Data Science in Life Science

SS20

Quentin Quarantino

PUBLISHED BY PUBLISHER

BOOK-WEBSITE.COM

First printing, March 2019

Contents

Part 1

1. Introduction

In this weeks project each group member was assigned one overarching topic pertaining to the current Covid-19 epidemic. The current epidemic is globalized, with severe consequences to social, health and economic order. As of today 212 countries are affected, with a total of 4,215,274 confirmed cases and a death toll of 284,672 [20]. Within each topic a short introduction to the general concept is given. The understanding of these concepts is then deepened by real world code examples, showing a glimpse of what is possible in each topic in regards to Covid-19.

2. Topic1: Spreading Models

2.1 Background

Modeling the spread of infectious diseases is not only an essential tool in understanding the transmission rates and the trajectory of future cases but also has a significant influence on the appropriate guidelines to control the course of an epidemic. The approaches towards modeling the spread can range from computational models (e.g. agent-based) to mathematical modeling (e.g. compartmentalized models). In this short introduction we will focus on a SIR-model which is part of the compartmentalized subgroup. These models follow a deterministic pattern where each sub-population is divided into groups. In SIR-models each letter stands for one group: $S = susceptible$, $I = infectious$ and $R = recovered/death$. Then it follows that for each time independent point t the rates for each subgroup can be calculated by:

$$dS/dt = \nu N - \beta SI/N - \mu S$$
$$dI/dt = \beta SI/N - \gamma I - \mu I$$
$$dR/dt = \gamma I - \mu R$$

with γ denoting the time rate of death/recovery, β denoting the number of new infections one case causes per time point t, μ denoting general death rate and ν denoting being the birthrate. hey

2.2 Data and Methods

This code is based on the kaggle notebook from *https://www.kaggle.com/lisphilar/covid-19-data-with-sir-model*. It uses python and a basic framework of libraries e.g pandas, sklearn, datetime etc.. The main data used is from the World Health Organization showing novel corona infections by country. Furthermore supplementary data is used to include the age pyramid for each country. The WHO Data set is preprocessed to include the variables: Date, Country, Province, Confirmed, Infected, Deaths and Recovered. A first visualization shows the global rate of infected, deaths and recovered people (Figure 2.1). Next the growth factor is calculated, which is given by: G_n/G_{n-1} with $G = confirmedcases$. Countries with growth factor higher then one have an increasing number

of cases. In contrast growth factor lesser then one shows a declining number of cases. The actual analysis is done for 5 countries: Italy, Japan, India, USA and New Zealand. Giving one case as an

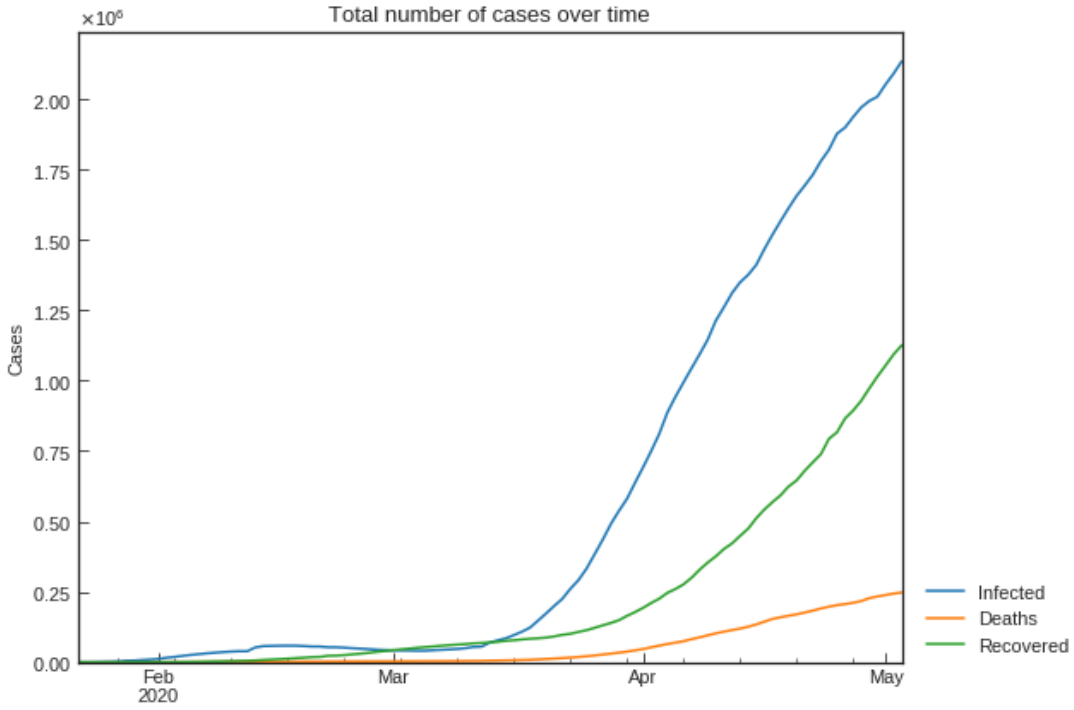

Figure 2.1: Global infections, deaths and recovered people

example as a first step a S-R trend is plotted (Figure 2.2). It shows the trend of susceptible against recovered people. 5 change points can be identified. Next the SIR-F model parameters are estimated for each change point. As a last step the changes in the p value are contrasted with measures taken by the country. While these results are interesting SIR modeling also has its limitations.

2.3 Discussion & Results

Even though the SIR-Model is one of the most basic infectious disease models available it can show promising results with careful consideration for parameter selection and data processing steps. In the case of Italy 3 measures could be shown to reduce the p value: quarantine of person contacted with positive patients, school closure and lock-down. It also has to be considered that the SIR model is based on very basic assumptions. For example the number of susceptible people is treated as fixed as well as the rates of change.

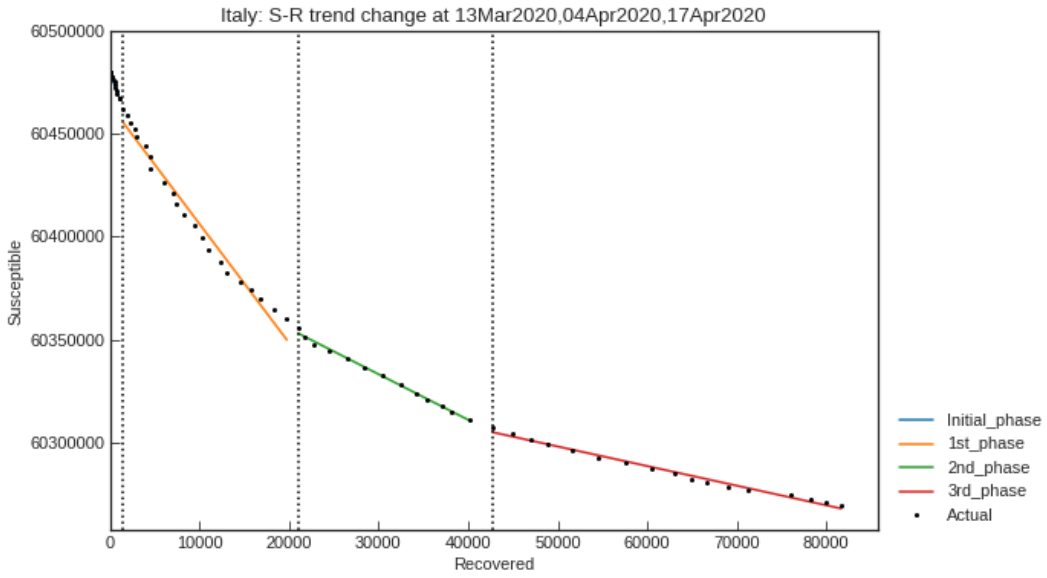

Figure 2.2: Trend of susceptible people versus recovered. 5 distinct change points can be identified.

3. Topic 2: Data-based Time Series Prediction

3.1 Background

Many governments around the world are building their political decisions around the number of current confirmed cases of people infected by COVID-19. Nonetheless, not only the current number of confirmed cases is from greater interest, but also how the virus spreads in the future. One way of forecasting the spread of the virus is by using data-based time series prediction.

3.2 Data and Methods

Therefore machine learning models are calibrated using publicly available data sources like the WHO health report. Time series forecasting can be framed as a supervised learning problem. Other than agent-based spreading simulation such as the SIR model, the models used here do not simulate a population. The forecasting is performed using pythons numpy and sklearn libraries. At first the data is downloaded. Since no data points are missing no preprocessing else than converting integers into date times and reorganizing dataframes is performed. The data contains for a wide range of countries the number of infected people per day starting January 22. A support vector machine model is implemented to forecast the number of infected people. The parameters that have been set can be seen in figure 3.1. The test and test training data sets are generated by splitting them without shuffling them, such that the time series is preserved.

```
[ ]  1  # svm_confirmed = svm_search.best_estimator_
     2  svm_confirmed = SVR(shrinking=True, kernel='poly',gamma=0.01, epsilon=1,degree=5, C=0.1)
     3  svm_confirmed.fit(X_train_confirmed, y_train_confirmed)
     4  svm_pred = svm_confirmed.predict(future_forcast)
```

Figure 3.1: Parameters set for SVM Model.

The model has been trained using the first 75 days since January 22. In figure 3.2 it can be seen that the model over estimates the number of confirmed infections by over 1.5 Million.

3.3 Results

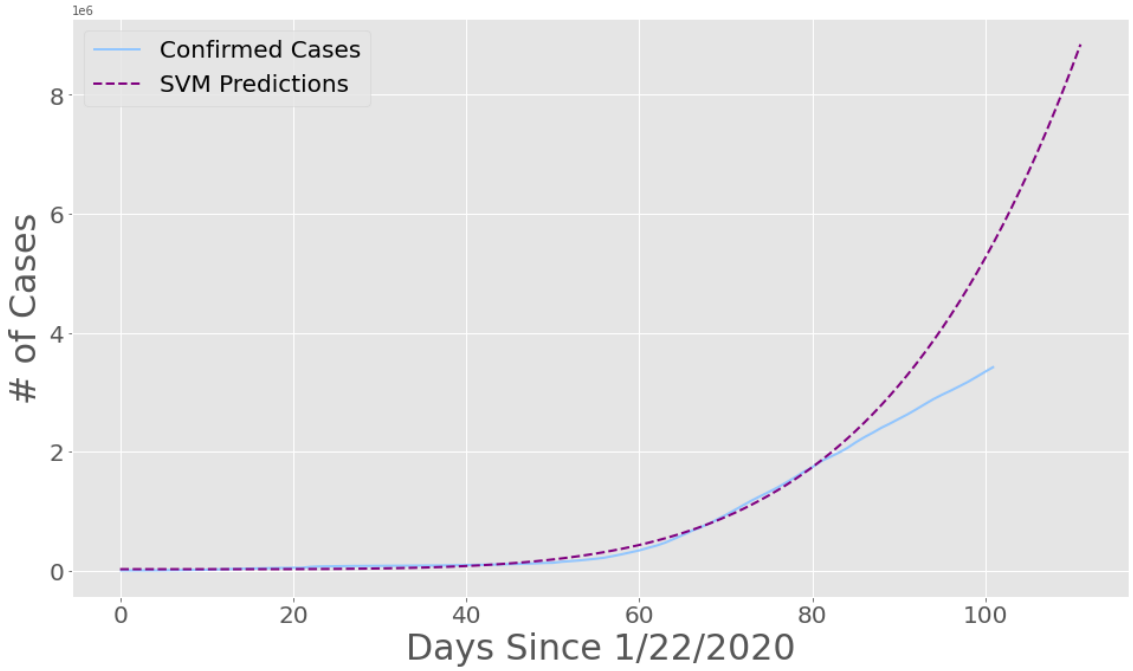

Figure 3.2: Comparison between the observed and the estimated number of infected people.

3.4 Discussion

The shown results are quite underwhelming. This fact has several reasons. One pandemic curves usually increase at the beginning exponentially but then flatten down e.g. because of restrictions in society to decrease the spread of the virus. The model is trained using data from the beginning of the crises where the number of cases rapidly grow. Based on this assumption the estimated number of infected people overshadows the confirmed number. Nonetheless due to different test capacities around the world the estimated might be closer to the real number than it seems to be the case shown in figure 3.2. Still the used model was quite simple and no testing was shown how the parameters were found. A more complex model may gives a better insight to the spread of the virus.

4. Topic 3: Risk Factor Analysis

4.1 Background

The potential dangers of 2019-nCoV have prompted a number of studies on its epidemiological characteristics. It is essential to estimate the number of infections (including those that have not been diagnosed), to be able to analyze the spread of the diseases. To better assess the epidemic risk of 2019-nCoV, among the key parameters to be approximated are the basic reproduction number R0 and the incubation period . Initially we estimate the cumulative number of cases in China outside Hubei province after 23 January, using a time-dependent compartmental model of the transmission dyamics and then we use that number as an input to the global transportation network to generate probability distributions of the number of infected travellers arriving at destinations outside China. Finally using a Galton–Watson branching process to model the initial spread of the virus.

4.2 Data and Methods

The analysis is performed using the python libraries namely numpy, matplot,kiwis solver, scipy and cycler. We computed the risk of the of the individual countries with the selected possible parameters like connectivity and Rloc where Rloc is the local reproduction number of the infection, Getting all the combination of the variables from the data surrounds the neighbour of the china to generate the Heat map.

