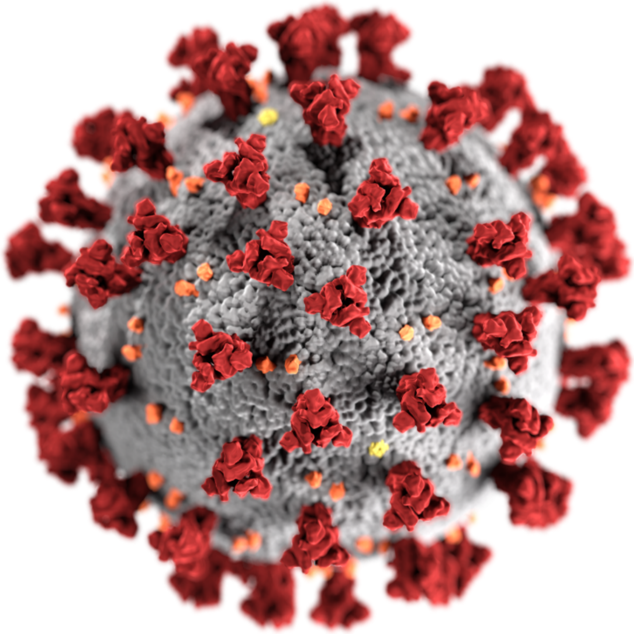


# Modeling Prediction on COVID-19

DEPARTMENT OF COMPUTATIONAL AND DATA SCIENCE

CDS-403-001: MACHINE LEARNING

Liying (Lily) Lu | Deja Watkins | Felicia Natalie Wijaya

# Index

## Overview

On March 11, 2020, the World Health Organization (WHO) characterized the coronavirus disease 2019 (COVID-19) outbreak as a pandemic. COVID-19 is a novel betacoronavirus which causes respiratory diseases and has its origin in bats. Researchers have found that the virus is in the same family as MERS-CoV and SARS-CoV and is particularly similar to SARS-CoV. Therefore, COVID-19 is now also named SARS-CoV-2. The virus could be spread from animals (bats) to humans and from humans to humans. The first case of COVID-19 was reported in China on December 31, 2019. The first recorded case outside of China was in Thailand on 13 January, 2020. The virus soon spread across the world, recording 118,613 confirmed cases worldwide on the day of the pandemic declaration by the WHO. As of May 11, 2019, there are 4.06 million confirmed cases worldwide. The cases increased 34 folds over the course of two months. However, the actual number of total cases is likely to be higher than the recorded number due to limited testing.

The coronavirus pandemic has affected the lives of global citizens, causing economic, social, and health distress. It would be useful to develop a predictive model to forecast the trend of the cases and death in each region in the world. This could help the governments, hospitals, and medical supply manufacturers to anticipate an estimated peak number of cases and prepare for appropriate resource allocation and strategy planning. The goal of the project is to find any possible trends in the growth of recorded cases and death. We would also analyze the correlation of factors which are claimed to affect the spread of the disease by the news and other publications.

## Motivation

We wanted to participate in a Kaggle competition at least once before we graduate. The weekly COVID-19 Forecasting competitions seem like an appropriate topic for the semester and give us the opportunity to work with real-time data. It seems simple enough and we had confidence that we could pull this off.

## Related Work

We discussed numerous regression models in class, such as ridge regression, lasso regression, and logistic regression. These models could be useful in our competitions. We are also interested in the stacking method which combines two or more regression models as one.

(https://scikit-learn.org/stable/auto_examples/ensemble/plot_stack_predictors.html#sphx-glr-auto-examples-ensemble-plot-stack-predictors-py)

Besides the linear regression models, a polynomial regression model is also worth a try due to the nonlinear nature of the data.

(https://towardsdatascience.com/machine-learning-with-python-easy-and-robust-method-to-fit-nonlinear-data-19e8a1ddbd49).

Cascella, M., Rajnik, M., Cuomo, A., Dulebohn, S., & Napoli, R. (2020). Features, Evaluation and Treatment Coronavirus (COVID-19). Ncbi.nlm.nih.gov. Retrieved 11 May 2020, from https://www.ncbi.nlm.nih.gov/books/NBK554776/.

This article discusses the reproduction number, rate of infectiousness, and other parameters which would be useful for building a SEIR model.

Coronavirus (COVID-19) Cases - Statistics and Research. Our World in Data. (2020). Retrieved 11 May 2020, from https://ourworldindata.org/covid-cases.

