Mannheim Master of Applied Data Science & Measurement

Mannheim Business School

Examiner: Frauke Kreuter

Supervisor: Stefan Bender

Real-time analysis of predictors of COVID-19 infection spread in the countries of the European Union

User's Manual

Anikó Balogh, Anna Harman

Draft

May, 2021

GitHub code repository:

https://github.com/Annalilla/COVID

Table of Contents

Intro

This is a user's manual for the COVID-19 visualization app intended for a public of interest in Statistics and Machine Learning.

There are two ways to access our app: either open it on our temporary server at http://www.covidmdmmasterteam.tk:3838/ or download the data and run the visualization app on your own computer. Very basic knowledge of the R software is required for this latter desktop version.

1. What is this COVID-19 app good for?

The COVID-19 app[1] is an interactive visualization tool of COVID-19 related data for the countries of the European Union.

- You can *explore* the number of COVID-19 infections by countries along with many predictors of the infections like mask usage, direct contact, vaccination, average daily temperature, and restriction measures applied in the selected countries.
- You can *compare* the differences between *predictor importance* on COVID-19 new infections between countries on an interactive bump chart.
- You can check the effect of the predictors one by one per country on our model of COVID-19 new confirmed infections on partial dependence graphs.

In order to use our app, you can simply click here (server version) and the app opens straight away. You can also run the codes yourself (desktop version). In this case you have to download the data with the help of our automated data collection program and start the application. Before starting the application the data preparation and modelling will be performed automatically. All the steps are shown in the next chapters.

[1] You can find a detailed description about the structure and the operation of the application in the Appendix.

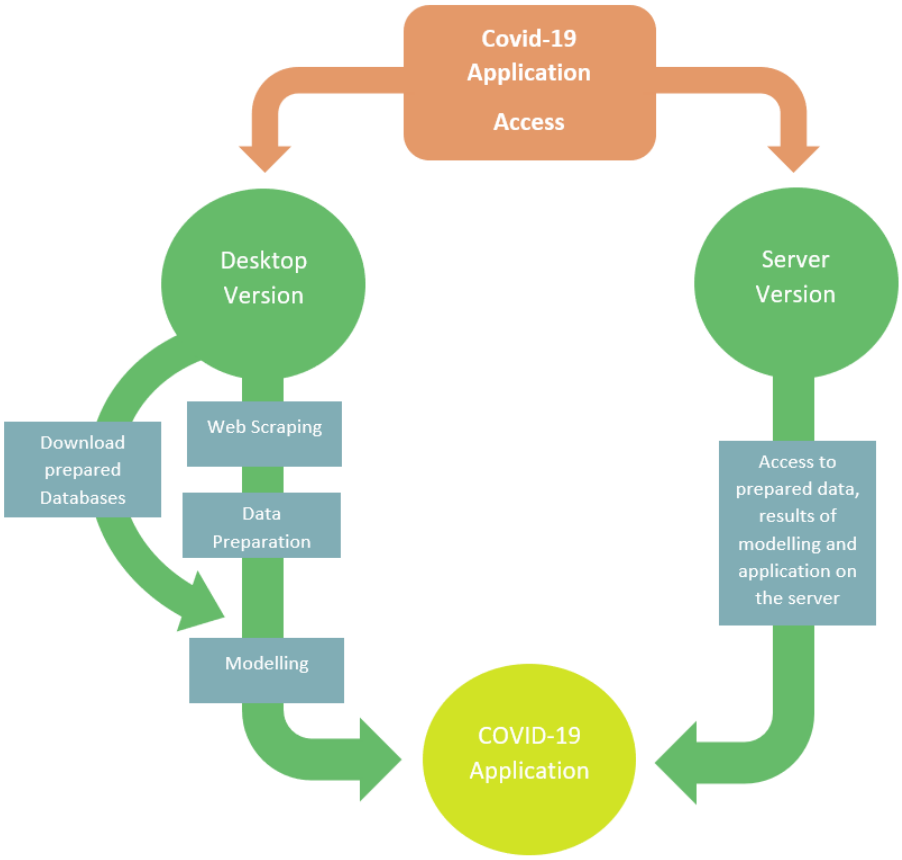

Figure x. Workflow of the Covid-19 application

2. What do you need to use this app?

For the server version, you only need

- a computer with internet connection (the app is not fully adapted to phone or tablet)
- Internet browser.

For the desktop version you need

- a computer with a stable internet connection,
- Internet browser,
- the R software installed on your computer.

If you decide to use the server version, you can start to explore the data and the results of the random forest models immediately. If you would like to use the desktop version, you have to run the codes

3. Where do you find the codes for the app?

You can reach all the codes commented in our GitHub repository, arranged in a clear folder structure.

For the server version, you do not need the codes, just click on this.

For the desktop version, if you would like to run our visualization app on your computer, you have to *download and unzip the codes* and run them locally on your computer.
Here is the screenshot of our GitHub repository. Click the green 'Code' button, then select 'Download ZIP'.

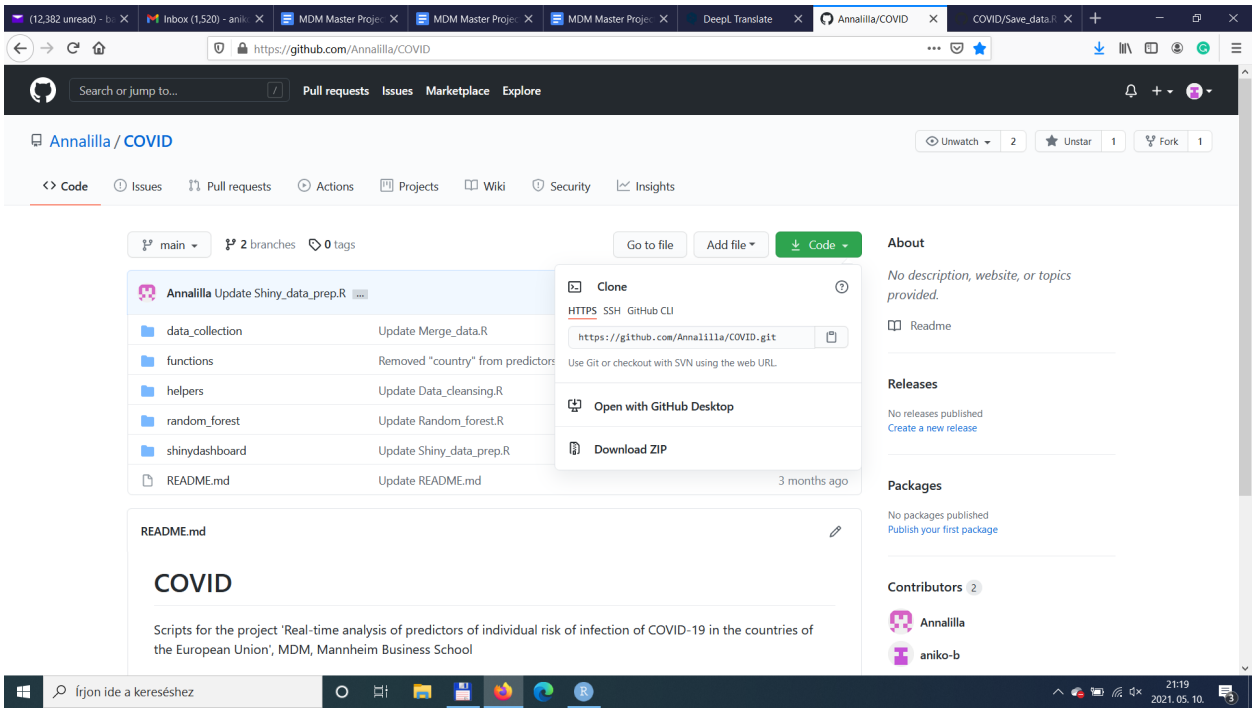

Figure x. Screenshot of our GitHub repository to access codes

After downloading, you should unzip the files.

4. How can you download our data for the app?

For the server version, you do not need the data, just click on this.

For the desktop version, you have now our codes unzipped on your computer. Before downloading the data, some preparatory steps are needed. Using the R software, you have to set your working directory in this code and then run it. Your working directory should be the directory where you put the codes from our GitHub repository like this: "path_where_you_saved_the_unzipped_repository/COVID-main".
After this, you can download, or better to say webscrape and merge the data by running the first three rows of this code.[2] Since the data will be scraped from many different sources, this process will take a while. As a result, you will get two databases, one for time-dependent variables and the other for time-constant variables. Save them locally on your computer by running the 4th and 5th rows of the code.

Also, a regularly updated version of these two databases is available on our password-protected google sheets account. This allows you to download the databases faster as you don't have to scrape the data from different sources. However, in order to use this simpler method you have to possess a key to our google sheets account. If you have a key you can save the downloaded data by running the 6th and 7th rows of the code in use, and download and prepare the two databases with this code.

5. How do you run the app?

For the server version, you do not need to run anything, just click on this.

[2] More details about the data collection and preparation can be found in the Appendix

You have to run the app on your own computer in order to use it. After you downloaded the data according to step 4, you should open this code in R Studio and run it simply by clicking the 'Run App' arrow in the top right corner of your script window. The app opens in a separate window.

6. How does this app work? What does it show to you?

For the server version, just click on this and the app opens in your browser.

For the desktop version, after running the app according to step 5, the app opens in a separate window.

On the left-hand side you can choose between the tabs:

6.1 Exploratory tab

The application starts with the Exploratory tab. On this tab, the number of infections is displayed per country for the whole time interval. You can select a country from a dropdown menu. You can choose between the smoothed and the unsmoothed version of the continuous variables (default is smoothed). Also, you can set the time interval and add additional variables to the chart (mask coverage, direct contact, vaccination, average daily temperature, and restriction measures applied in the selected country). Since the change in the number of new cases follows the change of the predictors with a delay, you can add a lead to the number of infections to make it better comparable with the other variables that can be added to the chart.