4.3 Results

Heat map generated gives the information about the outbreak risks as functions of Θ and Rloc, when C = 200,000. The arrows show the directions corresponding to the largest reductions in the risk, which is shown in the figure 4.1

4.4 Discussion

By combining three different modelling approaches helps to assess the risk of 2019-nCoV outbreaks in countries outside of China. This risk depends on three key parameters: the cumulative number of

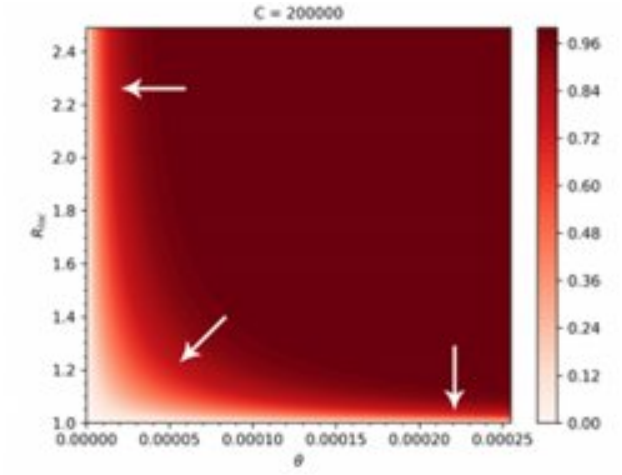

Figure 4.1: Heatmap of the outbreak risks as functions of Theta and Rloc

cases in areas of China which are not closed,the connectivity between China and the destination country, and the local transmission potential of the virus in countries with low connectivity to China but with relatively high Rloc, the most benefecial control measure to reduce the risk of outbreaks is a further reduction in their importation number either by entry screening or travel restrictions. Knowing Rloc and the generation interval are needed not only to have a better quantitative risk estimation, but also for guidance as to which types of control measures may reduce the outbreak risk the most effective.

5. Topic 4: Diagnostic

5.1 Background

The objective of diagnostics is to help effectively diagnose COVID-19 disease. Diagnostics based on RT-PCR-analysis is not very secure due to a high number of false positives. Diagnosis using X-Ray / CT scan images has objective to help effectively diagnose COVID-19 disease with the help of X-Ray/CT scan images in order to improve speed accuracy and scale of diagnosis.

5.2 Data and Methods

X-Ray Detection method reproduced here is done by training a deep learning model using x-ray images (see Figure 5.1) with TensorFlow and Keras in Python to predict whether a patient has COVID-19. The full list of required tools are here (see Figure 5.2)

5.3 Results

So at first X-Ray data were downloaded from the source and python scripts were downloaded. In the next step,anaconda was installed as it contains a lot of preinstalled packages. In separate environment all the packages listed (see Fig 5.2) were installed with needed help tools and also other needed packages needed (like cuda toolkit and cudnn) to run tensorflow were installed (see Figure 5.3) Then the step augmentation of given X-Ray images was performed for both classes covid positive and normal respectively (see Figure 5.4, 5.5) . In this step using 70 covid and 28 normal X-Ray data were 5088 covid and 2424 normal augmented data generated. In the next step the model was trained and tested using augmented data. The augmented data were divided in train and test data. 80% (6009 data) of augmented data were used as train data and were included in model and 20% (1503 data) of data were used as test data for predictions (see Figure 5.6, 5.7). The model was validated for 1503 test data 100 times with 46 repetitions . Using confusion matrix specificity, sensitivity and accuracy values were estimated and plotted.

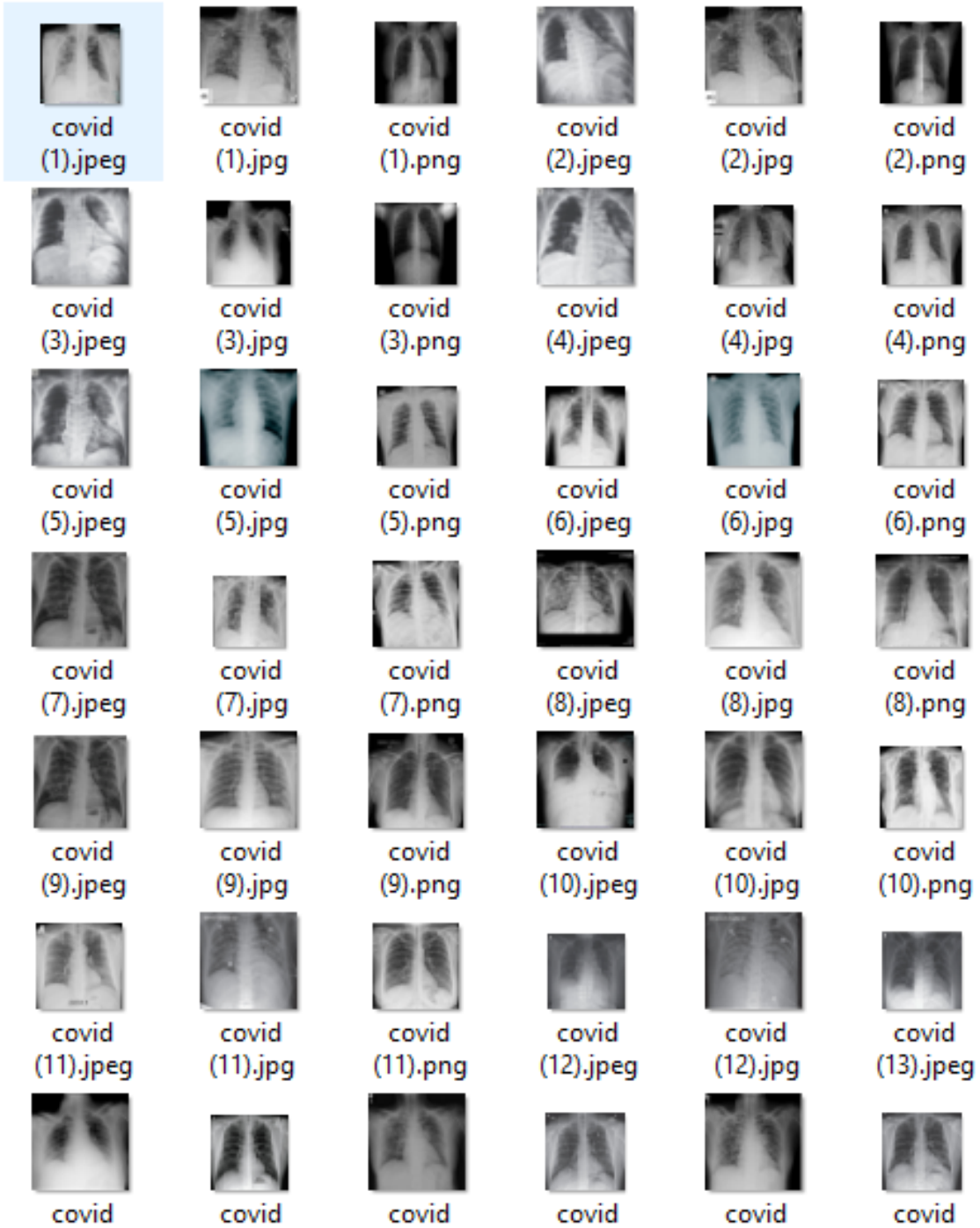

Figure 5.1: Xray data

```
absl-py==0.9.0
astor==0.8.1
cachetools==4.0.0
certifi==2019.11.28
chardet==3.0.4
cycler==0.10.0
gast==0.2.2
google-auth==1.11.3
google-auth-oauthlib==0.4.1
google-pasta==0.2.0
grpcio==1.27.2
h5py==2.10.0
idna==2.9
imutils==0.5.3
joblib==0.14.1
Keras==2.3.1
Keras-Applications==1.0.8
Keras-Preprocessing==1.1.0
kiwisolver==1.1.0
Markdown==3.2.1
matplotlib==3.2.0
numpy==1.18.2
oauthlib==3.1.0
opencv-python==4.2.0.32
opt-einsum==3.2.0
pandas==1.0.2
Pillow==7.0.0
protobuf==3.11.3
pyasn1==0.4.8
pyasn1-modules==0.2.8
pyparsing==2.4.6
python-dateutil==2.8.1
pytz==2019.3
PyYAML==5.3
requests==2.23.0
requests-oauthlib==1.3.0
rsa==4.0
scikit-learn==0.22.2.post1
scipy==1.4.1
six==1.14.0
sklearn==0.0
tensorboard==2.1.0
tensorflow==2.1.0
tensorflow-estimator==2.1.0
tensorflow-gpu==2.1.0
tensorflow-gpu-estimator==2.1.0
termcolor==1.1.0
urllib3==1.25.8
Werkzeug==1.0.0
wrapt==1.12.1
```