We also tried the ‘modified’ SIRS model, known as SEIRS with vital dynamics. We called this model SEIRCD (Susceptible, Exposed, Infectious, Recovered, Confirmed, Death) model. Here are some references on the subject.

Dynamic Models in Epidemiology [R package EpiDynamics version 0.3.1]. (n.d.). Retrieved from https://cran.r-project.org/web/packages/EpiDynamics/index.html

SEIR and SEIRS models. (n.d.). Retrieved from https://www.idmod.org/docs/hiv/model-seir.html#seirs-model

Finally, we tried time series regression, a topic that has yet been covered in any Computation and Data Science or Statistics courses at George Mason University. We referred to several online guides to make it work.

Chatterjee, S. (2018, February 5). Time Series Analysis Using ARIMA Model In R. Retrieved from https://datascienceplus.com/time-series-analysis-using-arima-model-in-r/

Hyndman, R. J. (2020, May 5). CRAN Task View: Time Series Analysis. Retrieved from https://cran.r-project.org/web/views/TimeSeries.html

kalyanikalyani 37911 gold badge33 silver badges44 bronze badges, DanDan 54333 silver badges55 bronze badges, & AlphaAlpha 37133 silver badges88 bronze badges. (1962, August 1). What are the values p, d, q, in ARIMA? Retrieved from https://stats.stackexchange.com/questions/44992/what-are-the-values-p-d-q-in-arima

Prabhakaran, S. (n.d.). eval(ez_write_tag([[728,90],'r_statistics_co-box-3','ezslot_3',109,'0','0']));Time Series Analysis. Retrieved from http://r-statistics.co/Time-Series-Analysis-With-R.html

sumi25. (2018, August 20). Understand ARIMA and tune P, D, Q. Retrieved from https://www.kaggle.com/sumi25/understand-arima-and-tune-p-d-q

Time Series Analysis: Building a model on non-stationary time series. (2015, August 15). Retrieved from

https://www.r-bloggers.com/time-series-analysis-building-a-model-on-non-stationary-time-series/

This website provides some basic visualization for the up-to-date total number of cases in the world and in each country.

Coronavirus disease 2019 (COVID-19): Epidemiology, virology, clinical features, diagnosis, and prevention. Uptodate.com. (2020). Retrieved 12 May 2020, from https://www.uptodate.com/contents/coronavirus-disease-2019-covid-19-epidemiology-virology-clinical-features-diagnosis-and-prevention.

These articles provide epidemiology, virology, clinical features, diagnosis, and prevention measures.

Coronavirus Disease 2019 (COVID-19) Situation Summary. Centers for Disease Control and Prevention. (2020). Retrieved 11 May 2020, from https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/summary.html.

This website provides the summary of the COVID-19 situation in the U.S. and the world. It also provides the timeline of the development of the COVID-19 pandemic and the efforts exercised to counter the disease.

WHO Timeline - COVID-19. Who.int. (2020). Retrieved 11 May 2020, from https://www.who.int/news-room/detail/27-04-2020-who-timeline---covid-19.

## Questions

What is the trend of the recorded cases and death worldwide? What are the factors which are correlated to the spread of the virus such that they could be used in the forecast?

Over the course of the project, we changed the direction of our question. What are the factors which we could include in our regression models for a more accurate prediction? Scientists and the health organizations claimed that certain factors such as weather and crowded events impact the spread of the disease. Could we use these factors in our models and see if they are reliable?

## Data

We used the Kaggle competition data and the data prepared and shared by other competitors for our project.

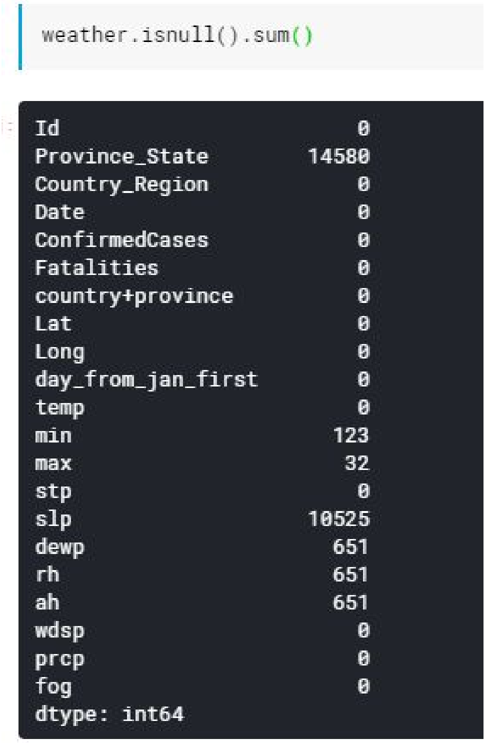
The datasets used include:

