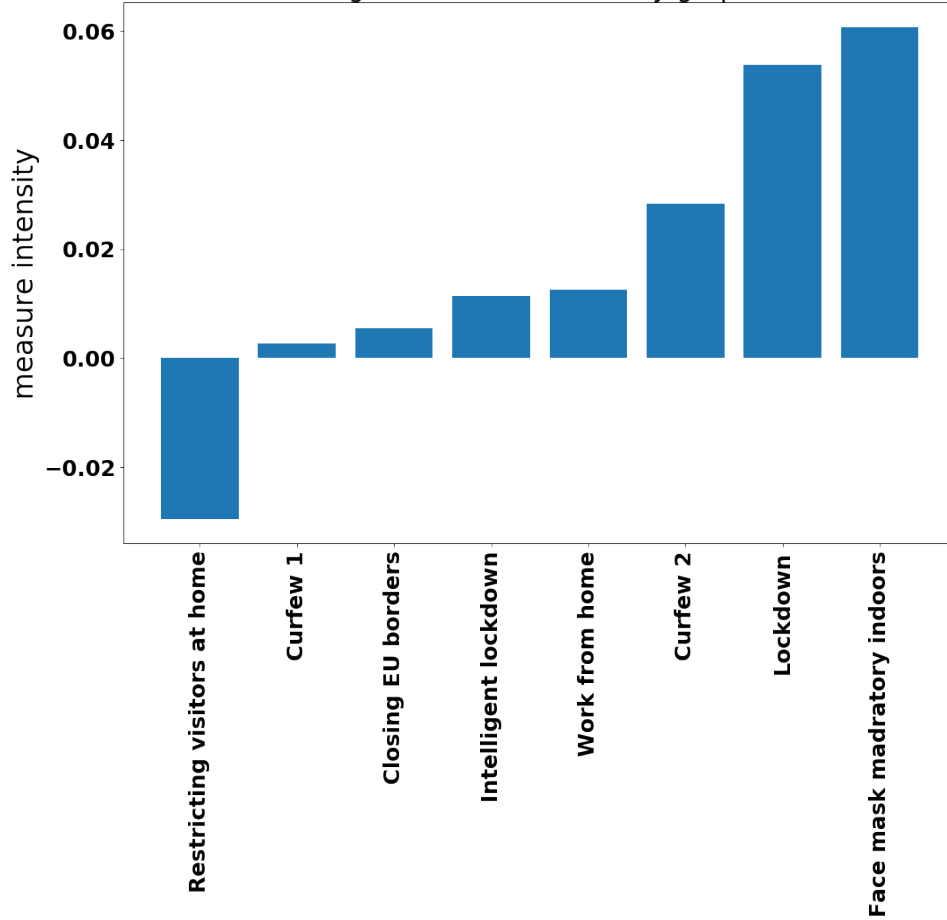
4.1 Interpretation

We have a rather interesting set of results.

Periods: At the start of the pandemic, some weak measures were introduced which led to moderate curve flattering. Most measures were removed in summer of 2020, so we don't observe any flattedness for the second wave (or at least for the start of it). Then from November, some harsh measures were in place and we suddenly see an effect - we get high flattedness. With strict curfew and lockdown we observe an even more significant effect in the last period.

Regarding measures: Mandatory face masks, Hard Lockdown and Curfew-2 seem to have the highest impact. Curfew-1 and closing borders seem to not have large effect. We also find the impact of restricting visitors at home to be negative. We don't consider this result reliable and have a

conjecture about source of possible mistake: the algorithm 'intends' to assign all accumulated errors to this variable. Note that in second period m_4 (restricting visitors at home) has the lowest weight, so when the algorithm needs to "reallocate" some excess measures to satisfy this equation, it will set this variable as negative. Why does it happen in equation 2? Because c_2 has the smallest value and in fact it's very binding. Because the measures in the second period didn't actually work but were still present in our framework, to satisfy the constraint one of them has to have negative impact. It seems to be a problem of evaluated coefficient c_2 and not estimation techniques. From a practical perspective, it is also difficult to measure the effectiveness of restricting home visits because it is more difficult to regulate than the other measures. So implementation of the measure does not necessarily guarantee its enforcement.