Figure 5.2: List of required tools

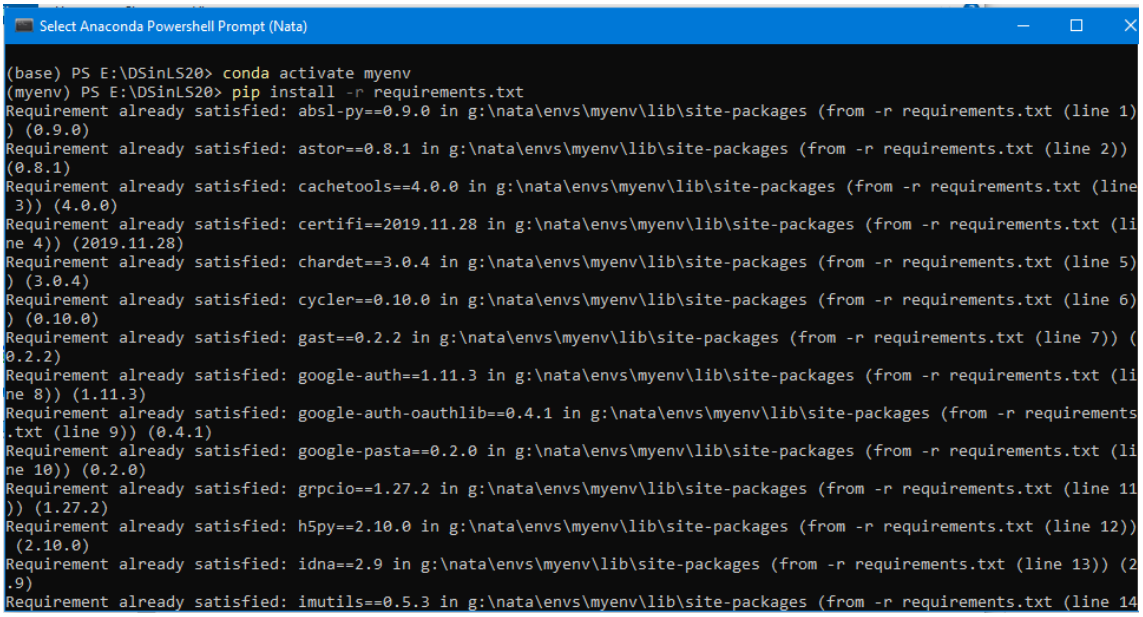

Figure 5.3: Installation of packages

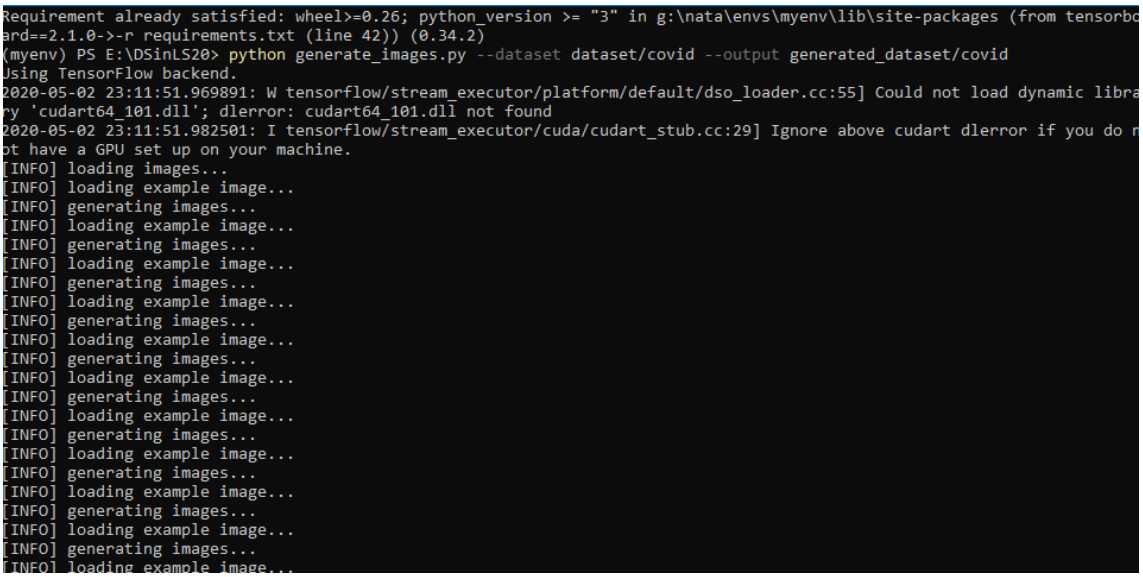

Figure 5.4: Augmentation step for class covid

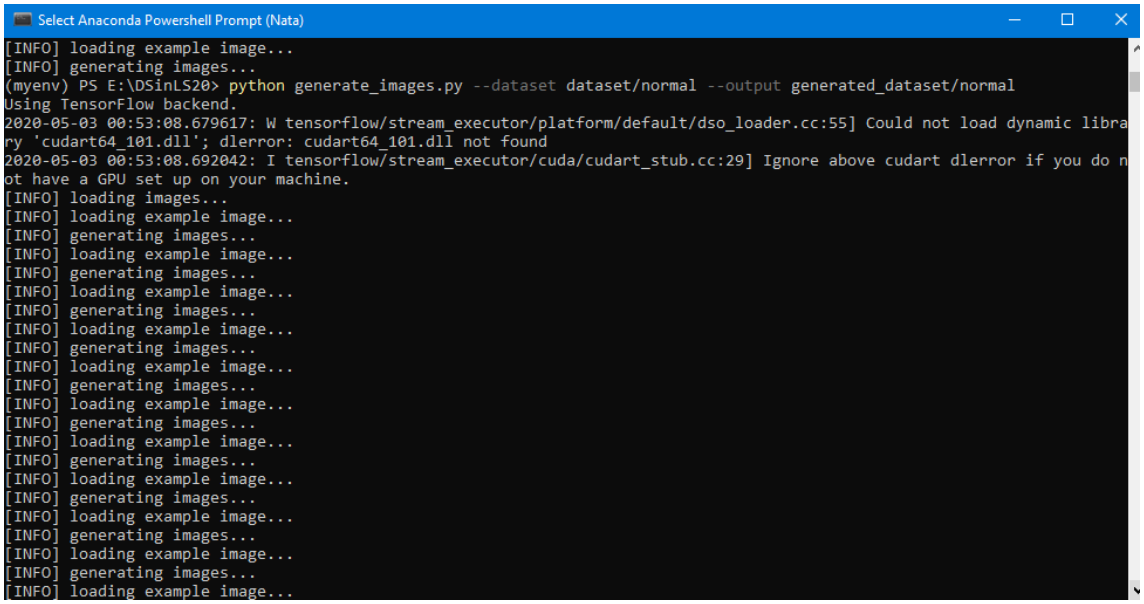

Figure 5.5: Augmentation step for class normal

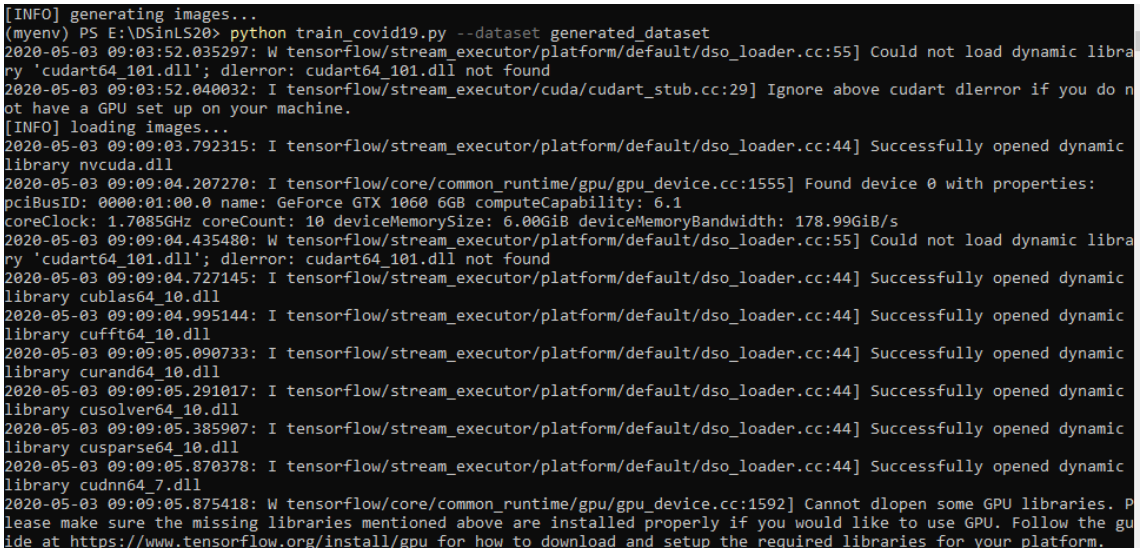

Figure 5.6: Training and Validation step

```
45/46 [============================>.] - ETA: 29s - loss: 9.0710e-04 - accuracy: 1.00
46/46 [==============================] - 1686s 37s/step - loss: 8.8856e-04 - accuracy
: 1.0000 - val_loss: 1.0893e-04 - val_accuracy: 1.0000
[INFO] evaluating network...
              precision    recall  f1-score   support

       covid       1.00      1.00      1.00      1018
      normal       1.00      1.00      1.00       485

    accuracy                           1.00      1503
   macro avg       1.00      1.00      1.00      1503
weighted avg       1.00      1.00      1.00      1503

[[1017    1]
 [   0  485]]
acc: 0.9993
sensitivity: 0.9990
specificity: 1.0000
[INFO] saving COVID-19 detector model...
(myenv) PS E:\DSinLS20>
```

Figure 5.7: Screenshot of Validation process

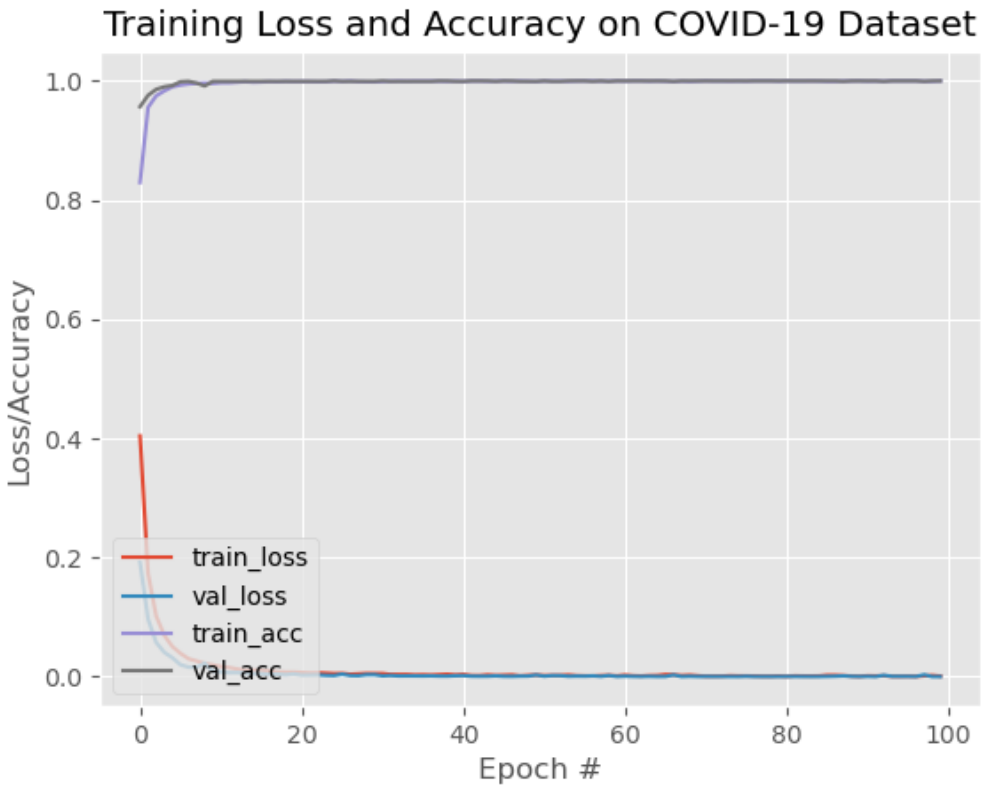

Figure 5.8: Plot of Validation

5.4 Discussion

Approach based on deep Learning described here is a very promising tool for Covid-19 detection in lungs. But on the other hand it is very time consuming. All the steps of this pipeline are very time consuming. Augmentation of images took 2.5 hours. Training and testing using model took about 40 hours. The accuracy, specificity and sensitivity are very high (see Figures 5.7, 5.8) and prove that this approach is very useful.

6. Topic 5: Origin Analysis

6.1 Background

Phylogenetic analysis aims to reconstruct phylogenies both for a group of species and also for the individuals within those species. For pathogens, we are interested in the evolution of the diseases at the genetic level, and what this can tell us about their past and present diversity [18]. An important fact about the Coronaviviridae family is that it's member tend to "jump" from one species to another. When the transmission occurs from a non-human host to a human host it is called zoonosis [8]. The determination of the most recent common ancestor of the human SARS-CoV-2 and the zoonotic transmission can provide important information about biological features, key mutations and properties of the virus. A detailed understanding of how an animal virus jumped species boundaries to infect humans will help in the prevention of future zoonotic events. [2].

6.2 Data and Methods

We will compare the genetic sequence of SARS-CoV-2 with other viruses of the Coronaviridae family in different hosts. The following analysis is based on a Github repository of Simon Burgermeister [5]. Six complete genomes were considered, whose names and hosts are listed in Table 6.1. The sequence data (fasta files) were downloaded from the NCBI Virus public library [9]. To compare the genetic sequences, a multiple sequenced alignment needed to be performed. Clustal Omega is a software that uses seeded guide trees and HMM profile-profile techniques to generate alignments between multiple sequences. Unfortunately, my local computer was not able to compute the alignment due to RAM exceedance. Therefore, I submitted a request to the online version of Clustal Omega [14]. Based on the resulting alignment, a distance matrix was calculated with the *TreeConstruction* package from Biopython. Afterwards, the same package was used to create the phylogenetic tree base on the UPGMA algorithm.

6.3 Results

The resulting phylogenetic tree (Figure 6.1) shows that our human SARS-CoV-2 sequence is most similar to the SARS-like coronavirus sequence of the Rhinolophus (horseshoe bat) with a similarity

Accession number	Host	Description
MN996528	H. Sapiens	Human SARS-CoV-2
NC_019843	H.Sapiens	Human MERS-CoV
JQ065048	Anatidae	Ducks, geese and swans
MG772934	Rhinolophus	Horseshoe bats
NC_034972	Apodemus chevrieri	Rodent
KX38909	Gallus gallus	Chicken
MT084071	Manis javanica	Pangolin

Table 6.1: Considered Coronaviridae strains and hosts.

of 96%. The host with the next similar sequence is the Manis javanica (Pangolin) with a similarity of 0.89% between their genomes. The human MERS-Cov genome and the SARS-CoV-2 genome share only a sequence similarity of 0.74%.

6.4 Discussion

As many early cases of COVID-19 were linked to the Huanan market in Wuhan [21], it is possible that an animal source was present at this location. Given the similarity of SARS-CoV-2 to bat SARS-CoV-like coronaviruses, it is likely that bats serve as reservoir hosts for its progenitor. Although the similarity of 96% to the coronavirus sequence hosted by the Rhinolophus, Andersen et al. [2] identified that its spike protein diverges in the receptor binding domain (RBD), which suggests that it may not bind efficiently to the human ACE2 receptor. Furthermore, it is assumed in this and other studies [11, 13] that an intermediate host was probably involved.

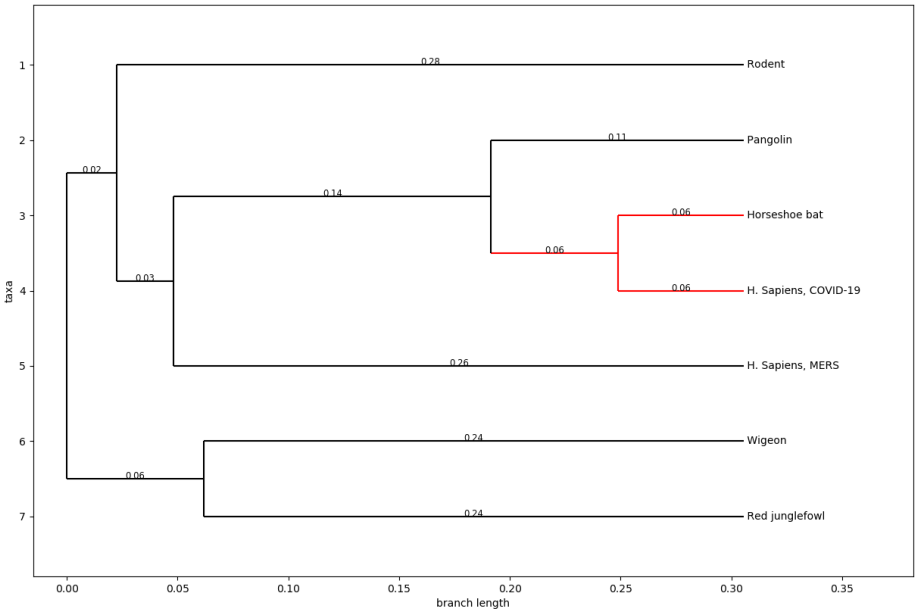

Figure 6.1: Phylogenetic tree of the origin detection analysis.

II

Part 2

7. Introduction

7.1 Goal of the Project

The overwhelming amount of daily published papers correlated to the corona virus makes it difficult, even for health professionals, to keep up with new information about the virus. One way of managing the flood of information is by clustering them according to their topics to simplify the search. Therefore, we have performed a cluster analysis of the CORD-19 dataset, which contains roughly 60.000 articles.
After parsing the body of each article in the dataset, the extracted information is transformed into a feature vector. We afterwards apply dimensionality reduction using PCA and performed a k-means clustering. Subsequently, t-SNE is applied to project the original feature vector into two dimensions such that clusters become visible in the two dimensional space.
Each course participant selected five scientific papers that cluster in the same group as the article they introduced in *Part* I. The submissions have been used to create a new dataset. K-meanswas also applied to this dataset, to determine the cluster assignments and investigate patterns in the data. Finally, the selected articles of *Part* I were added to the CORD-19 dataset. The clustering was redone to see if the papers will be assigned to the expected clusters. In addition to re performing the clustering, two methods for selecting the best k value and two distance metrics were compared: Silhouette Scoring vs Distortion and Euclidean vs Cosine Similarity.