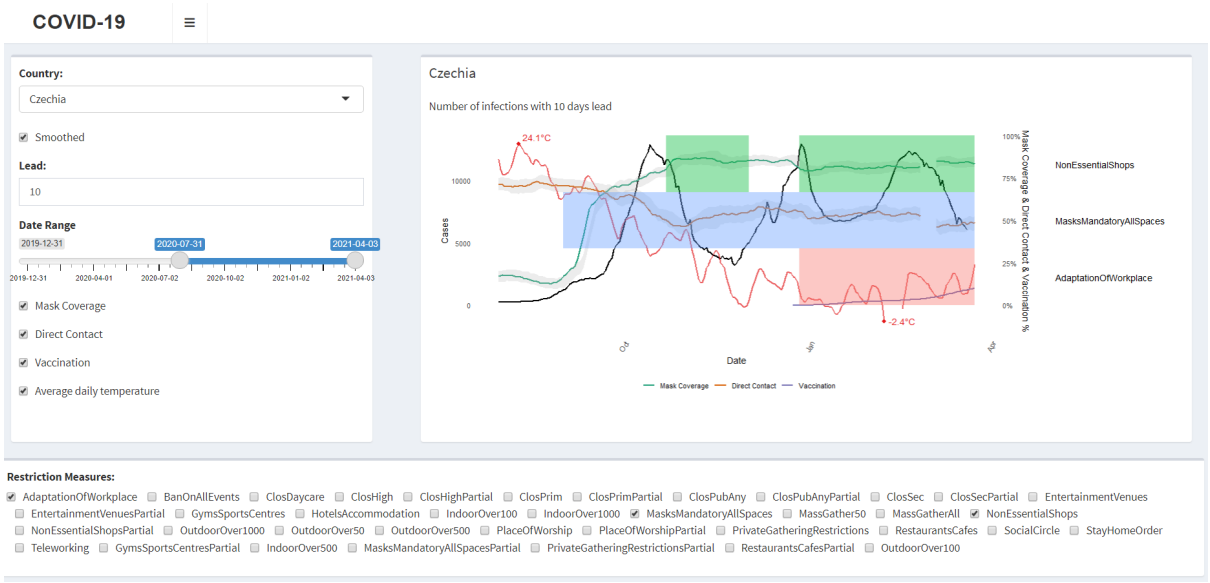

Figure X. Exploratory tab of the dashboard

Example interpretation

The picture above presents an example of the exploratory tab of the application. The number of infections in Czechia between 2020.07.31 and 2021.04.03 is shown on the black line with a 10-day lead. The continuous variables are smoothed. Mask coverage, direct contact, vaccination, and average daily temperature are added to the plot. The time intervals, in which the selected restriction measures were applied are also shown: the non-essential shops were closed in November and again since January 2021, the workplaces were adapted also since January, the usage of mask coverage was mandatory in all spaces from autumn 2020 until the end of the selected time interval.

6.2. Partial Dependence tab

This is the second tab on the left-hand side menu of our app. On this tab, you can see different versions of a partial dependence plot based on our Random Forest model predictions.

A partial dependence plot illustrates the functional relationship between a predictor and our prediction on new COVID-19 cases. It shows how the prediction partially depends on the values

of the predictors, it can also show the type of relationship, such as a step function, curvilinear, linear, and so on.

On this tab, you can select a country and any of the response measures from the relevant drop-down menus, and see the Partial Dependence Plot accordingly only by one.

Figure X. Partial Dependence Plot tab of the dashboard

Example interpretation

The plot below shows the relationship (according to the model that we trained) between the number of new COVID-19 infections and the average daily temperature in country x. Here, we see that the number of new COVID-19 infections increases as we increase the average daily temperature up to x. After that, it does not change the number of infections.

6.3 Bump Chart tab

Bump charts are often used to express changes in rank over time (R-bloggers 2018). However, instead of time, in this project, we use a bump chart for the visualization of differences in the rank of predictors by their variable importance[3] over countries.

On the right side of the tab, you can select predictors and countries for the visualization. Also, all predictors and countries can be selected or unselected with the action buttons under the chart. The predictors are ordered by their variable importance.

[3] The variable importance measures the contribution of a predictor in predicting the response. In the R caret package, the contribution of each predictor to the random forest regression is calculated in the following way: "The random forest algorithm estimates the importance of a variable by looking at how much prediction error increases when (OOB) data for that variable is permuted while all others are left unchanged. The necessary calculations are carried out tree by tree as the random forest is constructed." (Liaw 2002: 18).

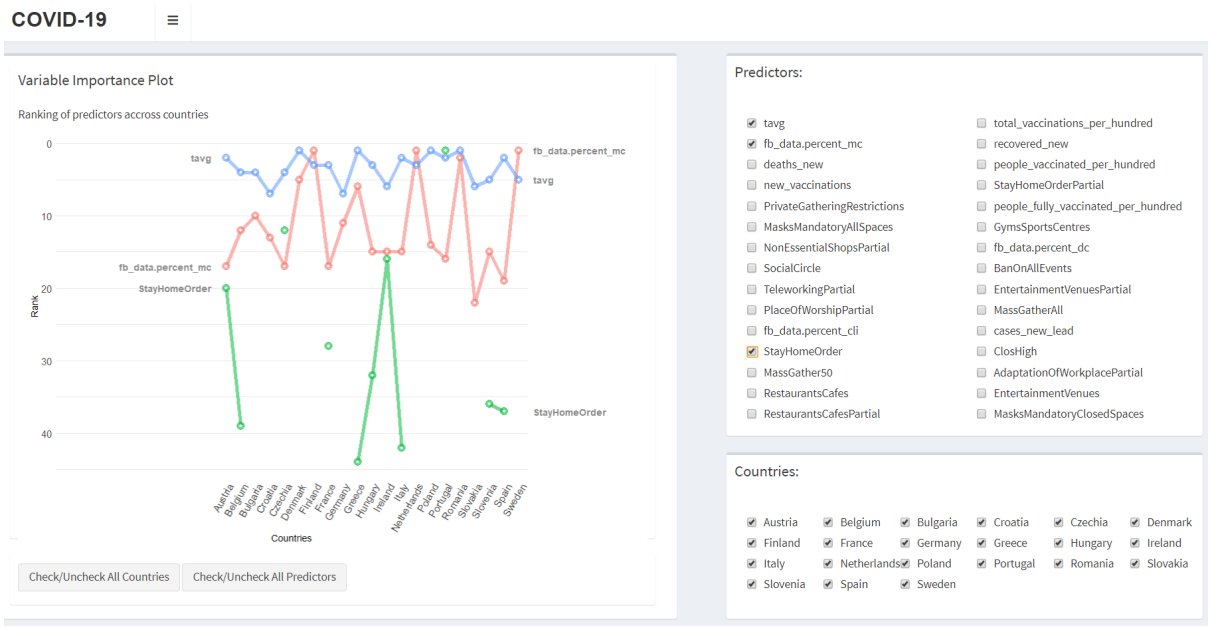

Figure X. Bump Chart tab of the dashboard

Example interpretation

All countries and the predictors average daily temperature, percentage of the population using mask coverage, and application of the restriction measure stay-at-home order for the general population (also referred as "lockdown") are selected. As we see, the variable importance of the same predictors can vary widely between the countries.

6.4 Predictors vs Country Characteristics Tab

On this tab you can compare the effect of predictors on the number of confirmed new COVID-19 cases in countries with different characteristics. The countries were assigned to groups with hierarchical clustering[4]. The predictors are ranked by their variable importance (variable with highest variable importance in a country has rank 1, variable with second highest variable importance got rank 2, and so on.). The similarity of the rank of the predictors in countries within clusters and the differences between clusters are visualized here.
***Visualization is not yet implemented

[4] More details about the hierarchical clustering can be found in the Appendix

6.5 Datasources tab

6.6 Documentation tab

7. How can you update our data?

For the server version, you just click on this and can see the version that we updated.

For the desktop version, if you do not have our data, first you have to download it according to step 4. If several days have passed since the last use of the app, you can *update* your data to get the latest available data to visualize in our app. This program updates all the different parts of the database with new records since the last download and saves it.
You should run this code. Based on the length of the update period and your internet connection this may take a while.

8. How can you revise if data providers changed the data between updates?

For the server version, you do not need to check any revisions, just click on this.

For the desktop version, if you have the data for a longer period, you may want to perform a revision to see if changes were applied on the earlier values of the data. You can do this with this code, after you updated your data. You can make comparisons between your updated and the freshly downloaded data and update the variables with the new values if you like.

9. Where do you find more details on our data and methodology?

Our data is collected and merged from various sources. You can find the list and description of available variables, the various methods of web scraping and the reference for the data sources in chapter x of the Appendix.

The Bump Chart and the Partial Dependence chart are based on a group of Random Forest models[5].

[5] You can learn more about the modeling process in the Appendix.

Appendix

List of appendices

1. Data collection and preparation
2. Modelling
   2.1. Times series methods revisited
   2.2. Model selection
   2.3 Our model
   2.4 Time serie cross-validation
   2.5. Future plans to improve our predictions
3. Shiny Dashboard - description
4. Data sources and Database Description
5. Code structure
6. Bibliography
7. List of figures
8. R packages

1. Data collection and preparation

For the automatization of the data collection we used different web scraping techniques. The data is collected from 7 different online sources with a starting date of 02.28.2020. In some cases APIs are available, in other cases the data is accessible for download as a csv file on the homepage of the providers.