4.2 Limitations and Further Improvements

The main problem is this paper is not necessarily the lack of an adequate research method but rather the inconsistency in data measurements and availability which makes it close to impossible to get accurate Covid-19 data. There is also a lag in reporting data so the true effect of government measures is not captured, but due to the systemic nature of lag in the Netherlands, our results still hold on a comparative front. On this front, we've done the best anyone can realistically do: only relying on official WHO data which is the highest quality of data publicly available.

We base this research on the implicit assumption that only government measures affect the change in Covid-19 cases. There are obviously more complex environmental, economic and social factors, but they are very difficult to include in a single model. Right off the bat, it is obvious that these results possibly suffer from corruption due to the presence of latent unobserved variables, for example the effect of measures such as handwashing which is difficult to ascertain. This induces possibly a systematic bias in the results, which could be corrected with a damping factor: for example, clustering all non-government measures and using qualitative techniques to evaluate their effect and then discounting the effect of government measures.

On the other hand, the presence of asymptomatic cases could also result in an over-significance of government measures. An initial idea we had but did not proceed with due to obvious complications is introducing a parameter which represents the individual responsibility of people (such as fear of the disease) and measuring its effect against that of government measures.

For technical limitations see Appendix A.

4.3 Conclusion

We have two main results. Firstly, we (or government) learned how to flatten the curve. The flattedness parameter increases significantly around November, which we consider as evidence of us being finally able to properly combat the virus waves. Secondly, the most effective measures seem to be mandatory face masks, Lockdown and Curfew 2 and the most inefficient - Curfew 1 and closing borders. The impact of restricting home visitors is undetermined. What we have is an easily scalable model where more tailored equations can be added for other countries, with better estimates based on our suggestions.

Furthermore, more research is never bad, and maybe we'll do it ourselves when we grow up.

References

Abir, M., Beckett, M., Huang, W., Al-Ibrahim, H., Chang, J., Schmitzberger, F., ... Hussey, P. (2021). *A comparison of national and international approaches to covid-19-related measures.* Santa Monica, CA: RAND Corporation. doi: 10.7249/RRA438-1

Brauner, J. M., Mindermann, S., Sharma, M., Johnston, D., Salvatier, J., Gavenčiak, T., ... Kulveit, J. (2021). Inferring the effectiveness of government interventions against covid-19. *American Association for the Advancement of Science*, *371*(6531). doi: 10.1126/science.abd9338

Haug, N., Geyrhofer, L., & Londei, A. e. a. (2020). Ranking the effectiveness of worldwide covid-19 government interventions. *Nat Hum Behav*, *4*, 1303–1312. doi: https://doi-org.eur.idm.oclc.org/10.1038/s41562-020-01009-0

Morawska, L., Tang, J. W., Bahnfleth, W., Bluyssen, P. M., Boerstra, A., Buonanno, G., ... Yao, M. (2020). How can airborne transmission of covid-19 indoors be minimised? *Environment International*, *142*, 105832. doi: https://doi.org/10.1016/j.envint.2020.105832

Okhuese, A. (2020, May 13). Estimation of the probability of reinfection with covid-19 by the susceptible-exposed-infectious-removed-undetectable-susceptible model. *JMIR Public Health Surveill*, *6*(2), e19097. Retrieved from `http://www.ncbi.nlm.nih.gov/pubmed/32369029` doi: 10.2196/19097

van Riel, D., & de Wit, E. (2020). Next-generation vaccine platforms for covid-19. *Nat. Mater*, *19*, 810-812. doi: https://doi-org.eur.idm.oclc.org/10.1038/s41563-020-0746-0

Zhai, P., Ding, Y., Wu, X., Long, J., Zhong, Y., & Li, Y. (2020). The epidemiology, diagnosis and treatment of covid-19. *International Journal of Antimicrobial Agents*, *55*(5), 105955. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0924857920301059`

Appendix A: Further technical improvements

We have worked within the restricted assumptions of the most basic compartmental model. Better results could perhaps be obtained with the SEIR model, which is also based on the SIR but includes another compartment: **E**xposed individuals. These are individuals who carry the virus but are not yet infectious. To define this class, we need an additional parameter such as latency of the virus. Including such a parameter corrects the assumption that all those who catch the virus immediately spread it and hence β as the contact parameter will have an additional variable used in its estimation. As such data on Covid-19 is unclear and sparsely available, we would need to estimate the latency as well which is too computationally demanding.