7.2 Outcome

The five papers we selected in *Part* I were clustered into 3 different groups, whereas three of the papers have been assigned to the same cluster. Comparing both, the method of choosing k by elbow point or silhouette scoring and the distance metrics euclidean and cosine similarity, we determined that silhouette scoring and euclidean distance performed better. 10 clusters with unique topics were found (Table 8.2).

8. Tasks

8.1 Part1

The literature clustering pipeline started with the data import of the CORD-19 dataset. Since we wanted to perform the calculations in a Google Colab notebook, we decided to create an API connection to the kaggle database. Using the API, we were able to download and unzip the dataset on our personal Google drive the fastest way possible. The resulting metadata dataframe listed 59.887 entries of coronavirus related publications.
The metadata information are subsequently merged with the body text of the papers that are stored in separated json files. Due to partially missing information only 43.331 entries of the metadata could be merged with the json files. To get an overview of the average text length of the abstracts and the body text information (on which the clustering will be performed) the overall and unique number of words were calculated. The result was an average abstract length of 157 words and an average body text length of 4.528 word (1376 unique). Since the data was uploaded by many different sources, duplicates were present in the dataset. These need to be filtered out such that 30.960 publication remained in the set. The subsequent calculation steps of the pipeline will require very high computing resources. Therefore, we randomly subsamples (seed=42) the dataset to a maximum of 10.000 instances. Unfortunately, we noticed afterwards that both, entries containing null values (1073) and non-english publications (242) were still present in the data. Since these would massively reduce the interpretability of the clustering result, they were also dropped. The final dataset consisted then of 8685 entries.
Another applied proprocessing step was to detect and remove stop words. These are common words in the written text, that do not contribute to the content and act as noise in the clustering procedure. The *spacy* package was used to determine the stopwords. Additionally, a predefined list of stopwords was appended to the list, that contained frequently used words of scientific publication in general. The last step of the preprocessing was to vectorize the cleaned data. Hereby,the string formatted data is converted into a vector-based measure of how important each word is to the instance out of the literature as a whole usind the *tf-idf* package. This method creates a very high feature space and since a clusering by k-means need to be performed, a Principle Component Analysis (PCA) was applied to reduce the amount of feature by simultaneously keeping 95% of the

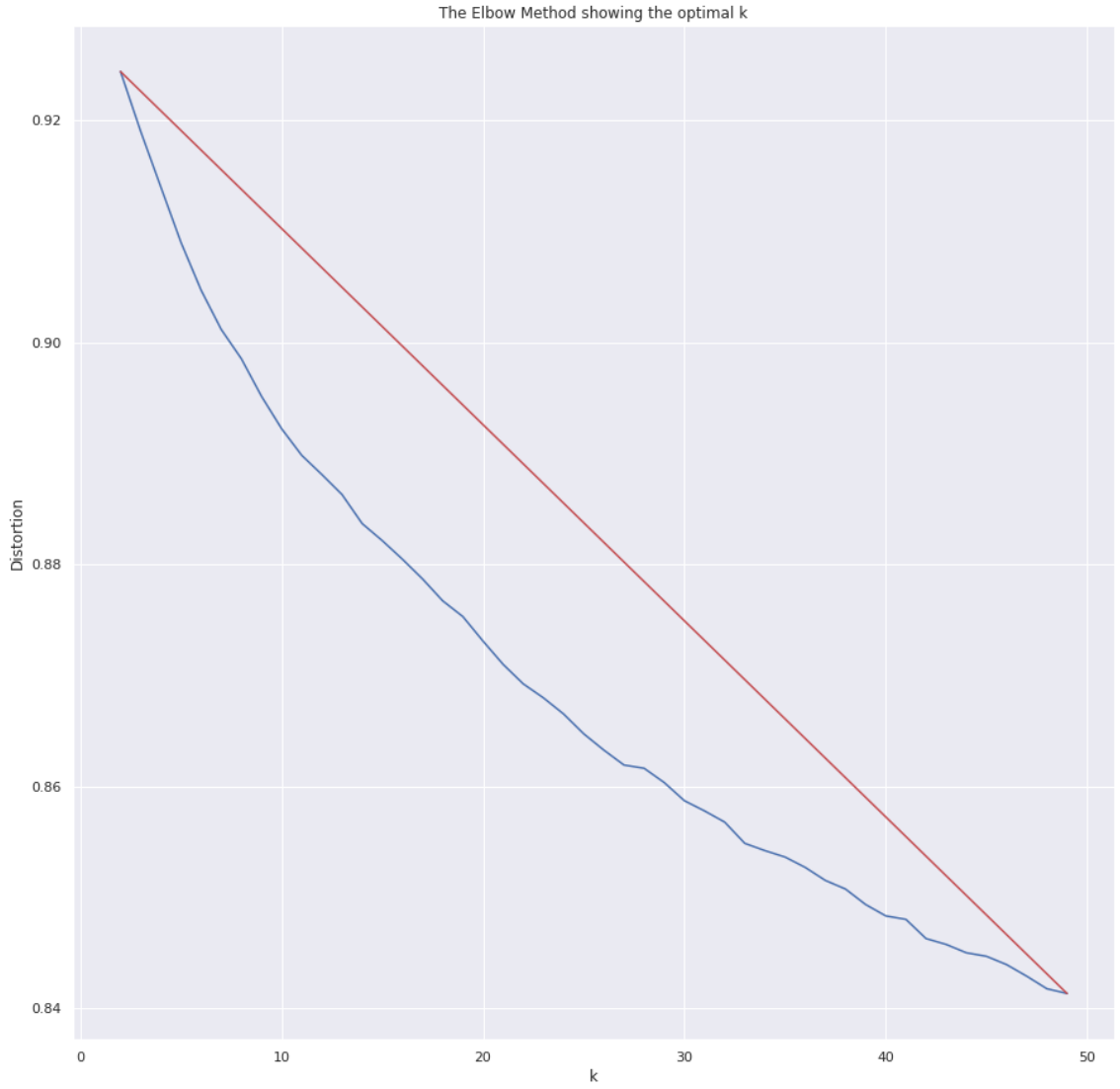

Figure 8.1: The figure shows an elbow plot of the k-means clustering the the distortion on the y-axis and the number of clusters on the x-axis.

data's variance and immensely reducing the algorithm's runtime. The best k number of clusters was determined by iterating through different values of k from two to 50. The resulting elbow plot (Figure 8.1) shows the elbow point at k=27, which is subsequently used as the best number of clusters.
A t-Distributed Stochastic Neighbor Embedding (t-SNE) was used to reduce the high dimensional features vector to two dimensions. This step provides to possibility to represent the clustering result in a plain coordinate system. The aim of the entire pipeline was to create an interactive bokeh plot. To create the plot, the results of all previous calculations are brought together. The location of each paper on the plot is determined by t-SNE while the label (color) is determined by k-means. Interestingly, the assignments match each other very well, even though they were calculated separately (Figure 8.2). Now, the clusters are calculated, but the information about the kind of papers, that are matched together is still missing. To solve this task, a Latent Dirichment Analysis (LDA) was performed to model the most important topics for each cluster. This information is also included in the final bokeh plot.

Figure 8.2: The plot shows the clustering result with the t-SNE positioning and the k-means labels.

8.2 Part2

In this task we were supposed to add the metadata information of the selected papers from *Part* I to the CORD-19 dataset and perform the clustering again. We wanted to find out if this approach improves and facilitates the search of papers for a specific topic field. In order to solve this task, we tried to generate a dataframe using the *pyPDF2* package to extract the metadata from the pdf files of the articles. Unfortunately, it did not work for all pdfs, because they did not contain uniform metadata fields. For this reason we decided to create a csv file and added all relevant metadata fields manually (Table 8.3). The manually created table was then added to the CORD-19 dataset (Figure 8.4).

Link	Title
https://www.ncbi.nlm.nih.gov/pubmed/32276116	Rapid and visual detection of 2019 novel coronavirus (SARS-CoV-2) by a reverse transcription loop-mediated isothermal amplification assay
https://www.nature.com/articles/s41598-018-37483-w	A method to identify respiratory virus infections in clinical samples using next-generation sequencing
https://www.tandfonline.com/doi/full/10.1586/14737159.2014.888313	Advances and challenges in biosensor-based diagnosis of infectious diseases
https://ann-clinmicrob.biomedcentral.com/articles/10.1186/1476-0711-7-18	Predicting the sensitivity and specificity of published real-time PCR assays
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3522074/	Application of Molecular Diagnostic Techniques for Viral Testing

Table 8.1: Papers to diagnostics

	A	B	C	D	E	F	G	H
1	paper_id	doi	abstract	body_text	authors	title	journal	abstract_summary
2	week_2_spreading_models	10 3390/ijerph 16234683	Infectious diseases are an important cause of human death The study of the pathogenesis spread regularity and development trend of infectious diseases not only provides a theoretical basis for future research on infectious diseases but also has practical guiding significance for the prevention and control of their spread In this paper a controlled differential equation and an	Infectious diseases are diseases that can be transmitted from person to person from person to animal or from animal to animal after proto-microorganisms and parasites infect human beings or animals [1–3] Infectivity epidemic and uncertainty are the three main characteristics of infectious diseases A thorough study of the spread	Bin Sheng Sun Gengxin Chen Chih-Cheng	Spread of Infectious Disease Modeling and Analysis of Different Factors on	International Journal of Environmental Research and Public Health	something
3	week_2_risk	10 3390/jcm90 20571	We developed a computational tool to assess the risks of novel coronavirus outbreaks outside of China We estimate the dependence of the risk of a major outbreak in a country from imported cases on key parameters such as: (i) the evolution of the cumulative number of cases in mainland China	A cluster of pneumonia cases in Wuhan China was reported to the World Health Organization (WHO) on 31 December 2019 The cause of the pneumonia cases was identified as a novel betacoronavirus the 2019 novel coronavirus (2019-nCoV recently renamed as SARS-CoV-2	Boldog Péter Tekeli Tamás Vizi Zsolt Dénes	Risk Assessment of Novel Coronavirus COVID-19 Outbreaks	Journal of Clinical Medicine	something
4	week2_forecasting	https://doi org/10 1016/j idm 2020 03 002	Based on the official data modelling this paper studies the transmission process of the Corona Virus Disease 2019 (COVID-19) The error between the model and the official data curve is quite small At the same time it realized forward prediction and backward inference of the epidemic situation and the relevant analysis help relevant countries to	At the end of 2019 the new coronavirus (COVID-19) spread widely in China and a large number of people became infected At present the domestic outbreak has been effectively controlled while the new coronavirus is spreading rapidly in other areas Currently Europe has become the center of	Li Lixiang Yang Zihang Dang Zhongkai Meng Cui Huang	Propagation analysis and prediction of the COVID-19	KeAi	something
5		https://ww	Testing for COVID-19 has been unable to keep up	Identifying who has the COVID-19 virus is	Hall	Finding		

Figure 8.3: csv-table

We prepossessed the dataframe and applied the previously describes clustering pipeline. Finally, the cluster membership of each paper from *Part* I could be identified. The titles of papers from the cluster were saved as .csv-file respectively (see Figure 8.5).

It could be figured out that three papers (spreading models, databased time-series prediction and risk factor analysis) from *Part* I belong to the same cluster 6 (Figure 8.6). The paper related to diagnostics was assigned to cluster 15 (Figure 8.7) and the origin analysis article was a member of cluster 9 (Figure 8.8).

2.3 Appending metadata of papers from week2 ¶

Loading metadata of papers from week2 from csv file as dataframe, preprocessing (stripping whitespace, removing char, adding word counts of abstract, body text and unique words) and appending metadata to a general dataframe containing all the papers), dropping duplicates, data summary

In [39]:
```
import pandas as pd
df_papers_week2 = pd.read_csv('C:/Users/Natalja/shared_folder/DSinLS20/week3/Week2_Papers.csv', sep=';',  dtype={'paper_id' : str
for column in df_papers_week2:
    df_papers_week2[column]= df_papers_week2[column].apply(lambda x: x.strip())
df_papers_week2
df_papers_week2['title']=df_papers_week2['title'].str.replace('<br>',' ')
df_papers_week2.drop_duplicates(subset='title', keep='first', inplace=True)
df_papers_week2['abstract_word_count'] = df_papers_week2['abstract'].apply(lambda x: len(x.strip().split()))  # word count in abs
df_papers_week2['body_word_count'] = df_papers_week2['body_text'].apply(lambda x: len(x.strip().split()))  # word count in body
df_papers_week2['body_unique_words']=df_papers_week2['body_text'].apply(lambda x:len(set(str(x).split())))  # number of unique wo
df=df_papers_week2.append(df)
df.drop_duplicates(['abstract', 'body_text'], inplace=True)
df['abstract'].describe(include='all')
```

Out[39]:
```
count     10005
unique     7205
top
freq       2794
Name: abstract, dtype: object
```

Figure 8.4: Appending metadata of week2 papers to a data frame containing metadata to CORD-19-research-challenge

Identifying of cluster for each week2 paper

using helper function save_cluster_week2_titles clusters of each paper from week2 identified and paper titles of this cluster will be saved in extra .csv-file