The time-constant country-specific data is extracted from Eurostat, using the API provided by them. The data from the last year available for the given database is used for the analysis (2019 for population characteristics, 2018 for health expenditures and 2015 for cultural participation). For the population characteristics the values about the total number of population, the number of males and females, number of population in age groups 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-59, 60-64, 65-69, 70-74, 75-79, 80-84, above 75, above 80, above 85 and under 5 years per country are selected. To measure the health care expenditures in the analysed countries a variable for the total healthcare expenditure is created. Variables about cultural participation per country are also included in our analysis. The percentage of 16 years olds and older, under 30, above 75 years olds who didn't attend any cultural event in the last 12 months are selected for the final database.

The country-specific response measures are downloaded from the homepage of the European Centre for Disease Prevention and Control. The database is downloaded as a csv file with the data.table package of R. The link of this database changes from time to time, therefore it has to be extracted first from the html code of the homepage. This extraction is done with the rvest R package. The format of the downloaded database is not appropriate for the analysis. The database contains four variables: country, response measure, start and end date. After the download a variable is created for every response measure, which takes the value “1”, if the response measure was applied on the given day in the given country, and “0”, if not.

The data containing the number of new infections, deaths and recoveries related to COVID-19 are extracted from the homepage of the John Hopkins University using the coronavirus package. In some cases the number of new infections is negative. This can occur, if there is a change in the

counting methodology or data resource, the updating of the new cases happens another day than they were counted or if there are errors in the raw data (Krispin 2020).

The daily average temperatures for the analyzed time interval are extracted from the homepage of the National Centers for Environmental Information using the rnoaa R package. The downloaded daily average temperatures are measured on weather stations closest to the capitals of the countries. In some cases the nearest weather station is not functioning for a longer period. In these cases another station is selected manually from the next nearest stations for better data coverage on the time interval of the analysis.

The publicly available aggregated non-US data of the COVID-19 World Symptom Survey with the variables reported COVID-like illness symptoms, mask usage and direct contact is downloaded with it's open API. The databases containing the mentioned variables are downloaded separately for the analyzed time interval, and merged together at the end as a preparation for the analysis and visualization.

The data about new and total number of vaccinations and proportion of vaccinated people are downloaded as a csv file from the homepage of Our World in Data with the data.table R package. The missing values for the variables about vaccinations in the database occured before the first day with vaccination in the given country were replaced with 0, assuming that before this day there were no vaccinations there.

All variables were checked for implausible and missing values. The type of the variables were set to factor, numeric or date and labels were added.
All time-constant country-specific variables are merged together in one database, and all time-changing variables are merged together in another one.

Another data preparation step takes place to prepare the data for the visualisation. The moving averages of new infections, daily average temperature, percentage of vaccinated people, percentage and their standard error of reported COVID-like illness symptoms, mask usage and direct contact from the Facebook Symptom Survey are calculated here, as well as the coordinates

of the partial dependence plots. Also, some values are determined and saved here for later usage, to accelerate the interactive visualization. These values include the maximum limit of the y axis, the applied restrictions per country on the exploratory tab, and selecting and ordering variables with highest variable importance in all countries in the random forest model for the bump chart tab.

As a last step to enable effective automatic update, a back-check is programmed, so if we update the database, a list is created automatically for the overlapping periods showing the differences between the newly downloaded data and its previous version. This way the user can follow the corrections made by the data providers.

2. Modelling

2.1. Times series methods revisited

When having time varying data, it comes naturally to use time series analysis methods. As our main focus is not to forecast a single time series, but to reveal the effects of many predictors, so ARIMA, ARCH/GARCH models were rejected. Though VAR (Vector Autoregression) is a multivariate forecasting algorithm that is used when two or more time series influence each other, according to whom it is not very powerful with epidemiological outcomes.

2.2. Model selection

When selecting a model, we have to precise the role of the model. Shmueli helps in clarifying the distinction between explanation and prediction (Shmueli 2010). We apply the conceptual framework of *prediction* here. In accordance with Shmueli, as "The prediction literature raises the importance of evaluating predictive power using holdout data, and the usefulness of algorithmic methods..." (Shmueli 2010:294), we use a Machine Learning approach, capturing the association instead of casual function between outcome and predictors. The amount of data and the complexity of predictors also support this direction. As Shmueli states, "Newly available large and rich datasets often contain complex relationships and patterns that are hard to hypothesize" (Shmueli 2010:292), and assumptions on variable distribution would be problematic as well.

Though we have time series data, we do not use our results for fortune telling, but stick ourselves to show present (and continuously updated) trends in our data.

In summary, our selected model should meet the following criteria in order of importance:

- give an adequate type of answer for the research question for time series data
- be able to deal with numerous predictors
- (almost) constrain-free predictor introduction
- interpretability (even with numerous predictors)
- visualizability of the results

The Random Forest method fulfills these criteria and the literature review of its application on epidemiological time series highlights the adequacy of the choice.

Random forests are an ensemble of learning methods for classification or regression (Breiman 2001). Here we use it for regression. It is a method of combination of trees such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. This method constructs a multitude of decision trees at training time and outputs mean prediction. The generalization error for forests converges as to a limit as the number of trees in the forest becomes large. The generalization error of a forest of tree classifiers depends on the strength of the individual trees in the forest and the correlation between them. Internal estimates are also used to measure variable importance.

Kane shows that Random Forest outperformed ARIMA time series models for prediction of avian influenza H5N1 outbreaks (Kane et al. 2014). Yeşilkanat achieved good results for COVID-19 when used spatio-temporal prediction on worldwide daily cases of COVID-19 applying random forest machine learning algorithm (Yeşilkanat 2020).
Turning from forecasting of new cases to analysis of predictors of the infection, Cobb examines the effect of social distancing on the compound growth rate of COVID-19 comparing statistical analyses and a random forest machine learning model and favoured random forest (Cobb et al. 2020)

2.3 Our model

We want to predict the variable importance of many predictors of the daily confirmed new COVID-19 cases across the countries of the European Union over time with Random Forest algorithm.
Our data was introduced in the chapter 'Data collection and preparation'.

As one of our goals is to compare the EU countries we considered data hierarchy and case dependency by setting up separate models for each country[6] with the same parameters like

[6] Except for Cyprus, Estonia, Latvia, Lithuania, Luxembourg, Malta, as the UMD-Facebook Symptom Survey does not contain data for these countries and exploratory analyses showed that its variables tend to be very important factors among other predictors.)

Chakraborti, who compared the five continents exploring determinant factors of the present pandemic comparing the results of five runs of their Random Forest model (Chakraborti et al. 2021). Technically, we split the data by countries generating a list with countries at the first level and Random Forest was implemented throughout the list via functional programming.

During data *preprocessing*, to express change as well, we prepared a cumulative version of the outcome variable, the confirmed new COVID-19 cases. Smoothing was implemented with 7-day rolling averaging.

As data is automatically updated, it is important to define a period from the latest datapoint, which expands until the latest possible data, but excludes the last few days where some parts of the data has not been published yet or frequent revisions occur. Considering these aspects, we cut the last 9 days for the random forest model.

Before entering into the model, the predictors are standardized, their correlation is checked. No highly correlated predictors (correlation over 0.7) have been found so far, so there is no need to use Conditional Forests (Stroble et al. 2007).
Deaths, recovered, average temperature, COVID-like illness, mask coverage, direct contact and vaccination variables are smoothed with a 7-day rolling average.

A 14-day lead for new COVID-19 cases were introduced as well, in order to consider the evolution of the virus.

2.4. Time series cross-validation

As we have time series data, we should also consider that our observations over time are not independent, so when resampling for training and test, simple random sampling of time points is not the best way. For Random Forest models (and other Machine Learning techniques), the *rolling forecasting origin* technique, introduced by Hyndman and Athanasopolous seems to deal with the problem by moving the training and test sets with predefined, fixed lengths in time (Hyndman/Athanasopolous 2018).

In this procedure, there are a series of test sets, each consisting of fixed lengths of observations. As Hyndman and Athanasopolous say, "The corresponding training set consists only of observations that occurred prior to the observation that forms the test set. Thus, no future observations can be used in constructing the forecast" (Hyndman/Athanasopolous 2018. 3.4 Evaluating forecast accuracy).

This technique is implemented in the R caret package.

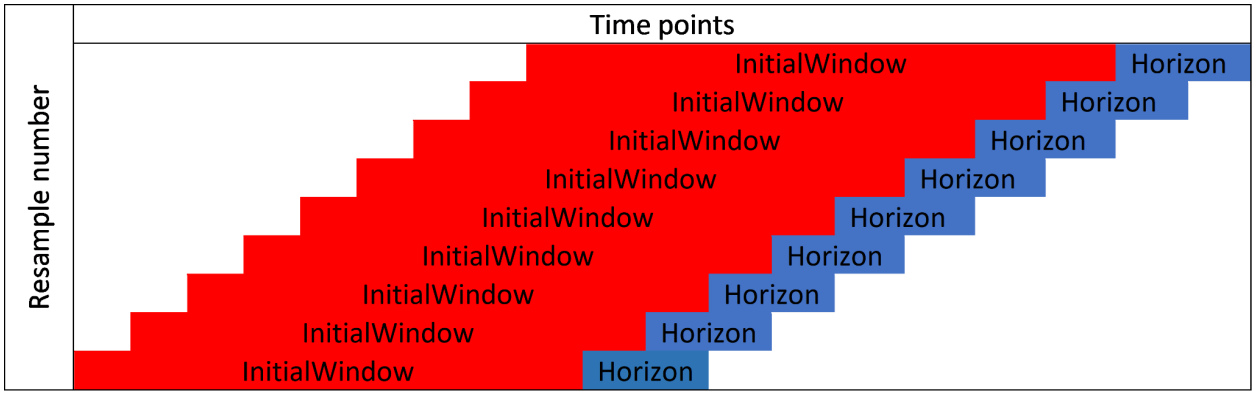

Figure x. Schema of the rolling forecasting origin method (based on Kuhn 2019)

The number of consecutive values in each training set sample (called initialWindow in R) is set to 28 days in order to cover a period long enough to contain enough time to possibly show an effect of a response measure considering the combination from the incubation period of COVID-19 with a median 4,5 to 5,8 days (95% CI) (McAloon et al. 2020), and the test delay (time until doctor visit and test evaluation time) (Dehning et al 2020).