- COVID-19: confirmed cases and deaths in each country/region
- Country information: population, area, density, continent, latitude, and longitude

- Weather: average temperature, mean station pressure, mean sea level pressure, mean dew point, relative humidity, absolute humidity, mean wind speed, total precipitation, and fog.

Clean up of the data:

- Using the COVID-19 data, columns for the combined country and province, previous cases and deaths, cumulative cases and death, and days since the first cases are created for the model. The cumulative cases and deaths are changed to logarithmic scale for the linear regression models.
- The country information data uses “Americas” to represent both North and South America, so “Americas” are replaced with “North_America” and “South_America” according to the lists of the countries in each continent. Then a column for the combined country and province is created to use as a key to combine the train and test data to the country information data. The continents are changed to integer codes for regression.
- The weather data contains only up-to-date data and some unknown for some features. The unknowns are filled with the mean of the recorded data for each country. However, this method of dealing with the unknown faces problems when the country/province has no recorded data. The image below shows that there are many null values in the weather dataset.



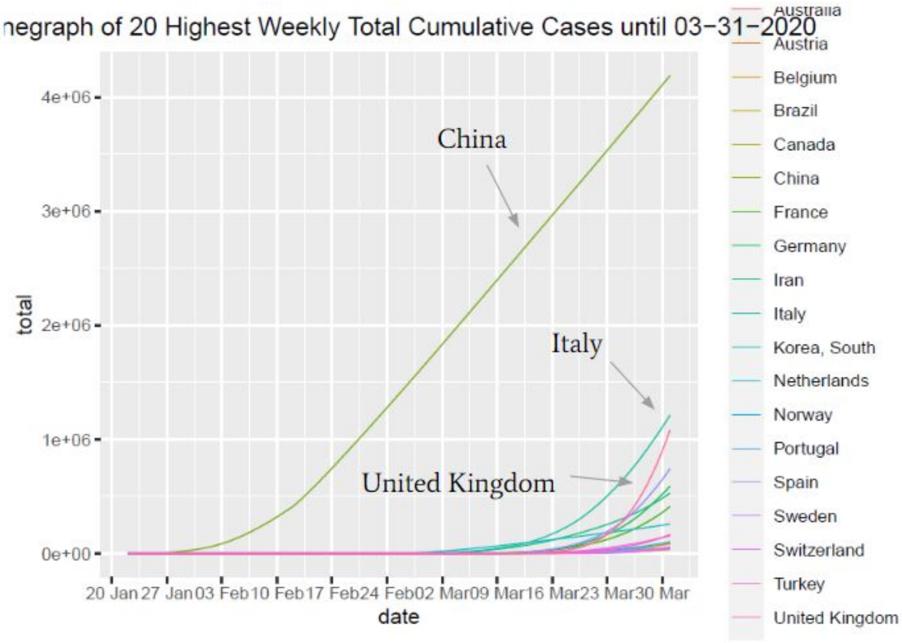
```
weather.isnull().sum()
```

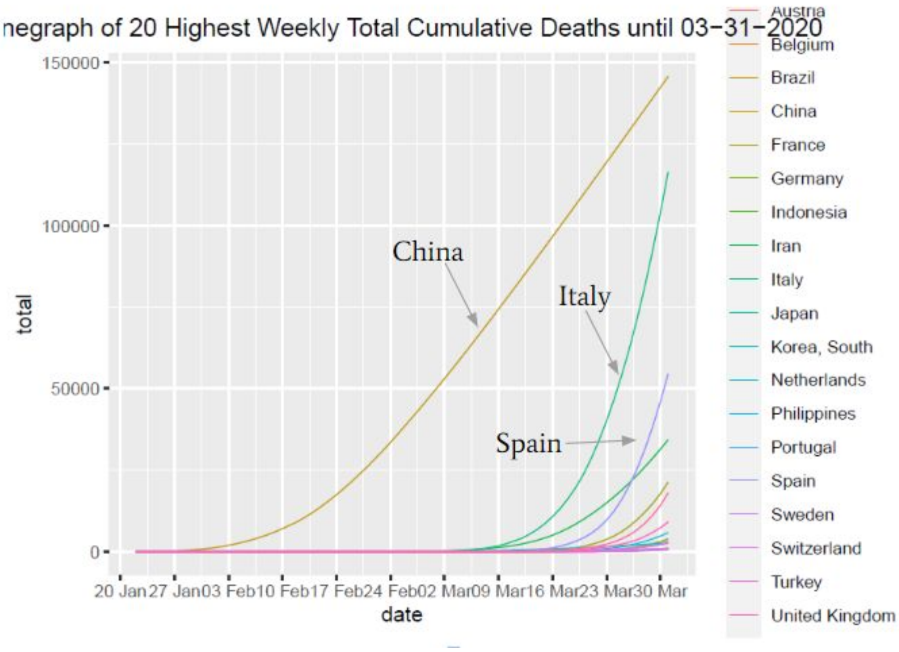
```
Id                       0
Province_State       14580
Country_Region           0
Date                     0
ConfirmedCases           0
Fatalities               0
country+province         0
Lat                      0
Long                     0
day_from_jan_first       0
temp                     0
min                    123
max                     32
stp                      0
slp                  10525
dewp                   651
rh                     651
ah                     651
wdsp                     0
prcp                     0
fog                      0
dtype: int64
```

- In order to perform time series forecasts, the training data is stationalized to fit the requirement of time series data, but the transformed data still does not fulfil the requirement.

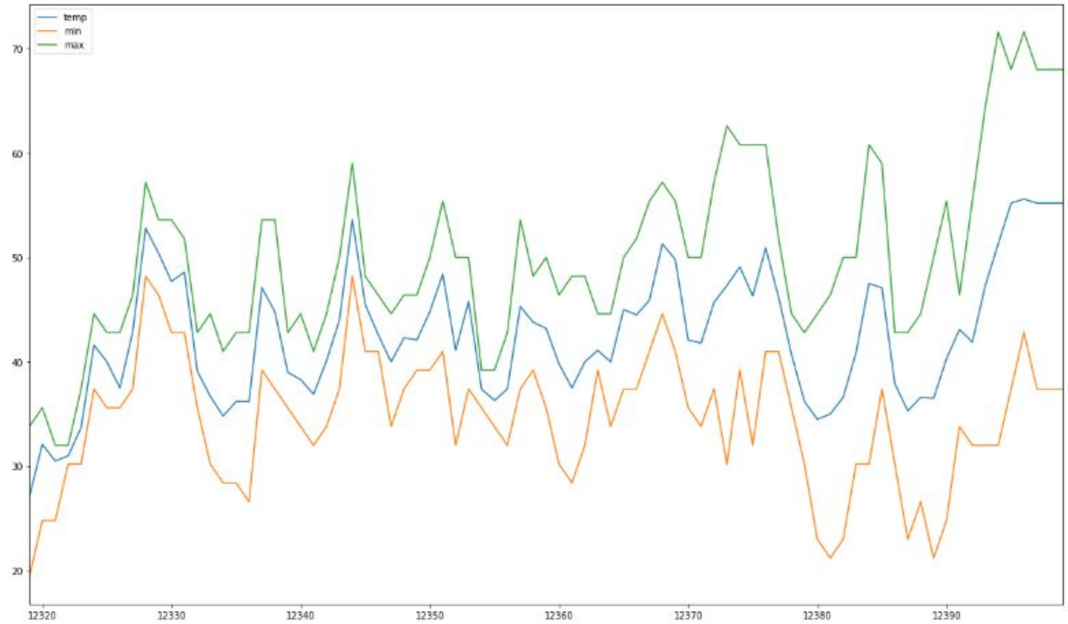
## Exploratory Data Analysis

The confirmed cases and deaths experience exponential growth. Therefore, it would be best to transform the cumulative cases to logarithmic scale for a linear regression model. However, the log-scale of the cumulative is not a simple straight line such that the gradient of the log-scale line changes as the days increase. Therefore, we speculate that we would need to fit a polynomial regression model to the logarithmic scale data.