In [113]:
```
def save_cluster_week2_titles(title, table):
    cluster=table.loc[table['title']== title, 'y'].values
    #print(cluster)
    cluster_value=cluster[0]
    print('Cluster {}'.format(cluster_value))
    cluster=table.loc[table['y'] == cluster_value, 'title']
    nameId_week2=table.loc[table['title'] == title, 'paper_id'].values
    nameId_week2=nameId_week2[0]
    print(nameId_week2)
    savePath = 'C:/Users/Natalja/shared_folder/DSinLS20/week3/df_covid_cluster_topic_{}.csv'.format(nameId_week2)
    cluster.to_csv(savePath, index = False)
for index, row in df_papers_week2.iterrows():
      print ('Title of week 2 paper is {}'.format(row['title']))
      save_cluster_week2_titles(row['title'], df)
```

```
Title of week 2 paper is Spread of Infectious Disease Modeling and Analysis of Different Factors on Spread of Infectious Diseas
e Based on Cellular Automata
Cluster 6
week_2_spreading_models
Title of week 2 paper is Risk Assessment of Novel Coronavirus COVID-19 Outbreaks Outside China
Cluster 6
week_2_risk
Title of week 2 paper is Propagation analysis and prediction of the COVID-19
Cluster 6
week2_forecasting
Title of week 2 paper is Finding COVID-19 from Chest X-rays using Deep Learning on a Small Dataset
Cluster 15
week2_diagnostics
Title of week 2 paper is Identification of a new coronavirus
Cluster 9
week2_phylogenetic_analysis
```

Figure 8.5: Clustering and identifying cluster of week2papers

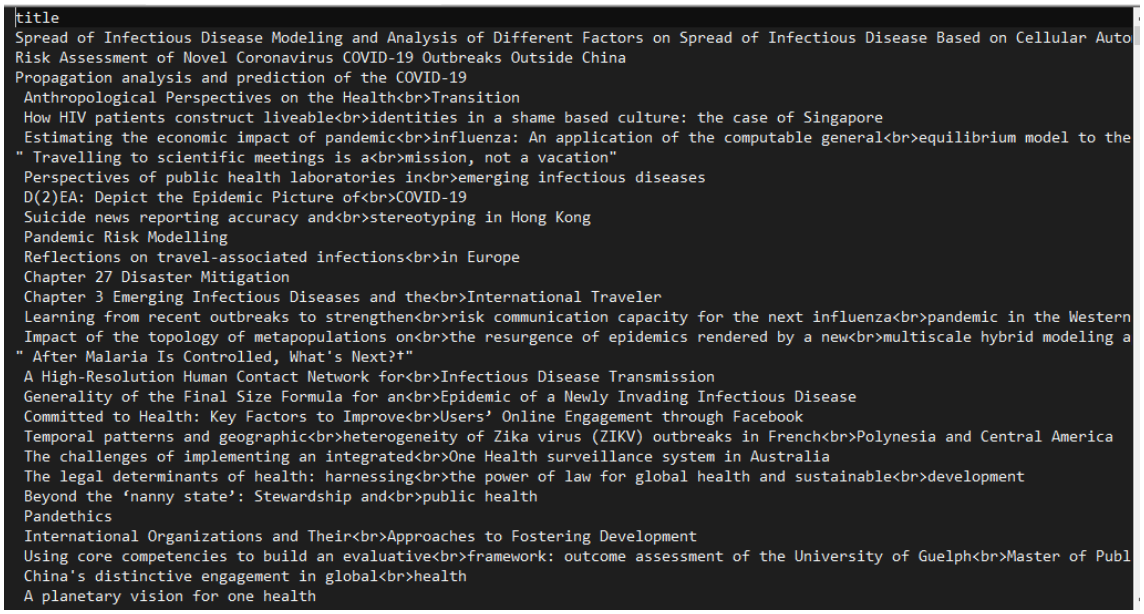

Figure 8.6: Cluster 6: titles of related papers to the papers from week2 about risk factor analysis, spreading and forecasting

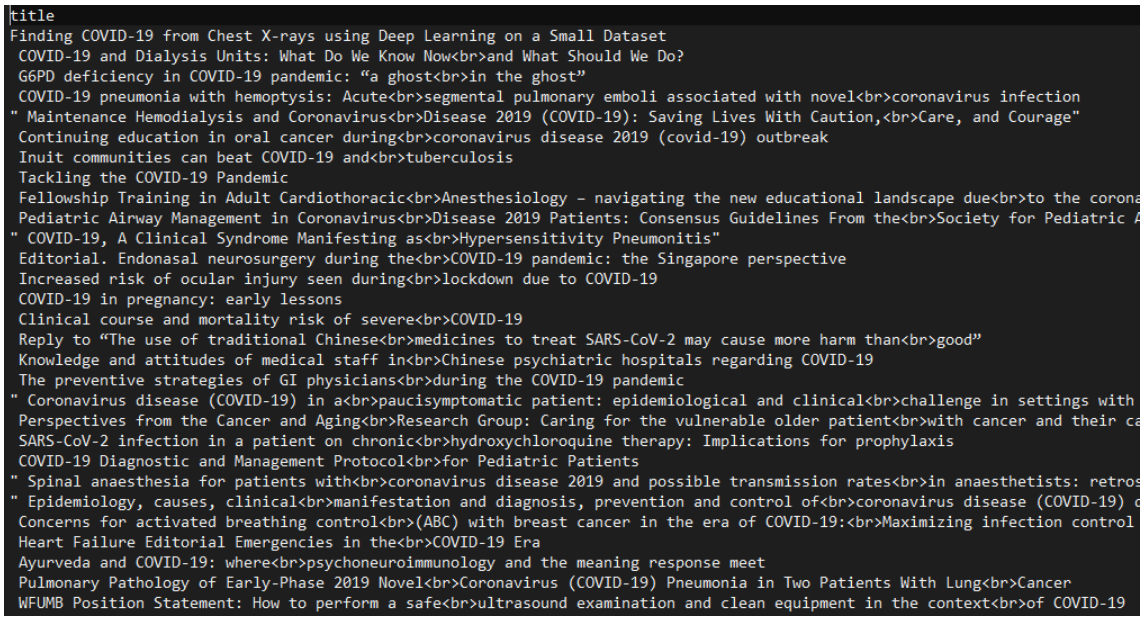

Figure 8.7: Cluster 15: titles of related papers to the paper from week 2 about diagnostics

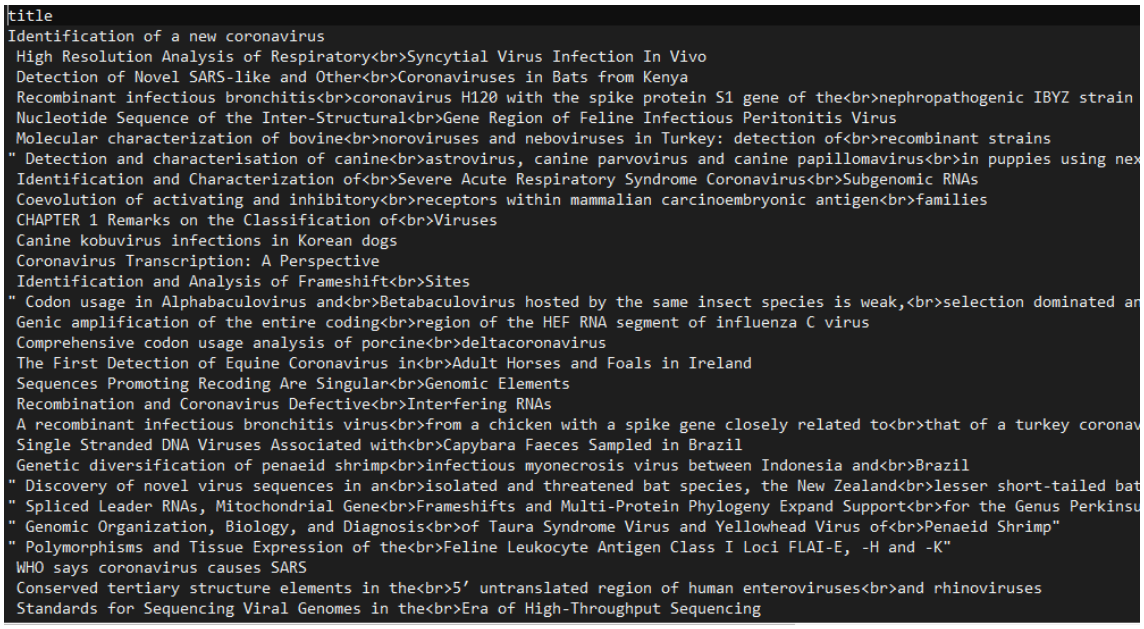

Figure 8.8: Cluster 9: titles of related papers to the paper from week 2 about origin analysis

As it can be seen the titles of the articles in the identified clusters are relates to the papers from *Part* I. Therefore, it can be a helpful approach to find a related papers.

8.3 Part3

8.3.1 Loading in the Student Dataset

After adding only the information about the five papers that we have chosen in *Part* I to the CORD-19 dataset, we created a new dataset containing all submitted papers by the course participants. It turned out that opening each of the 60.000 json files in the CORD-19 dataset to filter those jsons that do not match one of the submitted articles is too time-consuming. Title names have been stripped since traveling whitespaces result in missmatches. The runtime was dramatically decreased by first joining the course dataset with the metadata. The included paper id ("sha") was then used to search in the file names of the jsons for the papers of interest. This leads to a runtime reduction from several hours to less than 3 minutes. We matched 177 of the 195 submitted articles. And after removing duplicated entries our new dataset based on the submissions of the course participants contains 146 papers.

8.3.2 Preprocessing, Clustering and Results

Next, we proceeded with the preprocessing and final clustering step. For the preprocessing, we followed the previously describes pipeline of the kaggle notebook closely. We removed common stopwords as they act as noise. Next, a vectorizer with a noise filer of 2^{12} is applied, counting words and scoring less frequent words higher. Afterwards, the dimension of the dataset are reduced by PCA from over 4069 to 119. For the clustering we deviated from the kaggle notebook. First we tested a second method to determine the best k value. This was done by computing the silhouette score (Figure 8.9). Second we added another preprocessing step, to transform the data to a unit vector of 1, thus using the equivalent of cosine similarity as distance metric. After carefully comparing both methods with and without cosine similarity, one of elbow distortions and the other of silhouette scoring, we determine k by silhouette scoring and euclidean to be the best method (Figure 8.10). Thus we proceeded with k of 18. We choose 18, because it showed a noticeable bump in scoring, but is still a reasonable estimate based on the chosen paper data set by students. After running a t-SNE visualisation (Figure 8.11) and a Keyword extraction per cluster using LDA, with a minimum number constraint of 10 topics per cluster, we found 10 distinct clusters with an overarching topic. 8 clusters were removed due to insufficient number of data points. The results can be found in the Table 8.2.

The assignment of topics to each unique cluster was surprisingly easy, indicating a meaningful result. Some problems could be identified, for one some clusters had not enough articles assigned to them. Thus we could stipulate that the chosen k value of 18 is too high, rerunning the clustering at a lower value might give better results. On the contrary some clusters with high assignment could be identified (cluster 11). This could mean two things: Not much variation in the chosen cluster topics from the students or a more granular clustering is needed.

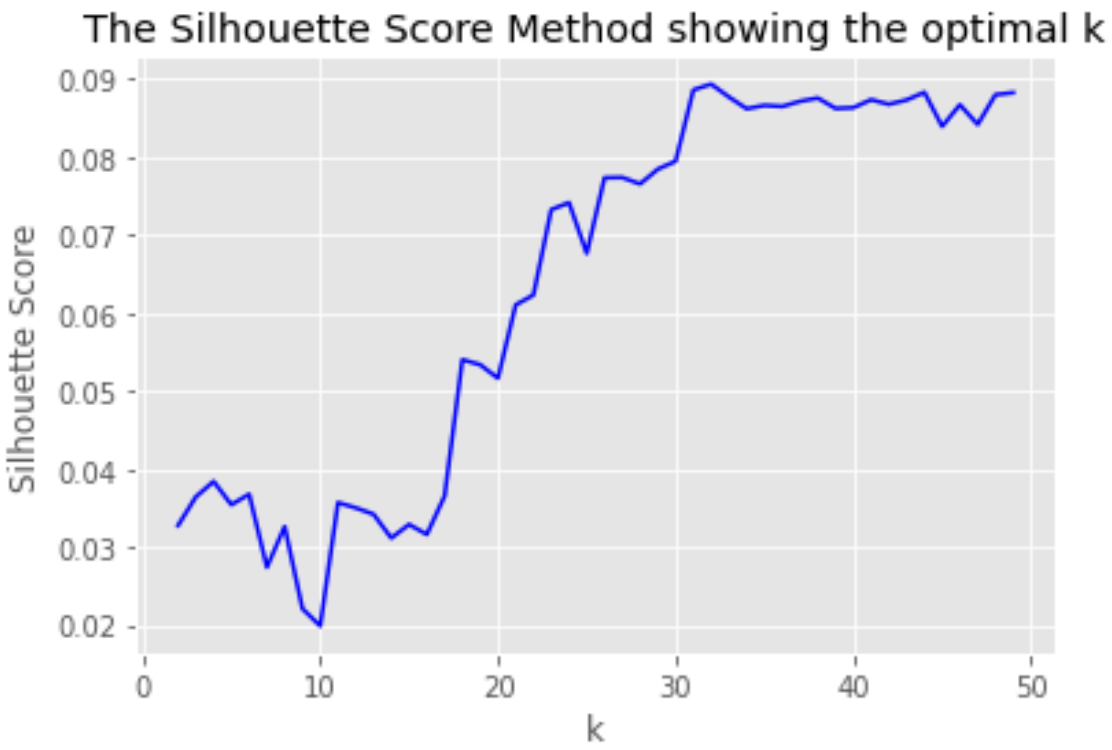

Figure 8.9: Silhouette scoring method for k-means. At the cluster point 17 a significant bump in scoring can be seen.