The number of consecutive values in the test set sample (called Horizon in R) is 5 to allow for a relatively high number of resamples without 'running out' of the time series over time.

Our model ended with 236 to 356 (to be updated when fresh data comes in x) samples varying per country implemented with the Rolling Forecasting Origin resampling technique. RMSE was used to select the optimal model using the smallest value. The final number of predictors tried at each split (mtry) used for each country model was 9[7] with 500 trees.

[7] defined by a grid of mtry = c(round(sqrt(ncol(data))), round(log(ncol(data))))

The percentages of variance explained, i.e. the measure of how well out-of-bag predictions explain the target variance of the training set, are over 98 for each country model, most of them are over 99.
We used the results of the Random Forest models for Partial Dependence Plots and for the Bump Chart (variable importance) in our Covid app.

The sensitivity analysis to finalize the parameters for our Random Forest model covered several versions of the extent of time lag between predictors and reported infections and tests on dimensionality reduction, i.e. we produced new version of restrictions by merging restrictions with partially relaxed measures (for example merging complete and partially closure of hotels and accomodation services). Further, we tested different parameters of resampling time slices during model training.

2.5. Future plans to improve our predictions

For the outcome variable, the new confirmed cases we used a cumulated version. Instead of cumulation, we could study using compound growth rate or growth curve slope estimates for the outcome variable.
Our data is hierarchized: cases are grouped by countries. We ran our model separately for each country but countries could be considered by using spatial models. Statistical models as Spatial Error Model (SEM), Spatial Lag Model (SLM) (Sannigrahi et al. 2020) or Geographically Weighted Regression (GWR), or its extension into the Machine Learning approach, namely Geographically Weighted Random Forest (GWRF) could be applied. The latter one is a local nonlinear nonparametric regression model considering topography, which integrates a spatial weight matrix into Random Forest[8].

Competing Machine Learning applications (Uddin et al. 2019) for our research question according to the literature are Recurrent neural network (RNN) and Long short term memory (LSTM) or Gradient Boosted Machine (GBM) (Chakraborti et al. 2021).

[8] Can be implemented in the R package SpatialML.

3. Hierarchical cluster

To find the typical groups of countries with the similar characteristics we performed a hierarchical cluster analysis. The variables included in the cluster analysis are time-constant, therefore this analysis was conducted only once, and not part of the above automated processes, as analysis of time-dependent data.

As we did not want to determine the number of clusters in advance and had a small dataset, we performed agglomerative hierarchical clustering with the hclust R package. The following variables were included in the clustering algorithm:

- population size,
- healthcare expenditures (1000 Euro per Capita),
- cultural participation of 16-year and older (percentage of those who didn't attend any cultural event in the last 12 months),
- percentage of population in age groups (under 20, 20-39, 40-59, 60-79, above 80 years olds),
- percentage of males.

We standardized the variables, and increased the scale of some of the variables, to give them a weight in the cluster analysis. We doubled the scale of the variables ‘population size’ and ‘percentage of males’, and multiplied the scale of variables ‘healthcare expenditures’ and ‘cultural participation’ with 2.8. The aspects of choosing the exact magnitude of the weights were the maximization of the cophenetic correlation and the achievement of a sufficient number of clusters when defining the optimal number of clusters (described in the following chapters).

Agglomerative hierarchical clustering starts with every country representing a single cluster, and in every step of the algorithm, one pair of clusters, the one with the smallest intergroup dissimilarity is merged into one group. The algorithm stops when there is only one cluster left. This one contains all the countries.

When performing hierarchical clustering the distance measure between the pairs of observations and the measure of dissimilarities between the clusters have to be defined. Many distance measures (for example Euclidean, Gower distance) and linkage methods (for example single linkage, centroid linkage) are available. To measure the dissimilarity between two countries we used the Euclidean distance because all our variables are on a continuous scale. The distance between two clusters is measured with the Ward’s method, because this method resulted in the highest (0.69) correlation between the cophenetic distances (height at which two clusters are combined) and the dissimilarity measures. With Ward’s method, two clusters are selected in each step in a way that their merge into one cluster results in the smallest increase in the within cluster variance compared to merging any of the other clusters.

We defined the optimal number of clusters with the average silhouette method. The silhouette width measures how close the points of a cluster are to the points of the neighbouring cluster.
A high value of average silhouette width indicates that the observations are clustered well. A low value indicates the opposite, observations lying between or in the wrong clusters.
The average silhouette method calculates the average silhouette width for every potential value for the number of clusters. The value with the highest average silhouette width is the optimal one for the clusters. In our case, the optimal number of clusters is 6y as you can see it below.

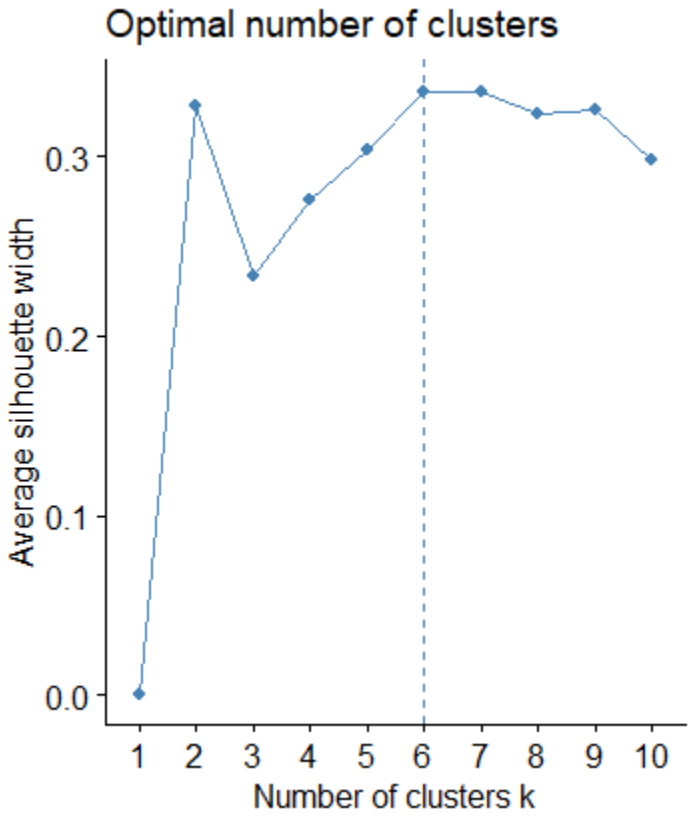

Figure X. Optimal number of clusters

The grouping process and the clusters created are visible on the following dendrogram.