In our model, we kept γ constant at $1/14$ to reduce the computational load and restrict the research to measuring the effect of a single parameter (β). In reality, the period an individual is infectious for varies from 0-14 days, sometimes even exceeding 14 for different variants of the virus. By weighing the measures, we account for the difference of their effects in each wave, but we do not precisely measure the effect with regards to each individual variant of the virus. This is not a big problem however as weighing measures in each wave implicitly measures their effects when different variants of the virus were dominant.

Appendix B: Code

```
1  import numpy as np
2  import matplotlib.pyplot as plt
3  from matplotlib.pyplot import figure
4  import matplotlib
5  import pandas as pd
6  from scipy.optimize import minimize
7  from scipy.optimize import Bounds
8  import scipy.stats
9
```

```python
10  #change default setting to make graohs look better
11  font =   {'weight' : 'bold',
12            'size'   : 22}
13  matplotlib.rc('font', **font)
14
15
16  def get_inf(beta,scale,duration, gamma = 1/14):
17      """
18      input: beta, scale parameters, duration - numbers of days
      we want to consider, gamma - fixed value
19      output: list of infections rates
20
21      """
22      Dt = 0.01     # timestep Delta t
23      S_start=99.9    #healthy but can be infected
24      I_start=0.1    #infectious (the actual number isn't
      relevant as long as it's small)
25      R_start=0     #recovered
26      N=S_start+I_start+R_start    # total population is constant
27      t_start = 0
28      t_end = duration
29      n_steps = int(round((t_end-t_start)/Dt))    # number of
      timesteps
30
31      #initial arrays we fill with data for each 0.01 of day
32      S_arr = np.zeros(n_steps +1)
33      I_arr = np.zeros(n_steps +1)
34      R_arr=np.zeros(n_steps +1)
35      t_arr = np.zeros(n_steps + 1)
36
37      #initialise starting parameters
38      t_arr[0] = t_start
39      S_arr[0] =S_start
40      I_arr[0]=I_start
41      R_arr[0]=R_start
42
43      # Euler's method
44      for i in range (1, n_steps + 1):
45              S = S_arr[i-1]
46              I = I_arr[i-1]
47              R = R_arr[i-1]
48              t = t_arr[i-1]
49              dSdt = - beta * I * S / N
50              dIdt = beta * I * S / N - gamma * I
51              dRdt = gamma * I
52
53              S_arr[i] = S + Dt * dSdt  # calculate  at next
      timestep, add to array
54              I_arr[i] = I + Dt * dIdt
55              R_arr[i] = R + Dt * dRdt
56              t_arr[i] = t + Dt   # add new value of t to array
57
58      return I_arr * scale
59
```

```python
60  # we use a moving average as daily data is too noisy due to the
        data collection issues
61  def movavg(data):
62
63      return [np.mean(data[i-7:i]) for i in range(7,len(data))]
64
65
66  def get_SSR(x):
67      """
68      input: beta, scale parameters, start and end date as index
        of data list
69      output: SSR
70
71      """
72      beta = x[0]
73      scale = x[1]
74      start = x[2]
75      end=x[3]
76      #level = x[4] # we decided not to add more parameters as it
        's clearly overfitting
77      cases = netherlands['New_cases'][int(start):int(end)].
        values
78      cases = list(map(lambda x: x/(max(cases)-min(cases)) ,
        cases )) # scaling
79
80      num_inf = movavg(cases) # smoothing data
81      theory_num = get_inf(beta=beta,scale=scale,duration=len(
        num_inf), gamma = 1/14) #theoretical points
82      theory_num=theory_num[::100][1:] #we had 100 points a day,
        need one
83
84      return  (np.sum([(num_inf[i]-theory_num[i])**2 for i in
        range(len(num_inf))]))
85
86
87  def do_all(start=10, end=90, margin = 5):
88      """
89      input: start and end dates (as positions in dates list)
90             margin - how much we allow algorith to move
        specified piece
91      output: optimisation result, graph, and some descriptive
        statistics
92
93      """
94      start_values=[start - margin, start + margin] # allowed
        values
95      end_values=[end - margin,end+margin]
96
97      # some constraints to simplify optimisation. In fact none
        of them is binding
98      bounds = Bounds((0, 0, start_values[0], end_values[0] ),
        (1.0, 0.08, start_values[1],end_values[1]))
99
100     result = minimize(get_SSR, x0=[0.3,0.03,(start_values[0]+
```