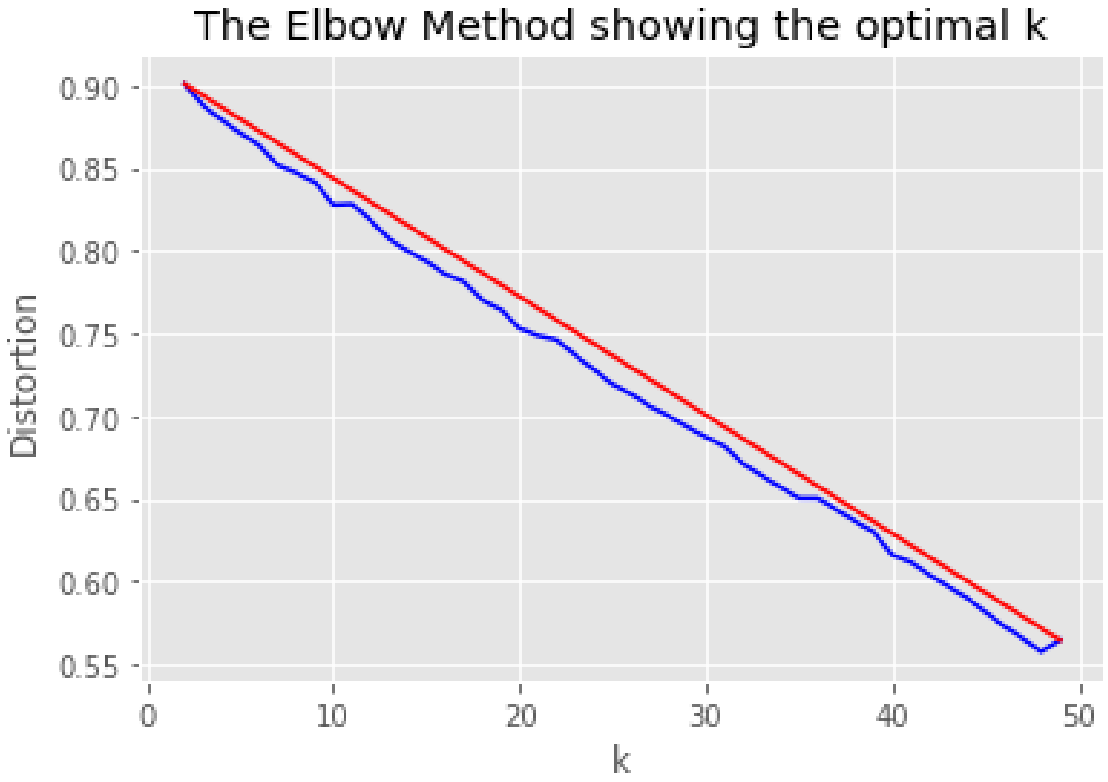

Figure 8.10: Distortion scoring method for k-means. Due to the low difference in scoring an almost linear line is produced. Thus in this case the method is unsuited to determine the optimal k for clustering

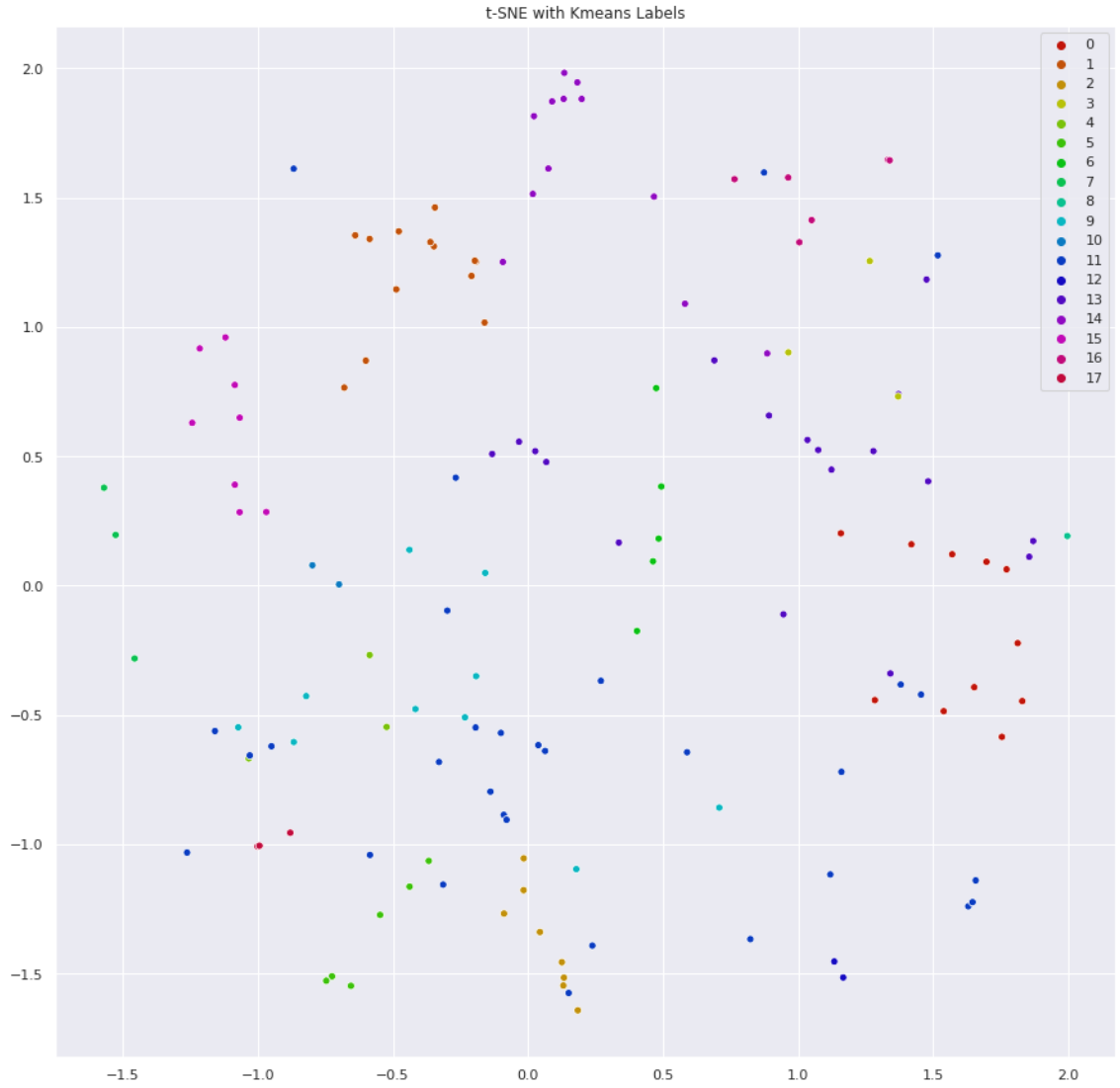

Figure 8.11: 2 dimensional tsne visualisation of the data set. Due to the low presence of articles the projection seems to be sparse. Still some clusters can be determined like: Top middle cluster 14: virus detection or middle left cluster 15: compartmentalized models. Big clusters like 11: modeling of spread are not clumped together.

Cluster	No. Articles	Assigned Topic	Keywords
0	11	phylogenetics	'sars', 'set', 'gene', 'rate', 'orf', 'recombination', 'datum', 'frequency', 'region', 'distance'
1	13	study of initial outbreak	'symptom', 'hospital', 'sars-cov-', 'china', 'country', 'mortality', 'use', 'respiratory', 'risk', 'evidence'
2	8	network-modelling of spread	'community', 'degree', 'threshold', 'mix', 'heterogeneity', 'group', 'node', 'use', 'transmission', 'size'
5	6	network-modelling of spread	'mix', 'wave', 'estimate', 'outbreak', 'total', 'community', 'human', 'delay', 'overall', 'additional'
9	11	disease forecasting	'case', 'disease', 'outbreak', 'estimate', 'influenza', 'process', 'interval', 'forecast', 'day', 'peak'
11	31	modelling of spread	'parameter', 'sequence', 'disease', 'method', 'change', 'city', 'spread', 'network', 'value', 'interaction'
12	18	origin detection	'case', 'camel', 'sars-cov-', 'human', 'protein', 'isolate', 'healthcare', 'bat', 'viral', 'sars-cov'
14	12	virus detection	'sars', 'sample', 'lung', 'serum', 'finding', 'detection', 'virus', 'care', 'study', 'swab'
15	8	compartmentalized modelling	'rate', 'risk', 'model', 'outbreak', 'death', 'datum', 'day', 'virus', 'state', 'patient'
16	6	diagnostics	'pcr', 'rsv', 'sequence', 'pneumonia', 'age', 'child', 'rhinovirus', 'associate', 'young', 'presence'

Table 8.2: Results of the clustering with unique topic assignment based on the first 10 keywords. 8 clusters are omitted due to low assignment of articles.

8.3.3 Creating a word cloud

Finally, we generated a basic word cloud. Here, we took the extracted keywords from the final clustering and used the word cloud package. The package provides a basic understanding of the word cloud with the use of some simple python libraries like numpy, pandas, matplot and pillow. The figure 8.12 shows the visualized wordcloud. Words like sars, virus and cov are bold and large which shows that the frequency of usage of these words is high and denotes their importance of it during the recent times. While these wordclouds are easy to look at, not much useful information can be extracted.

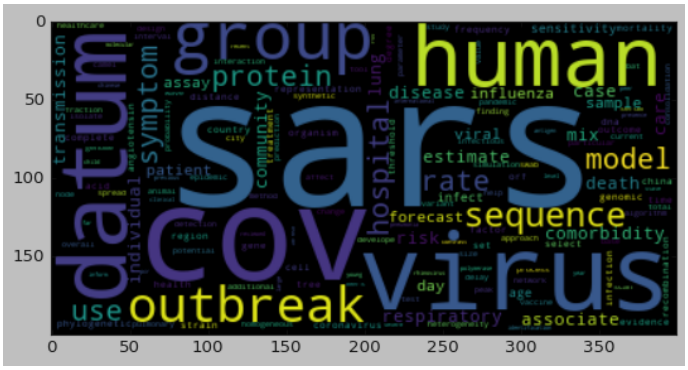

Figure 8.12: Word cloud of most frequent words.

Part 3

9. Introduction

9.1 Background

Dating back to the 1920's [10] for its first inception, a classical approach towards modeling the spread of diseases in epidemiology are SIR-models. SIR-Models are based on the idea of compartmentalization, where the dynamics of an epidemic are studied by dividing the populations into distinct subgroups. The name SIR is an abbreviation for its most simple form: **S** standing for **s**usceptible (i.e. individuals not yet infected), **I** standing for **i**nfectious (i.e. infected and infectious individuals) and **R** standing for **r**ecovered (i.e. individuals which are not infected and infectious anymore). Each compartment can be understood as a state, with a flow from one state to another. By using an equation of the simple form $N = S + I + R$ the whole population N stays static, while the ratios between the states change. Each state can than be modeled by differing differential equations, thus describing the fluctuations of each state at different timesteps t. Furthermore it is possible to freely add compartments by branching out from current ones, thus making the model adaptable to very different scenarios of an epidemic.

9.2 Goal of the Project

The objective of this weeks project is the application of a SIR-model on current Covid-19 case data taken either from a city (e.g. Berlin) or national (e.g. Germany) scale. The model itself is extended beyond the simple case by integrating two new states (Exposed, Dead) to the model and studying the impact of independent features (ICUbed-capacity, Age, Smoking and Gender) on the epidemic. By fitting the model to actual case data, possible projections can be made. Furthermore different scenarios such as lockdown, reducing social contacts and wearing masks, are explored by simulating their effect on the fitted model. Each prevention method is simulated over different periods and in combination with and without wearing a mask on top.

9.3 Outcome

A simple SIR model was implemented while exploring differences in rate of infection and time to recovery. Extending the model with the independent features age, smoking and gender significantly

altered the α values (i.e. rate of death) with smoking increasing the factor by more then double from 0.07 to 0.16. Two compartments were added, simulating the incubation period and extending the recovered individuals with death. A simulation with ICU bed capacity was performed, reaching the cap after 50 days. The fitting of the model to the the data of Germany resulted in some realistic numbers as the range R0 values was kept within the possible range and showed real declined behaviour. Other predicted curves for susceptible number, exposed, dead and recovery number were also kept within the realistic bounds. The performed simulations suggests that both the duration as well as the intensity of the restrictions plays an important role when fighting the outbreak of corona. The usage of masks is even more important for minor social reduction scenarios.

10. Tasks

10.1 A simple SIR model

The simple SIR-model includes three subgroups: ***S****usceptible*, ***I****nfectious* and ***R****ecovered* cases.

Figure 10.1: A flow diagramm showing the state transtions between the subgroups. The whole population is constant, while the flow is unidirectional. (Taken from [16])

Because each compartment can be understood as a state, we can visualize their transitions as a flow diagram (Figure 10.1). The variables above the state transitions describe the rates of individuals switching between the different compartments. For susceptible people becoming infected we introduce the factor β denoting the rate of one person infecting another person and for infected people becoming recovered we introduce γ denoting the rate of infected people developing immunity any given day. Thus we can infer three differential equations for each different subgroup, with N denoting the whole population:

$$dS/dt = -\beta * S * \frac{I}{N}$$
$$dI/dt = \beta * S * \frac{I}{N} - \gamma * I$$
$$dR/dt = \gamma * I$$

By integrating these equations over the time point t using the *odeint* function from the *sklearn* package in python3, we can develop a model simulation of the developing compartments for any initial starting conditions. As an example, we can compare the impact of tripling the rate of infection by using a β value of 1.0 compared to 3.0 (Figure 10.2 and 10.3). The β value of 3.0 shows a drastic change. All three compartments are shifted to the left, while the curve of infectious people has a much higher and steeper initial incline, which in turn results in a fast drop of susceptible people. In contrast reducing the γ value from $\frac{1}{4}$ to $\frac{1}{8}$ (Figure 10.4), results in a much lower incline of recovered people and much longer period of infectious people, as shown by the higher maximum of the yellow line. This shows the importance of both, the β and γ value. Thus, it is of high value to determine the ratio $\frac{\beta}{\gamma}$ denoted as R_0 to study the dynamics of a developing epidemic.