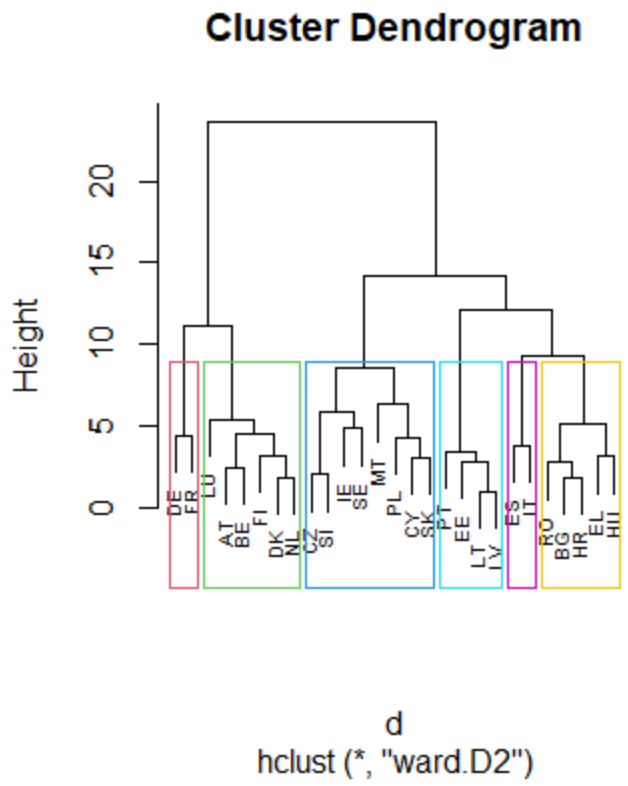

Figure X. Dendogram of the grouping process

Results

ToDO: Short description of the results

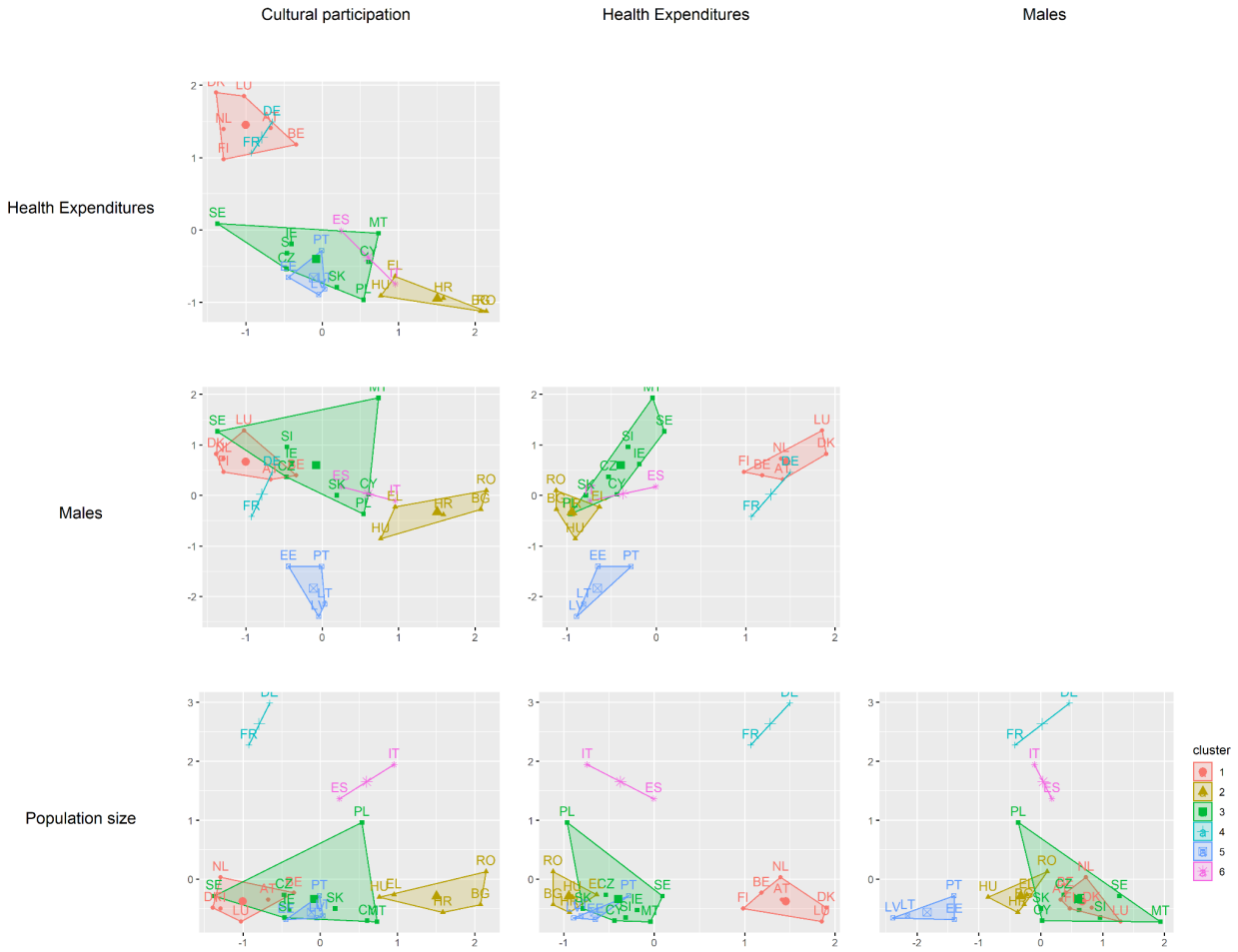

Figure X. Clusters visualized by variables 'cultural participation', 'healthcare expenditures', 'proportion of males' and 'population size' (Note that cultural participation represents the (scaled) percentage of population who did *not* attend any cultural events in the last 12 months.)

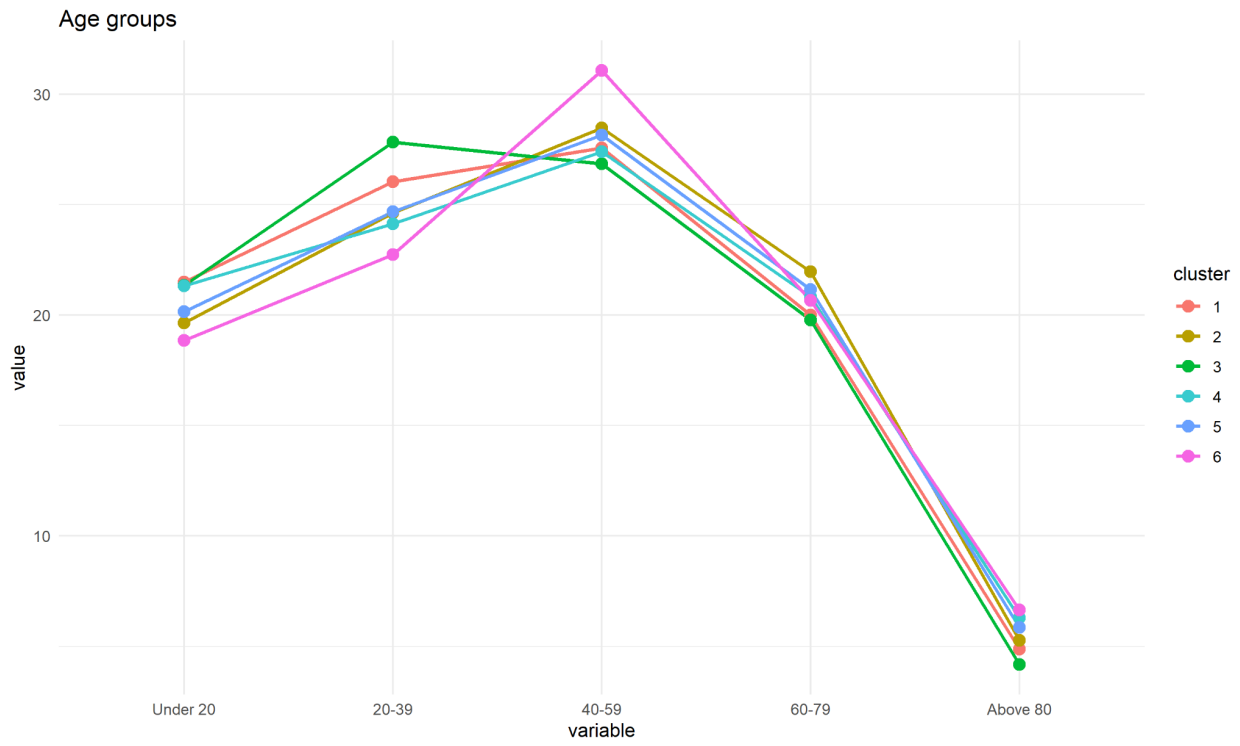

Figure X. Clusters by age groups

Cluster 1:
Countries: Austria, Belgium, Denmark, Finland, Luxemburg, Netherlands
Characteristics: Smaller population size, higher cultural participation, high healthcare expenditures and higher proportion of males. Percentage of population in younger age groups is higher, in older age groups lower than average.

Cluster 2:
Countries: Bulgaria, Greece, Croatia, Hungary, Romania
Characteristics: Low healthcare expenditures, and cultural participation, small population size. Percentage of population between 40 and 69 years is higher than average.

Cluster 3:
Countries: Cyprus, Czeh Republic, Ireland, Malta, Poland, Sweden, Slovenia, Slovakia
Characteristics: Cultural participation and proportion of males are between average and high, healthcare expenditures and population size are between average and small. Percentage of population in younger age groups is higher, in older age groups lower than average.

Cluster 4:
Countries: Germany, France

Characteristics: High healthcare expenditures, cultural participation and large population size. Percentage of population in the youngest and oldest age groups is higher.

Cluster 5:
Countries: Estonia, Lithuania, Latvia, Portugal
Characteristics: Small population size and high healthcare expenditures, cultural participation and proportion of males are somewhat lower than average. Percentage of the population in the younger age group is lower.

Cluster 6:
Countries: Spain, Italy
Characteristics: Large population size, somewhat lower cultural participation and healthcare expenditures. Percentage of population in the younger age group is lower, in the age group 40-59 years and above 80 years higher than average.

cluster	Countries	Population size	Males	Health Expenditures	Cultural participation	Under 20	20-39	40-59	60-79	Above 80
1	AT, BE, DK, FI, LU, NL	8255725	49,59%	4,56%	21,05%	21,50%	26,05%	27,56%	20,01%	4,87%
2	BG, EL, HR, HU, RO	10197620	48,45%	0,86%	62,10%	19,65%	24,62%	28,48%	21,98%	5,27%
3	CY, CZ, IE, MT, PL, SE, SI, SK	9082228	49,51%	1,70%	36,18%	21,35%	27,84%	26,86%	19,78%	4,17%
4	DE, FR	75098425	48,84%	4,29%	24,50%	21,32%	24,14%	27,40%	20,84%	6,29%
5	EE, LT, LV, PT	4078897	46,72%	1,30%	35,55%	20,16%	24,68%	28,16%	21,15%	5,86%
6	ES, IT	53376867	48,86%	1,74%	47,30%	18,85%	22,73%	31,09%	20,67%	6,66%

Figure X. Countries assigned to clusters and mean values within clusters

4. Shiny Dashboard - description

Our interactive data visualization tool was created with the shiny and shinydashboard R packages. The inputs of the application are the prepared databases and the results of the random forest models, described in the previous sections.

When starting, the application reads the input and opens the dashboard with the first tab. After that, the application will respond interactively to the user's actions. The user can change tabs and make inquiries (for example: setting time interval, selecting variable for visualization).

Functions with reactivity, which can be triggered by the user are in a separate R file. These functions process the data according to the user's queries.
It contains functions for

- selecting the smoothed or unsmoothed variables and setting the limit of the Y axis according to it,
- calculating the 99% confidence intervals for the percentage of reported COVID-like illness symptoms, mask usage and direct contact from the Facebook Symptom Survey from the standard error,
- adding lead to the number of new infections,
- calculating the X and coordinates for the visualization of time intervals in which the selected restrictions were applied on the exploratory tab,
- determining coordinates for labels,
- selecting and deselecting predictors,
- formatting checkbox groups.