24

```
            start_values[1])/2,(end_values[0]+end_values[1])/2],
101                     method='L-BFGS-B', bounds = bounds)
102     opt_param = result['x']
103
104     # get exact piece selected
105     cases = netherlands['New_cases'][int(opt_param[2]):int(
        opt_param[3])].values
106     cases = list(map(lambda x: x/(max(cases)-min(cases)) ,
        cases ))
107     theory_num = get_inf(beta=opt_param[0],scale=opt_param[1],
        duration=-int(opt_param[2])+int(opt_param[3]))
108     x=theory_num[::100][1:]
109
110     # plotting
111     fig = plt.figure(figsize=(15, 10), dpi=80)
112     plt.plot(range(len(movavg(cases))) ,movavg(cases),
        linewidth = 4)
113     plt.xlabel('time (in days)', fontsize = 20)
114     plt.ylabel('share of infections', fontsize = 20)
115     plt.plot(x, linewidth = 4 )
116     plt.show()
117
118     print('SSR = n* flatterness = '+str(get_SSR(opt_param))) #
        get SSR
119
120
121 def get_beta_flatterness(start=10, end=90, margin = 5):
122     """
123     input: start and end dates
124             margin - how much we allow algorith to move
        specified piece
125     output: value of optimal beta and flatterness - standart
        error of fit
126
127     """
128     start_values=[start - margin, start + margin]
129     end_values=[end - margin,end+margin]
130     bounds = Bounds((0, 0, start_values[0], end_values[0] ),
        (1.0, 0.08, start_values[1],end_values[1]))
131     result = minimize(get_SSR, x0=[0.3,0.03,(start_values[0]+
        start_values[1])/2,(end_values[0]+end_values[1])/2],
132                     method='L-BFGS-B', bounds = bounds)
133
134     opt_param = result['x']
135     cases = netherlands['New_cases'][int(opt_param[2]):int(
        opt_param[3])].values
136     cases = list(map(lambda x: x/(max(cases)-min(cases)) ,
        cases ))
137     theory_num = get_inf(beta=opt_param[0],scale=opt_param[1],
        duration=-int(opt_param[2])+int(opt_param[3]))
138     x=theory_num[::100][1:]
139     SE = get_SSR(opt_param)/len(cases) # standart errors
140
141     return opt_param[0], SE
```

```python
142
143
144  # read data and select the piece we need
145  df = pd.read_csv(r"C:\Users\Dima\Downloads\WHO-COVID-19-global-
         data (1).csv") #read file wit data
146  netherlands = df[df['Country']=='Netherlands'] #select country
147  dates = list(netherlands['Date_reported'].values) #initialise
         dates
148  netherlands.describe()
149
150
151  # remember flatterness parameteres
152  c1 = get_beta_flatterness (dates.index('2020-03-06'), dates.
         index('2020-05-11'))[1]
153  c2 = get_beta_flatterness (dates.index('2020-08-11'), dates.
         index('2020-11-16'))[1]
154  c3 = get_beta_flatterness (dates.index('2020-11-30'), dates.
         index('2021-02-07'))[1]
155  c4 = get_beta_flatterness (dates.index('2021-02-07'), dates.
         index('2021-05-19'))[1]
156
157
158  #implementation of Tikhonov regularisation
159
160  # squared norm of x
161  def HH(x):
162      return (sum([i**2 for i in x]))
163
164  # derivative of vector x
165  def dHH(x):
166      return 2*x
167
168  cons = [{'type': 'eq', 'fun': lambda x: 0.93*x[0]+0.91*x
         [1]+0.8*x[2]-c1 },
169          {'type': 'eq', 'fun': lambda x: x[0]+x[1]+x[2]+0.92*x
         [3]-c2 },
170          {'type': 'eq', 'fun': lambda x: x[0]+x[1]+x[2]+x[3]+x
         [4]+0.85*x[5]+0.68*x[6]-c3},
171          {'type': 'eq', 'fun': lambda x: x[0]+x[1]+x[2]+x[3]+x
         [4]+0.89*x[5]+0.5*x[7]-c4}]
172
173
174  #set initial values randomly and by running it several times we
          can be almost sure there is one optimal point
175  y_0 = [np.random.normal() for i in range(8)]
176  # perform optimisation
177  ssoln = scipy.optimize.minimize(HH, y_0, jac=dHH, method='SLSQP
         ', constraints=cons)
```