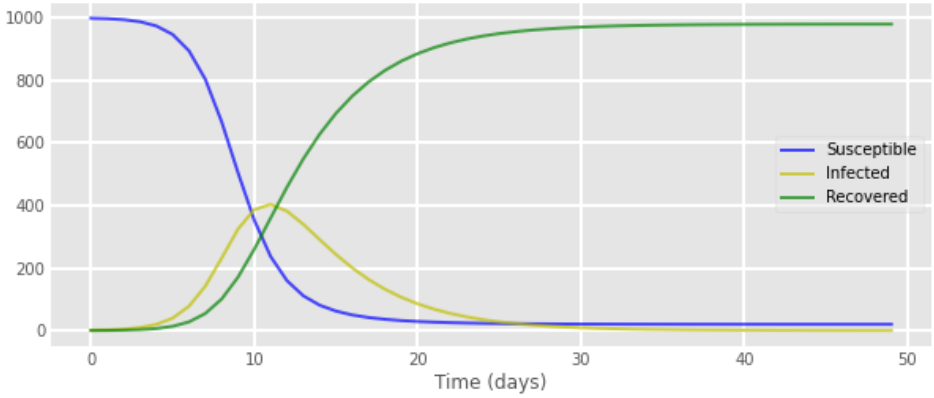

Figure 10.2: Basic SIR model simulation with starting values of S:999, I:1, R:0, β:1.0 and γ:1/4.

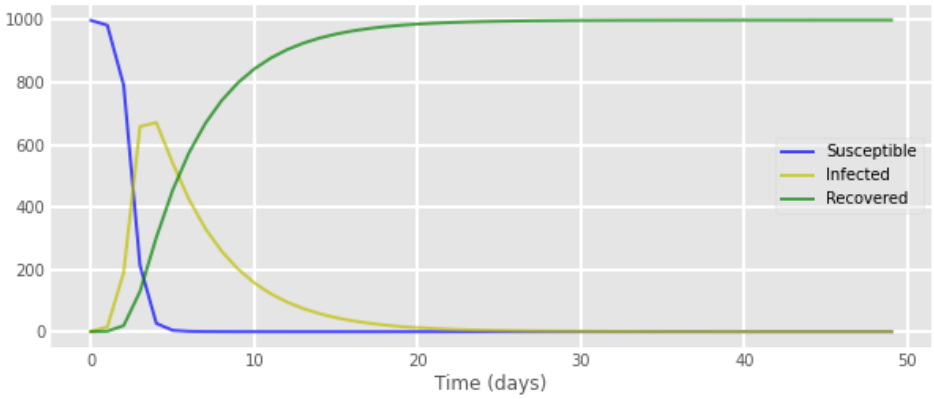

Figure 10.3: Basic SIR model simulation with starting values of S:999, I:1, R:0, β:3.0 and γ:1/4.

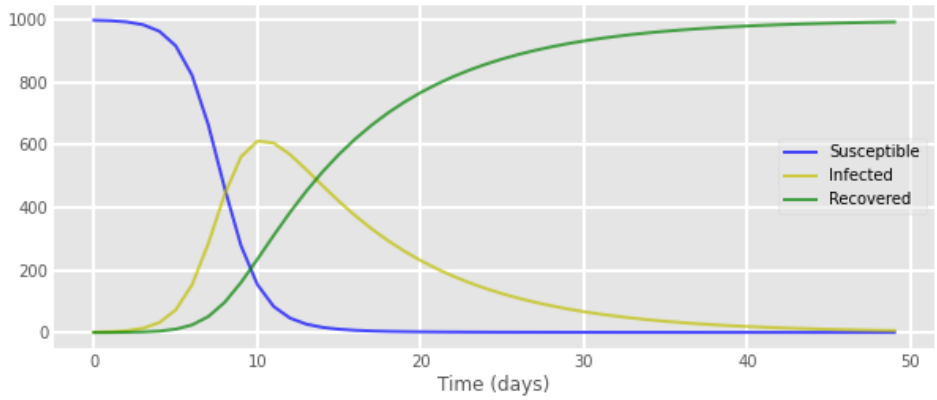

Figure 10.4: Basic SIR model simulation with starting values of S:999, I:1, R:0, β:1.0 and γ:1/8.

10.2 Extending the SIR model

The next step was to extend the basic SIR model with 2 new compartments (Figure 10.5).

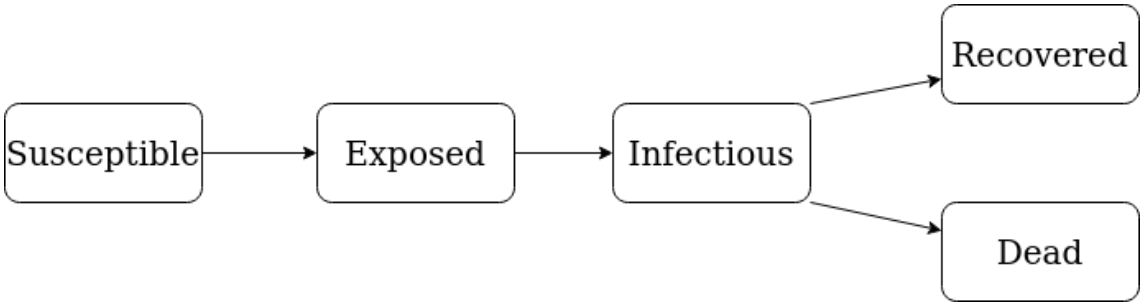

Figure 10.5: Flow diagram of the extended SIR-model. Two new compartments are added: Exposed and Dead.

Between susceptible and infectious people the exposed state is introduced. Exposed individuals carry the virus with an incubation period factor δ but are not infectious. Furthermore, the infectious group now branches out into recovered and dead. Thus, a death rate factor α is introduced to simulate the chance of death for infectious people, while also introducing a factor ρ for the length of time until death. It follows that the differential equations had to be altered:

$$dS/dt = -\beta * S * \frac{I}{N}$$
$$dE/dt = \beta * S * \frac{I}{N} - \delta * E$$
$$dI/dt = \delta * E - (1-\alpha) * \gamma * I - \alpha * \rho * I$$
$$dR/dt = (1-\alpha) * \gamma * I$$
$$dD/dt = \alpha * \rho * I$$

At next, the influence of different population proportions on the fatality rate factor α are introduced to the model. Since the age of infected people has an impact on the severity of the disease and the death rate [22], we created four age groups: 0-29, 30-59, 60-89 and 89+. Based on the death rate calculations by age groups in Italy [6], we assigned differing α values to each age group and added the percentages of the age group distribution in Germany (Table 10.1). The resulting α value for the entire population was 0.07726. Another factor that has a considerable effect on COVID-19 outcomes is the smoking behavior. We used an RKI report

about the prevalence of smoking in the adult population of Germany from 2013 [12] to integrate the proportion of daily smokers for each age group (Table 10.2). Abrams et al. [1] analyzed that current or former smokers have an increased COVID-19-related mortality by 2.4 [95% CI 1.43–4.04]. Therefore, we multiplied the α values of the smoking people in each age group by this value. This effected the overall α value to be increased to 0.16389.

Age group	% in Germany	α
0-29	30.1	0.001
30-59	41.51	0.013
60-89	28.13	0.2267
89+	0.27	0.285

Table 10.1: Alpha values assigned to different age groups.

Age group	smokers	non-smokers
0-29	31.95	68.05
30-59	26.375	73.625
60-89	8.45	91.55
89+	-	100.0

Table 10.2: Proportion of smoking in the adult population of Germany.

The third and last population proportion, we added to our model was the gender information. Zhang et al. analyzed potential risk factors in a study of n=663 Covid-19 patients and identified that male patients have an odds ratio of 0.486 [95% CI 0.311–0.758] to unimprove from the disease. We integrated this information to our model combined with the proportion of males and females in Germany from 2018 [4],[3]. Since there are 50,7% females and 49.3% males, α was slightly reduced to 0.16383. The final simulation model can be seen in Figure 10.6.

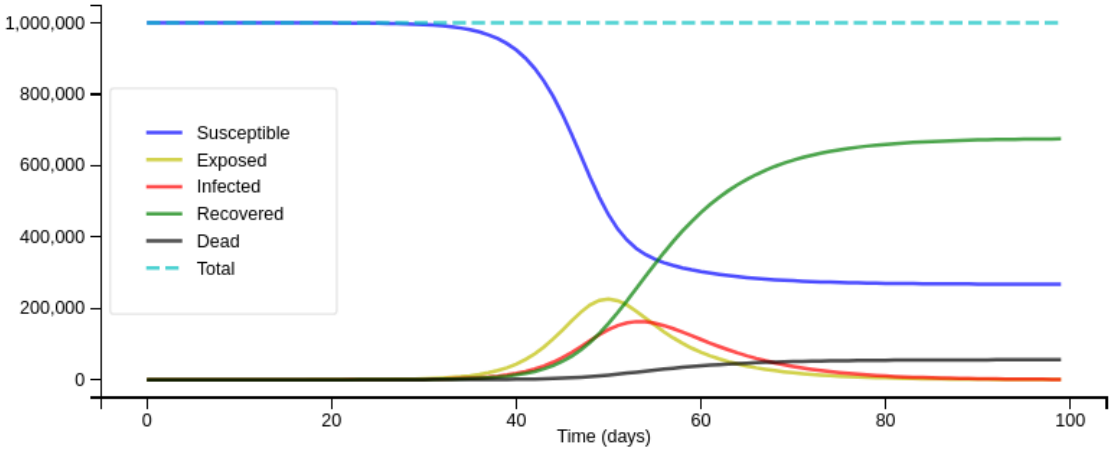

Figure 10.6: Resulting SIR model simulation with adjusted $\alpha value$.

The get the information how many ICU beds are in use at each time point of the simulation, we created an equation based on two information: 17% of patients infected with Covid-19 need hospitalization in Germany [7] and 48% of the hospitalized patients need ventilation [15] and

therefore an ICU bed. The resulting equation is $occupied\,ICU\,beds = I * 0.17 * 0.48$. When the capacity of ICU beds is exceeded the death rate in our model is changed to 0.6 (Figure 10.7).

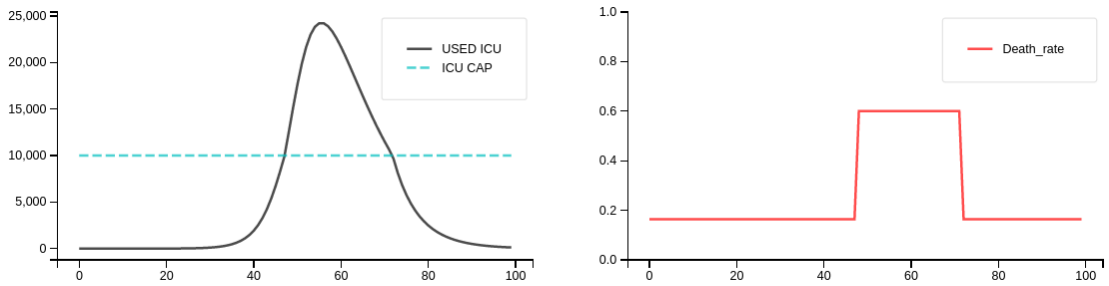

Figure 10.7: The left plot shows the occupied ICU beds (black) and the total amount of ICU beds (blue, dashed) while the right plot shows the corresponding death rate (red).

parameter	description	value
α	fatality rate	0.16
β	expected amount of people an infected person infects per day	1.25
γ	proportion of infected recovering per day	1/10 [15]
δ	incubation period (1/days)	1/5 [15]
$inf\_to\_dead\_d$	days from infection until death	50 [15]

Table 10.3: Initial parameter setup.

10.3 Parameter fitting

The next part deals with fitting the extended SIR model with time-dependent R_0 values and resource-dependent death rates to real Corona virus data of Germany, in order to come as close as possible to the real numbers and make informed predictions about possible future developments. The data we used is the data which contains the information about age groups and other parameters like fatality rates, R0 values, beginning of lockdown, etc in Germany and the data was parsed according to the range of events. The cases from 01.03.2020 – 30.04.2020 were only considered for fitting to our model to get the course idea.

Here, we initially loaded the data for the age groups, probabilities,created some look up dictionaries for easy access of the data parameters. We mainly focused on the the probabilities of infected to death. The equations of the model are translated to the coding with R_0-function and the whole model that takes the parameters to fit to calculate the curves of S, E, I, R, and D. Finally, for curve fitting we first set the parameters we knew and assumed with the upper and the lower bound values with unknown data, and defining the x values for the number of days to get the future predictions with the parameters fit.

The resulting simulations showed quite similar values with the real data with R_0 as 2.08 at the start of march 2020 and R_0 as 0.48 at the end of april, a death rate of 0.16. It is quite comparable with the real data and the many points are inline with the real data points indicating the best fit which is shown in the figure 10.9. So with the outbreak beginning on 21st January and the outbreak shift set to 30 days, so our model thinks that the main lock down took place in Germany nearly after 107 days i.e roughly in the middle of March which is very close to the real data. Figure 10.7 represents the prediction model, Here's the prediction in April — if the model is right, Germany has gone through the worst already as deaths per day is increasing which should decrease strongly over the

next months. R_0 will stay in the range, if it goes up again as lock downs are reverted, the numbers will start increasing again.