There are also functions, of which the purpose is the interactive visualization, responding to the user's action.
These are functions to

- determine titles, subtitles, possibly reflecting deficiencies (for example: data not available for the selected time interval, too big lead set, no predictor selected),
- create different type of plots,
- add variables to the plots.

The dashboard consists of 4 tabs: Exploratory, Partial Dependence, Bump Chart and Datasource. The user can move between these pages by selecting them on the left side of the application.

5. Data sources and Database Description

4.1. Time-constant country characteristics from Eurostat sources

Name of database: country_char

Sources and variables:

- Data source: eurostat, https://ec.europa.eu/eurostat
- Population: population on 1. January 2019 by age group and sex
    - Variables:
        - geo: Geopolitical entity
        - Y_LT5: Less than 5 years
        - Y5-9: From 5 to 9 years
        - Y10-14: From 10 to 14 years
        - Y15-19: From 15 to 19 years
        - Y20-24: From 20 to 24 years
        - Y25-29: From 25 to 29 years
        - Y30-34: From 30 to 34 years
        - Y35-39: From 35 to 39 years
        - Y40-44: From 40 to 44 years
        - Y45-49: From 45 to 49 years
        - Y50-54: From 50 to 54 years
        - Y55-59: From 55 to 59 years
        - Y60-64: From 60 to 64 years
        - Y65-69: From 65 to 69 years
        - Y70-74: From 70 to 74 years
        - Y75-79: From 75 to 79 years
        - Y_GE75: 75 years or over
        - Y80-84: From 80 to 84 years
        - Y_GE80: 80 years or over
        - Y_GE85: 85 years or over
        - T: Total
        - M: Males
        - F: Females
- Health expenditures: Total health care expenditure amount in millions of euro, 2018
    - Variables:
        - health_expenditures: Total health expenditures
- Cultural participation: Frequency of participation in cultural or activities in the last 12 months by age, 2015. Percentage of those who did not attend any cultural event (cinema, live performances or cultural sites) in the last 12 months by age groups
    - Variables:
        - cult_Y_GE16 : 16 years or over

        - cult_Y_GE75 : 75 years or over
        - cult_Y16-24: From 16 to 24 years
        - cult_Y16-29: From 16 to 29 years
        - cult_Y25-34: From 25 to 34 years
        - cult_Y25-64: From 25 to 64 years
        - cult_Y35-49: From 35 to 49 years
        - cult_Y50-64: From 50 to 64 years
        - cult_Y65-74: From 65 to 74 years

4.2. Time-varying COVID-19 related variables from various sources

Name of database: tdata

Sources and variables:

- Testing:
    - Data source: European Centre for Disease Prevention and Control, Data on testing for COVID-19 by week and country, https://www.ecdc.europa.eu/en/publications-data/covid-19-testing
    - Testing volume and positivity rate by week
    - Variables:
        - country
        - country_code: 2-letter ISO country code
        - testing_new_cases: Number of new confirmed cases
        - tests_done: Number of tests done
        - testing_population
        - testing_rate: Testing rate per 100 000 population
        - testing_positivity_rate: Weekly test positivity (%): 100 x Number of new confirmed cases/number of tests done per week
- Response measures:
    - Data source: European Centre for Disease Prevention and Control, Data on country response measures to COVID-19 by week and country, https://www.ecdc.europa.eu/en/publications-data/download-data-response-measures-covid-19
    - Non-pharmaceutical interventions taken by countries in response to the pandemics
    - Variables:
        - Country
        - date
        - year
        - week

- AdaptationOfWorkplace: Adaptation of workplaces(e.g. to reduce risk of transmission)
- AdaptationOfWorkplacePartial: Adaptation of workplaces (e.g. to reduce risk of transmission)-partially relaxed measure
- BanOnAllEvents: Interventions are in place to limit all indoor/outdoor mass/public gatherings
- BanOnAllEventsPartial: Interventions are in place to limit all indoor/outdoor mass/public gatherings-partially relaxed measure
- ClosDaycare: Closure of educational institutions: daycare or nursery.
- ClosDaycarePartial: Closure of educational institutions: daycare or nursery -partially relaxed measure
- ClosHigh: Closure of educational institutions: higher education.
- ClosHighPartial: Closure of educational institutions: higher education -partially relaxed measure
- ClosPrim: Closure of educational institutions: primary schools.
- ClosPrimPartial: Closure of educational institutions: primary schools -partially relaxed measure
- ClosPubAny: Closure of public spaces of any kind (including restaurants, entertainment venues, non-essential shops, partial or full closure of public transport, gyms and sport centers, etc).
- ClosPubAnyPartial: Closure of public spaces of any kind (including restaurants, entertainment venues, non-essential shops, partial orfull closure of public transport, gyms and sport centers etc) -partially relaxed measure
- ClosSec: Closure of educational institutions: secondary schools.
- ClosSecPartial: Closure of educational institutions: secondary schools -partially relaxed measure
- ClosureOfPublicTransport: Closure of public transport
- ClosureOfPublicTransportPartial: Closure of public transport-partially relaxed measure
- EntertainmentVenues: Closure of entertainment venues
- EntertainmentVenuesPartial: Closure of entertainment venues-partially relaxed measure
- GymsSportsCentres: Closure of gyms/sports centres
- GymsSportsCentresPartial: Closure of gyms/sports centres-partially relaxed measure
- HotelsAccommodation: Closure of hotels/accommodation services
- HotelsAccommodationPartial: Closure of hotels/accommodation services-partially relaxed measure

- IndoorOver100: Interventions are in place to limit indoor mass/public gatherings of over 100participants
- IndoorOver1000: Interventions are in place to limit indoor mass/public gatherings of over 1000participants
- IndoorOver1000Partial: Interventions are in place to limit indoor mass/public gatherings of over 1000participants-partially relaxed measure
- IndoorOver100Partial: Interventions are in place to limit indoor mass/public gatherings of over 100participants-partially relaxed measure
- IndoorOver50: Interventions are in place to limit indoor mass/public gatherings of over 50participants
- IndoorOver500: Interventions are in place to limit indoor mass/public gatherings of over 500participants
- IndoorOver500Partial: Interventions are in place to limit indoor mass/public gatherings of over 500participants-partially relaxed measure
- IndoorOver50Partial: Interventions are in place to limit indoor mass/public gatherings of over 50participants-partially relaxed measure
- MasksMandatoryAllSpaces: Protective mask use in all public spaces on mandatory basis (enforced by law)
- MasksMandatoryAllSpacesPartial: Protective mask use in all public spaces on mandatory basis (enforced by law)-partially relaxed measure
- MasksMandatoryClosedSpaces: Protective mask use in closed public spaces/transport on mandatory basis (enforced by law)
- MasksMandatoryClosedSpacesPartial: Protective mask use in closed public spaces/transport on mandatory basis (enforced by law)-partially relaxed measure
- MasksVoluntaryAllSpaces: Protective mask use in all public spaces on voluntary basis (general recommendation not enforced)
- MasksVoluntaryAllSpacesPartial: Protective mask use in all public spaces on voluntary basis (general recommendation not enforced)-partially relaxed measure
- MasksVoluntaryClosedSpaces: Protective mask use in closed public spaces/transport on voluntary basis (general recommendation not enforced)
- MasksVoluntaryClosedSpacesPartial: Protective mask use in closed public spaces/transport on voluntary basis (general recommendation not enforced)-partially relaxed measure
- MassGatherAll: Interventions are in place to limit mass/public gatherings (any interventions on mass gatherings up to 1000 participants included)

- MassGatherAllPartial: Interventions are in place to limit mass/public gatherings (any interventions on mass gatherings up to 1000 participants included)-partially relaxed measure
- NonEssentialShops: Closures of non-essential shops
- NonEssentialShopsPartial: Closures of non-essential shops -partially relaxed measure
- OutdoorOver100: Interventions are in place to limit outdoor mass/public gatherings of over 100participants
- OutdoorOver1000: Interventions are in place to limit outdoor mass/public gatherings of over 1000participants
- OutdoorOver1000Partial: Interventions are in place to limit outdoor mass/public gatherings of over 1000participants-partially relaxed measure
- OutdoorOver100Partial: Interventions are in place to limit outdoor mass/public gatherings of over 100participants-partially relaxed measure
- OutdoorOver50: Interventions are in place to limit outdoor mass/public gatherings of over 50participants
- OutdoorOver500: Interventions are in place to limit outdoor mass/public gatherings of over 500participants
- OutdoorOver500Partial: Interventions are in place to limit outdoor mass/public gatherings of over 500participants-partially relaxed measure
- OutdoorOver50Partial: Interventions are in place to limit outdoor mass/public gatherings of over 50participants-partially relaxed measure
- PlaceOfWorship: Closure of places of worship
- PlaceOfWorshipPartial: Closure of places of worship-partially relaxed measure
- PrivateGatheringRestrictions: Restrictions on private gatherings
- PrivateGatheringRestrictionsPartial: Restrictions on private gatherings-partially relaxed measure
- RegionalStayHomeOrder: Regional stay-at-home orders for the general population at least in one region(these are enforced and also referred to as 'lockdown')
- RegionalStayHomeOrderPartial: Regional stay-at-home orders for the general population at least in one region (these are enforced and also referred to as 'lockdown')-partially relaxed measure
- RestaurantsCafes: Closure of restaurants and cafes/bars
- RestaurantsCafesPartial: Closure of restaurants and cafes/bars-partially relaxed measure
- SocialCircle: Social circle/bubble to limit social contacts e.g. to limited number of households

        - SocialCirclePartial: Social circle/bubble to limit social contacts e.g. to limited number of households-partially relaxed measure
        - StayHomeGen: Stay-at-home recommendations for the general population (which are voluntary or not enforced)
        - StayHomeGenPartial: Stay-at-home recommendations for the general population (which are voluntary or not enforced) -partially relaxed measure
        - StayHomeOrder: Stay-at-home orders for the general population (these are enforced and also referred to as 'lockdown')
        - StayHomeOrderPartial: Stay-at-home orders for the general population (these are enforced and also referred to as 'lockdown') -partially relaxed measure
        - StayHomeRiskG: Stay-at-home recommendations for risk groups or vulnerable populations (such as the elderly, people with underlying health conditions, physically disabled people, etc.)
        - StayHomeRiskGPartial: Stay-at-home recommendations for risk groups or vulnerable populations (such as the elderly, people with underlying health conditions, physically disabled people, etc.) -partially relaxed measure
        - Teleworking: Teleworking recommendation
        - TeleworkingPartial: Teleworking recommendation or workplace closures -partially relaxed measure
        - WorkplaceClosures: Closures of workplaces
        - WorkplaceClosuresPartial: Closures of workplaces-partially relaxed measure
- Weather:
    - Data source: National Centers for Environmental Information, https://www.ncdc.noaa.gov/
    - Daily average temperatures measured in the capitals of the countries involved in the analysis
    - In a few cases some values are missing for shorter periods for some countries, for example: values are missing for Spain between 03.11.2020 and 03.14.2020.
    - Variables:
        - country_code: 2-letter ISO country code
        - date
        - tavg: average daily temperature
- Vaccination:
    - Our World in Data, https://ourworldindata.org/coronavirus
    - Variables:
        - iso_code: ISO country code
        - country

        - date
        - total_vaccinations: total number of doses administered
        - people_vaccinated: total number of people who received at least one vaccine dose. If a person receives the first dose of a 2-dose vaccine, this metric goes up by 1. If they receive the second dose, the metric stays the same.
        - people_fully_vaccinated: total number of people who received all doses prescribed by the vaccination protocol. If a person receives the first dose of a 2-dose vaccine, this metric stays the same. If they receive the second dose, the metric goes up by 1.
        - new_vaccinations: daily change in the total number of doses administered
        - new_vaccinations_smoothed: new doses administered per day (7-day smoothed (for countries that don't report data on a daily basis, the daily changes on doses assumed to be equal over the period in which no data was reported)
        - total_vaccinations_per_hundred: people vaccinated per 100 people in the total population of the country
        - people_vaccinated_per_hundred: people vaccinated per 100 people in the total population of the country.
        - people_fully_vaccinated_per_hundred: people fully vaccinated per 100 people in the total population of the country.
        - new_vaccinations_smoothed_per_million: daily vaccinations per 1,000,000 people in the total population of the country

- Covid cases:
    - https://github.com/RamiKrispin/coronavirus
    - The coronavirus package provides a tidy format dataset of the 2019 Novel Coronavirus COVID-19 (2019-nCoV) epidemic. The raw data is pulled from the Johns Hopkins University Center for Systems Science and Engineering (JHU CCSE) Coronavirus repository.
    - Variables:
        - cases_new: Confirmed daily new cases
        - deaths_new: Daily number of deaths
        - recovered_new: Daily number of the recovered
- UMD/Facebook Symptom survey:
    - https://covidmap.umd.edu/api/
    - variables:
        - fb_data.iso_code: ISO country codes
        - fb_data.percent_cli: weighted percentage of respondents that have reported Covid Like Illness

- fb_data.cli_se: standard error of percent_cli
- fb_data.percent_cli_unw: unweighted percentage of respondents that have reported CLI
- fb_data.cli_se_unw: standard error of percent_cli_unw
- fb_data.sample_size_cli: sample size for calculating CLI
- fb_data.smoothed_cli: smoothed percentage of respondents that have reported Covid Like Illness
- fb_data.smoothed_cli_se: standard error of smoothed percent_cli
- fb_data.sample_size_smoothed_cli: sample size for calculating smoothed CLI
- fb_data.percent_mc: weighted percentage of respondents that have reported using a mask
- fb_data.mc_se: standard error of percent_mc
- fb_data.percent_mc_unw: unweighted percentage of respondents that have reported use mask cover
- fb_data.mc_se_unw: standard error of percent_mc_unw
- fb_data.sample_size_mc: sample size for calculating mask coverage
- fb_data.smoothed_mc: smoothed percentage of respondents that have reported use mask cover
- fb_data.smoothed_mc_se_ standard error of smoothed percent_mc
- fb_data.sample_size_mc_smoothed: sample size for calculating smoothed mc
- fb_data.percent_dc: weighted percentage of respondents that have reported had direct contact (longer than one minute) with people not staying with them in last 24 hours
- fb_data.mc_se_dc
- fb_data.percent_dc_unw: unweighted percentage of respondents that have reported use have direct contact with people not staying with them
- fb_data.dc_se_unw: standard error of percent_dc_unw
- fb_data.sample_size_dc: sample size for calculating direct contact
- fb_data.smoothed_dc: smoothed percentage of respondents that have reported direct contact
- fb_data.smoothed_dc_se: standard error of smoothed percent_dc
- fb_data.sample_size_dc_smoothed: sample size for calculating smoothed dc

6. Code structure

All codes can be found in our GitHub repository.

data_collection

Create_database.R

- Creates the databases for the first time. Downloads data from all sources, creates and saves two databases:
    - country_car: country characteristics (merged eurostat databases)
    - tdata: data from all other sources, with time variable
- Uses the following scripts:
    - data_collection/Save_data.R
    - data_collection/Collect_data.R
    - data_collection/Merge_data.R

Save_data.R

- Contains two functions (save database to local or to online location) to save databases.
- If the database should be saved online, the data will be written to password protected Google Sheets spreadsheet. In this case, an authentication file is necessary to reach the appropriate Google Sheets account. This file has to be located under a directory called ".secrets" in the working directory.
- In both cases, if "archive" is set to true, if the database already exists, the old database will be kept and renamed.

Collect_data.R

- Downloads data from all sources.
- FIlter time variable for latest eurostat databases:
    - Demographics: 2019
    - Number of practicing physicians: 2019
    - Health expenditures: 2018
    - Cultural participation: 2015

Merge_data.R

- Formats the data and merges it into two databases:
    - country_car: country characteristics (merged eurostat databases)
    - tdata: data from all other sources, with time variable
- Country characteristics
    - Demographic variables:
        - total population

        - population by sex
        - population by age groups (under 30, above 75)
    - Health expenditures
    - Cultural participation:
        - Percentage of population by age groups who didn't attend on any cultural event in the last 12 months (cinema, live performances or cultural sites)
        - Age groups: 16 years and older, under 30, above 75
- Data with time variable:
    - Formats databases to enable merging them
    - Keeps all variables
    - Most data are per day (data on testing per week)
    - Uses the following scripts:
    - functions/Data_preparation_functions.R
    - functions/Data_cleansing_functions.R

Update_data.R

- Updates tdata with new records since the last download and saves it.
- Date of the last day of data availability is often different for data sources and for countries within data sources. The data will be updated for every variable from the first day when data is not available for all the countries.
- Non-missing values stay the same.
- Uses the following scripts:
    - functions/Data_preparation_functions.R
    - data_collection/Save_data.R

Revise_data.R

- Recollects and merges tdata from all data sources. Saves the updated dataset.
- Lists the differences between the old and the new tdata.
- Functions are available to examine the differences:
    - number of differences per variable
    - first n differences per variable
    - last n differences per variable
    - all differences for one variable
    - new variables that do not exist in the old tdata
- Functions are available to update the old tdata with the new values:
    - update selected record of a variable
    - update all different records of a variable
    - add new variable to old tdata
    - replace old tdata with new tdata
- Uses the following scripts:

    - data_collection/Save_data.R
    - data_collection/Collect_data.R
    - data_collection/Merge_data.R
    - data_collection/Data_revision_functions.R

functions

Data_preparation_functions.R
- Contains functions used during the preparation and merge of the data.

Data_cleansing_functions.R
- Contains functions for data cleansing.

Data_process_functions.R
- Contains functions to edit and process the data during the analysis and visualization.
    - Merging partially relaxed and not relaxed measures (partial restrictions with not partials)

Data_revision_functions.R
- Contains functions used during the data revision.
- Shows differences in details between two databases.
- Updates database.

Get_data.R
- Contains a function to load the databases form a local or online location.
- If the database should be loaded from Google Sheets, an authentication file is necessary to reach the appropriate Google Sheets account. This file has to be located under a directory called ".secrets" in the working directory.

RF_functions.R
- Contains functions to prepare data for RF modelling
- Standardizes predictors
- Runs RF model for all the countries

Shiny_prep_functions.R
- Contains functions to respond interactively to the queries in the shiny application.
- Prepares selected variables and time intervals for the visualization.
- Functions used for the plot on tab Overview:

    - Calculates moving average.
    - Adds the selected lead to the dataset.
    - Calculates the coordinates for the visualization of the restriction measures.

helpers

Near_stations.R

- Lists weather station IDs near to capitals.
- Normally the nearest station is selected.
- In some cases the nearest station is not functioning in the whole time interval, in these cases the station is selected manually from the next nearest stations.

Prepare_run.R

- Prepares the data collection and analysis.
- **Must be run before any other codes.**
- Sets the working directory and maximum date for data collection
- Creates a dataframe with the EU countries, capitals, different country codes, latitudes, longitudes and weather station IDs.