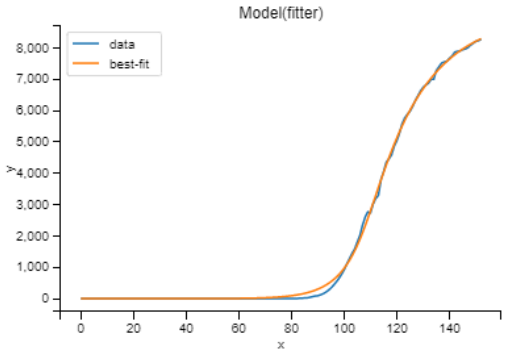

Figure 10.8: Model fit for Germany

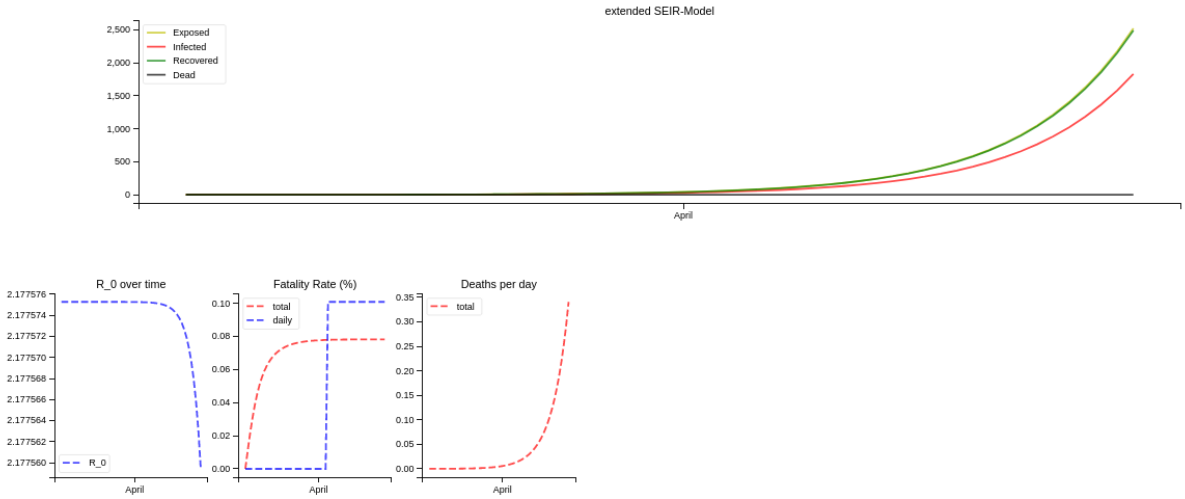

Figure 10.9: Prediction for Germany with suitable parameters

10.4 Scenario Studies

In the last step, the fitted model is used to simulate the spread of the virus with various prevention methods being implemented. Those methods affect the spread of the virus by reducing the expected amount of people an infected person infects per day (β). Recall in the equations in section 10.2, where the parameter β is obviously only present in the equations for the number of susceptible and the number of exposed people. The following prevention have been implemented to reduce β:

Reducing Social Contacts

Maybe the most effective way of decreasing the spread of the virus is to limit social contacting. A short and lose restriction period increases the risk of an uncontrolled spread of the virus or the arise of a second wave which will lead to many deaths. But a too long and tight restriction can lead to economic and psychological incisions. Performing simulations can help to see the effect of social distancing with varying periods and intensities. For our simulation we implemented social distancing by reducing β by 0/25/50/75%.

Lockdown

The so-called lockdown is a special case of social reduction where social contact is reduced to an absolute minimum by restricting the population leaving their house. But even for the case of a lockdown, peoples' social contacts cannot be stopped completely because a portion of the population like cashiers or hospital staff needs to go to work. Therefore we modelled the lockdown as a reduction of β by 90%.

Masks

Several studies [17][19] state that wearing masks can effectively reduce the spread of the virus by 8%-16%. Each of the social distancing simulations have been performed with and without the usage of masks. Wearing a mask is modelled as an additional reduction of β by 12%.

10.4.1 Methods

After a fixed unrestricted time of 30 days the restriction period starts, followed by 100 simulation days, where the length of the restriction within the simulation days is varied by 0/25/50//100% of the simulation days. As before we initialized our model with one individual being infected while the rest of the population is susceptible.

10.4.2 Results

From Figure 10.10 we can make several observations:

- With no restrictions (plot [1,1]) implemented the amount of infected people is with is by far the highest and the number of dead people is breaching linear growth
- Even minor restrictions for the smallest period reduce the number of infected people noticeably
- Wearing masks has absolute as well as relatively a higher impact when combined with less strict social restrictions
- Increasing the time of social contact reduction has less impact than the the intensity of social distancing
- The difference between 50% and 100% restricted simulated days is just minor when combined with masks and lockdown

The parameters that has not been fitted to German data have been set as shown in Table 10.3.

10.4.3 Discussion

The performed simulations suggests that both the duration as well as the intensity of the restrictions plays an important role when fighting the outbreak of corona. The usage of masks is even more important for minor social reduction scenarios. Nevertheless, the simulations seem to underestimate

the true case. The number of infected people might be underestimated. Recall, the model was fitted to Germany with the data from beginning of March to the end of April. At this point the government of Germany already introduced several restrictions to keep the spread of the virus under control, such like advising people to stay home, closing public locations and shifting many jobs to the home office. Thus, our data is already fitted to restrictions and then used to simulate restrictions, which doubles the effect of the implemented restrictions in the different scenarios. Surprisingly, without any simulated restrictions and with a model that has been fitted to data of period where restrictions were present in Germany, the model still simulates more than 10 million infected people within 130 days.

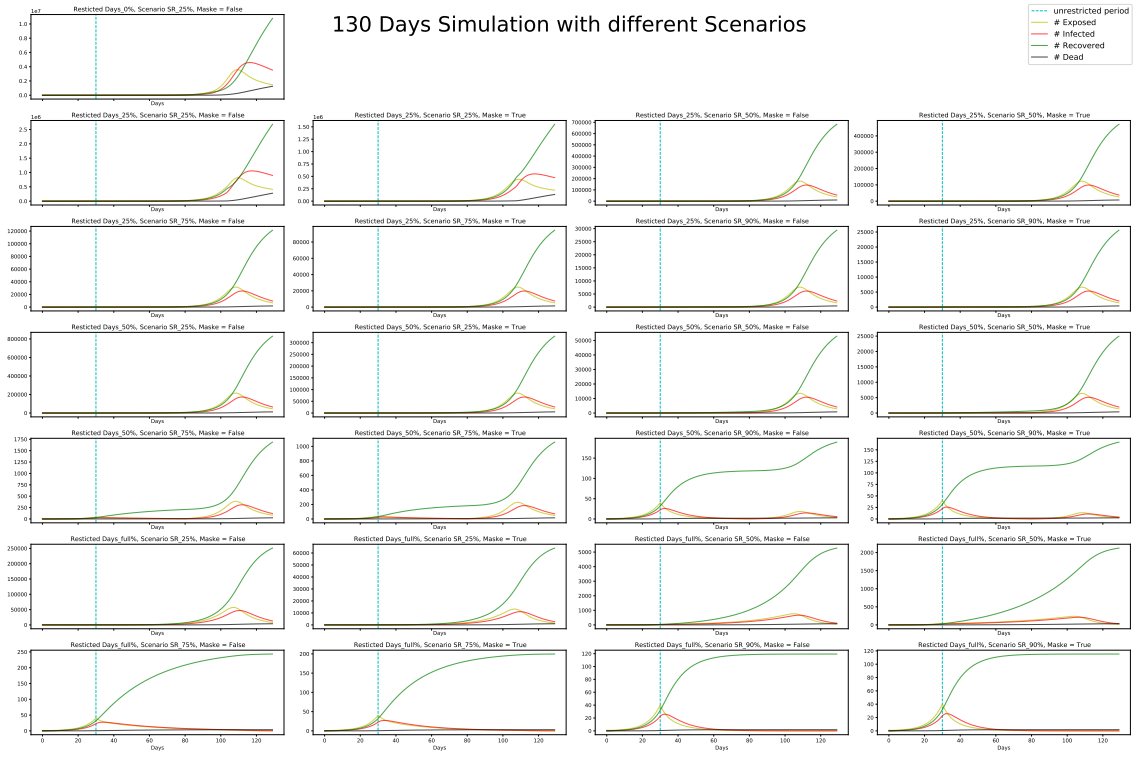

Figure 10.10: Each plot represents an independent simulation, starting with no restrictions (top left). The duration and intensity of the restrictions is varied through the simulations and once combined with the usage of a mask. The time without restrictions is denoted by the blue vertical line. Note that the number of susceptible people is not shown and the number of people in general (y-axis) is not normalized to improve visibility.

Bibliography

Articles

[1] Elissa M Abrams and Stanley J Szefler. “COVID-19 and the impact of social determinants of health”. In: *The Lancet Respiratory Medicine* (2020) (cited on page 56).

[2] Kristian G Andersen et al. “The proximal origin of SARS-CoV-2”. In: *Nature medicine* 26.4 (2020), pages 450–452 (cited on pages 27, 28).

[8] Kuldeep Dhama et al. “SARS-CoV-2: Jumping the species barrier, lessons from SARS and MERS, its zoonotic spillover, transmission to humans, preventive and control measures and recent developments to counter this pandemic virus”. In: (2020) (cited on page 27).

[9] Eneida L Hatcher et al. “Virus Variation Resource–improved response to emergent viral outbreaks”. In: *Nucleic acids research* 45.D1 (2017), pages D482–D490 (cited on page 27).

[11] Tommy Tsan-Yuk Lam et al. “Identifying SARS-CoV-2 related coronaviruses in Malayan pangolins”. In: *Nature* (2020), pages 1–6 (cited on page 28).

[12] Thomas Lampert, Elena von der Lippe, and Stephan Müters. “Prevalence of smoking in the adult population of Germany”. In: (2013) (cited on page 56).

[13] Zhixin Liu et al. “Composition and divergence of coronavirus spike proteins and host ACE2 receptors predict potential intermediate hosts of SARS-CoV-2”. In: *Journal of medical virology* 92.6 (2020), pages 595–601 (cited on page 28).

[14] Fábio Madeira et al. “The EMBL-EBI search and sequence analysis tools APIs in 2019”. In: *Nucleic acids research* 47.W1 (2019), W636–W641 (cited on page 27).

[17] Thorsten Suess et al. “The role of facemasks and hand hygiene in the prevention of influenza transmission in households: results from a cluster randomised trial; Berlin, Germany, 2009-2011”. In: *BMC infectious diseases* 12.1 (2012), page 26 (cited on page 59).

[19] Samantha M Tracht, Sara Y Del Valle, and James M Hyman. “Mathematical modeling of the effectiveness of facemasks in reducing the spread of novel influenza A (H1N1)”. In: *PloS one* 5.2 (2010) (cited on page 59).

[21] Fan Wu et al. “A new coronavirus associated with human respiratory disease in China”. In: *Nature* 579.7798 (2020), pages 265–269 (cited on page 28).

[22] Jin-jin Zhang et al. “Clinical characteristics of 140 patients infected with SARS-CoV-2 in Wuhan, China”. In: *Allergy* (2020) (cited on page 55).

Books

[10] McKendrick Kermack. *A contribution to the mathematical theory of epidemics*. Proc. Roy. Soc. A, Band 115, 1927 (cited on page 51).

[18] Michel Tibayrenc. *Genetics and evolution of infectious diseases*. Elsevier, 2017 (cited on page 27).

Webpages

[3] *Bevölkerung - Zahl der männlichen Einwohner in Deutschland nach Altersgruppen am 31. Dezember 2018*. https://de.statista.com/statistik/daten/studie/1112607/umfrage/maennliche-bevoelkerung-in-deutschland-nach-altersgruppen/ (cited on page 56).

[4] *Bevölkerung - Zahl der weiblichen Einwohner in Deutschland nach Altersgruppen am 31. Dezember 2018*. https://de.statista.com/statistik/daten/studie/1112611/umfrage/weibliche-bevoelkerung-in-deutschland-nach-altersgruppen/ (cited on page 56).

[5] Simon Burgermeister. *covid_sequence*. https://github.com/simonjuleseric2/covid_sequence. 2020 (cited on page 27).

[6] *Coronavirus (COVID-19) death rate in Italy as of May 20, 2020, by age group*. https://www.statista.com/statistics/1106372/coronavirus-death-rate-by-age-group-italy/. Accessed: 2020-05-25 (cited on page 55).

[7] *Coronavirus Disease 2019 (COVID-19) Daily Situation Report of the Robert Koch Institute*. https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Situationsberichte/2020-04-29-en.pdf?__blob=publicationFile (cited on page 56).

[15] *SARS-CoV-2 Steckbrief zur Coronavirus-Krankheit-2019 (COVID-19)*. https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Steckbrief.html (cited on pages 56, 57).

[16] *SIR flow diagramm*. https://www.lewuathe.com/covid-19-dynamics-with-sir-model.html/. Accessed: 2020-05-24 (cited on page 53).

[20] *Worldometers Kernel Description*. https://www.worldometers.info/coronavirus/countries-where-coronavirus-has-spread/. Accessed: 2020-05-11 (cited on page 9).

Index