Data_cleansing.R

- Explores the data to discover the necessary data cleansing steps.

Add_variable_labels.R

- Adds labels to the variables.

Change_variable_types.R

- Change variable types (to numerical, factor or date).

Set_up_authentication.R

- Generates the token to access the private Google Sheets where the data are stored.
- Sets the directory where the generated token will be stored.
- Opens a browser and starts an interactive authentication to generate the token.
- Don't run this. The purpose of this script is only to demonstrate, how the authentication file was created.

Get_and_prepare_data.R

- Loads the two (tdata and country_char) databases from Google Sheets.
- Adds variable labels and changes the variable types if necessary.
- Uses the following scripts:
    - functions/Get_data.R

    - helpers/Change_variable_types.R
    - helpers/Add_variable_labels.R

random_forest

Random_forest.R

- Reads in the data from googlesheets and prepares it.
- Preprocessing:
    - selects variables for the model
    - computes smoothing, leads, lags, etc
    - standardises predictors, checks for highly correlated predictors
    - runs RF with fixed window timeslices on cumulative smoothed outcome on all countries separately
- Uses the following scripts:
    - functions/RF_functions.R
    - functions/Get_data.R

shinydashboard

app.R

- Contains the content and functionality of the shiny dashboard application.
- Defines the header, sidebar and tabs of the application.

Call_shiny.R

- Reads in the data from googlesheets and prepares it.
- Prepares the data for the dashboard.
- Starts the application.
- Uses the following scripts:
    - helpers/Get_and_prepare_data.R
    - shinydashboard/Shiny_data_prep.R

Shiny_data_prep.R

- Prepares the data for the dashboard to enable fast visualisation and to improve interactivity.
- Plot on tab “Overview”:
    - Creates a list for the countries for a more rapid selection.
    - Adds variables to help fasten the calculation of the coordinates of the restriction measures for the visualization.

Shiny_prep_and_functions.R

- Reads in the data prepared by Shiny_data_prep.R when starting the shiny application.
- Contains functions to process the data by the users queries.

Shiny_vis_functions.R

- Contains functions for the interactive visualisation in the shiny application.

7. Bibliography

- Breiman, Leo, 2001: Random Forests. Machine Learning 45: 5–32.
- Chakraborti, Suman, Arabinda Maiti, Suvamoy Pramanik, Srikanta Sannigrahi, Francesco Pilla, Anushna Banerjee and Dipendra Nath Das, 2021: Evaluating the plausible application of advanced machine learnings in exploring determinant factors of present pandemic: A case for continent specific COVID-19 analysis. Science of the Total Environment 765.
- Cobb, Jared and M. A. Seale, 2020: Examining the effect of social distancing on the compound growth rate of COVID-19 at the county level (United States) using statistical analyses and a random forest machine learning model. Public Health, 185.
- Dehning, Jonas, Johannes Zierenberg, Paul Spitzner, Michael Wibral, Joao Pinheiro Neto, Michael Wilczek and Viola Priesemann, 2020: Inferring change points in the spread of COVID-19 reveals the effectiveness of interventions. Science369, eabb9789. DOI: 10.1126/science.abb97
- Hyndman, R.J., & Athanasopoulos, G. (2018) Forecasting: principles and practice, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on 15.05.2021.
- Kane, Michael J, Natalie Price, Matthew Scotch and Peter Rabinowitz, 2014: Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks. BMC Bioinformatics 15: 276.
- Krispin, Rami 2020. Total number of recovered cases negative. https://github.com/RamiKrispin/coronavirus/issues/55 accessed on 15.05.2021.
- Kuhn, M. 2019. The caret Package. https://topepo.github.io/caret/data-splitting.html accessed on 15.05.2021.
- Liaw, Andy, Matthew Wiener, 2002: Classification and Regression by randomForest. R News Vol. 2/3: 18-22.
- Luo, Yaowen, Jianguo Yan, and Stephen McClure, 2020: Distribution of the environmental and socioeconomic risk factors on COVID-19 death rate across continental USA: a spatial nonlinear analysis. Environ Sci Pollut Res 28, 6587–6599.
- McAloon, Conor, Áine Collins, Kevin Hunt, Ann Barber, Andrew W Byrne, Francis Butler, Miriam Casey, John Griffin, Elizabeth Lane, David McEvoy, Patrick Wall, Martin Green, Luke O'Grady and Simon J More, 2020: Incubation period of COVID-19: a rapid systematic review and meta-analysis of observational research. BMJ Open 10:e039652. doi:10.1136/bmjopen-2020-03965
- R-bloggers 2018. Bump Chart. https://www.r-bloggers.com/2018/04/bump-chart/ accessed on 15.05.2021.
- Sannigrahi, Srikanta ,Francesco Pilla, Bidroha Basu, Arunima Sarkar Basu and Anna Molter. 2020: Examining the association between socio-demographic composition and COVID-19 fatalities in the European region using spatial regression approach. Sustainable Cities and Society 62.
- Shmueli, Galit, 2010: To Explain or to Predict? Statistical Science 25: 289–310.

- Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis and Torsten Hothorn, 2007: Bias in random forest variable importance measures: Illustrations, sources and a solution. BMC Bioinformatics 8: 25.
- Uddin, Shahadat, Arif Khan, Md Ekramul Hossain and Mohammad Ali Moni, 2019: Comparing different supervised machine learning algorithms for disease prediction. BMC Medical Informatics and Decision Making 19:281
- Yeşilkanat, Cafer Mert, 2020: Spatio-temporal estimation of the daily cases of COVID-19 in worldwide using random forest machine learning algorithm. Chaos, Solitons & Fractals 140.
- Hastie, Trevor, Robert Tibshirani, Jerome Friedman (2008). The Elements of Statistical Learning. Data Mining, Inference and Prediction. Second Edition. Springer, New York City, USA 520-528.
- Kassambara, Alboukadel (2017). Practical Guide to Cluster Analysis in R. Unsupervised Machine Learning. STHDA (http://www.sthda.com) 130-140.

8. List of figures

To do

9. R packages

- tidyverse, Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686
- eurostat, (C) Leo Lahti, Janne Huovari, Markus Kainu, Przemyslaw Biecek. Retrieval and analysis of Eurostat open data with the eurostat package. R Journal 9(1):385-392, 2017. Version 3.6.84 Package URL: http://ropengov.github.io/eurostat Manuscript URL: https://journal.r-project.org/archive/2017/RJ-2017-019/index.html
- coronavirus, Rami Krispin and Jarrett Byrnes (2021). coronavirus: The 2019 Novel Coronavirus COVID-19 (2019-nCoV) Dataset. R package version 0.3.1. https://CRAN.R-project.org/package=coronavirus
- data.table, Matt Dowle and Arun Srinivasan (2020). data.table: Extension of `data.frame`. R package version 1.13.6. https://CRAN.R-project.org/package=data.table
- httr, Hadley Wickham (2020). httr: Tools for Working with URLs and HTTP. R package version 1.4.2. https://CRAN.R-project.org/package=httr
- rnoaa, Scott Chamberlain (2021). rnoaa: 'NOAA' Weather Data from R. R package version 1.3.0. https://CRAN.R-project.org/package=rnoaa
- jsonlite, Jeroen Ooms (2014). The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects. arXiv:1403.2805 [stat.CO] URL https://arxiv.org/abs/1403.2805.
- lubridate, Garrett Grolemund, Hadley Wickham (2011). Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25. URL https://www.jstatsoft.org/v40/i03/.
- compare, Paul Murrell (2015). compare: Comparing Objects for Differences. R package version 0.2-6. https://CRAN.R-project.org/package=compare
- googlesheets4, Jennifer Bryan (2020). googlesheets4: Access Google Sheets using the Sheets API V4. R package version 0.2.0. https://CRAN.R-project.org/package=googlesheets4
- Hmisc, Frank E Harrell Jr, with contributions from Charles Dupont and many others. (2020). Hmisc: Harrell Miscellaneous. R package version 4.4-2. https://CRAN.R-project.org/package=Hmisc
- rvest, Hadley Wickham (2020). rvest: Easily Harvest (Scrape) Web Pages. R package version 0.3.6. https://CRAN.R-project.org/package=rvest
- RColorBrewer, Erich Neuwirth (2014). RColorBrewer: ColorBrewer Palettes. R package version 1.1-2. https://CRAN.R-project.org/package=RColorBrewer
- zoo, Achim Zeileis and Gabor Grothendieck (2005). zoo: S3 Infrastructure for Regular and Irregular Time Series. Journal of Statistical Software, 14(6), 1-27. doi:10.18637/jss.v014.i06
- shiny, Winston Chang, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert and Barbara Borges (2021). shiny:

Web Application Framework for R. R package version 1.6.0.
https://CRAN.R-project.org/package=shiny

- shinydashboard, Winston Chang and Barbara Borges Ribeiro (2018). shinydashboard: Create Dashboards with 'Shiny'. R package version 0.7.1.
  https://CRAN.R-project.org/package=shinydashboard
- caret, Max Kuhn et al (2020).
  caret: Classification and Regression Training. R package version 6.0-86.
  https://CRAN.R-project.org/package=caret
- randomForest, Leo Breiman, Adele Cutler, Andy Liaw , Matthew Wiener (2018).
  randomForest: Breiman and Cutler's Random Forests for Classification and Regression. R package version 4.6-14.
  https://CRAN.R-project.org/package=randomForest
- Hmisc, Frank E Harrell Jr, Charles Dupont (2021).
  Hmisc: Harrell Miscellaneous. R package version 4.5-0.
  https://CRAN.R-project.org/package=Hmisc