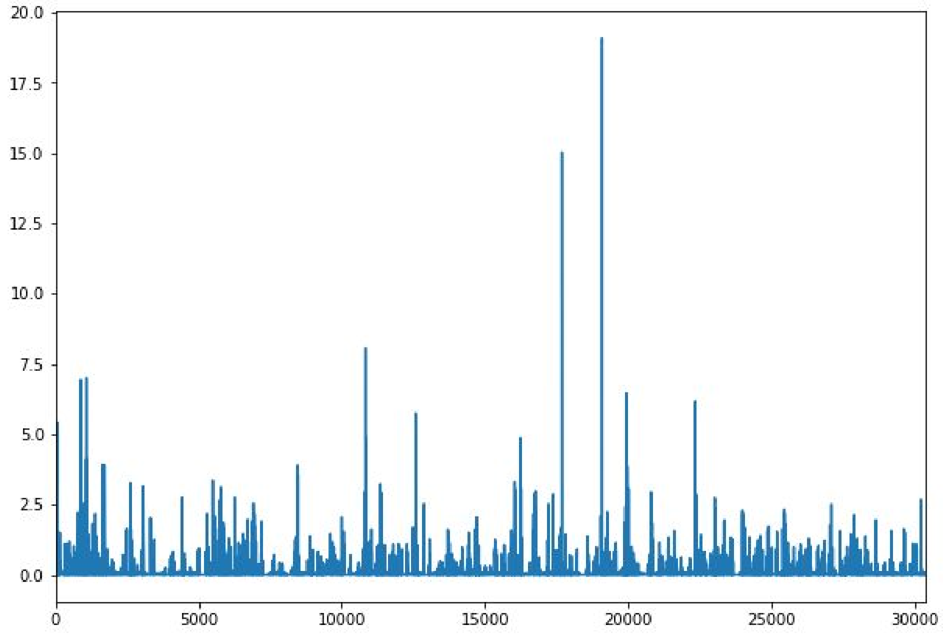




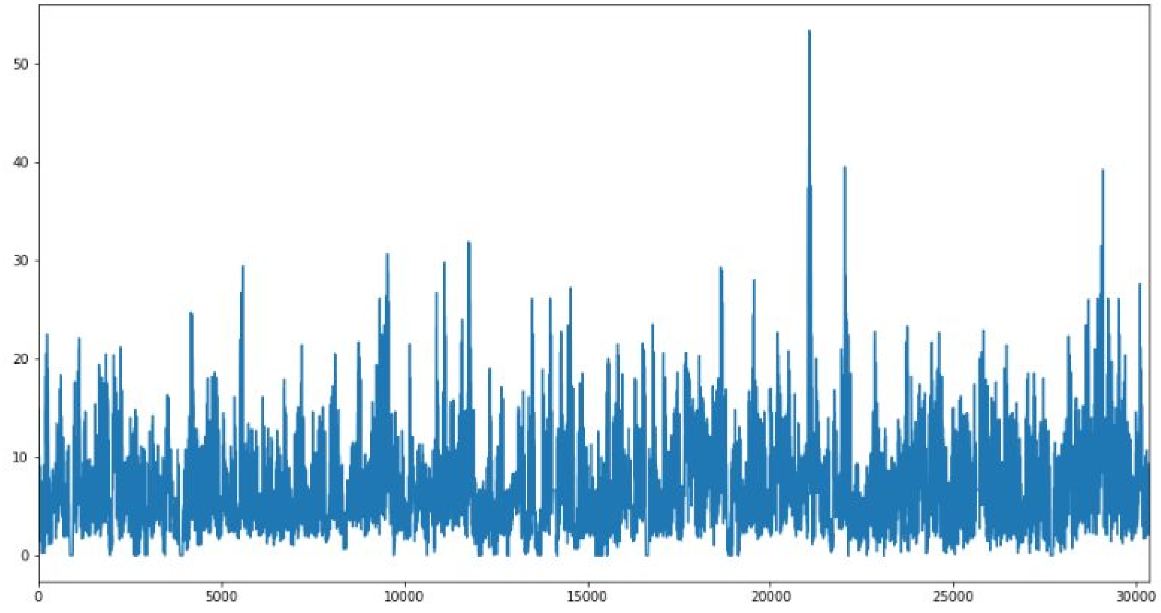
Visualizations were done to inspect some of the features of the weather data for a randomly chosen country Germany. The general trend of the temperature increases over the months. There does not seem to be a trend in the mean precipitation or mean wind speed data. Below is a plot of the minimum, average, and maximum temperature recorded in Germany from January 22 to April 11, 2020.



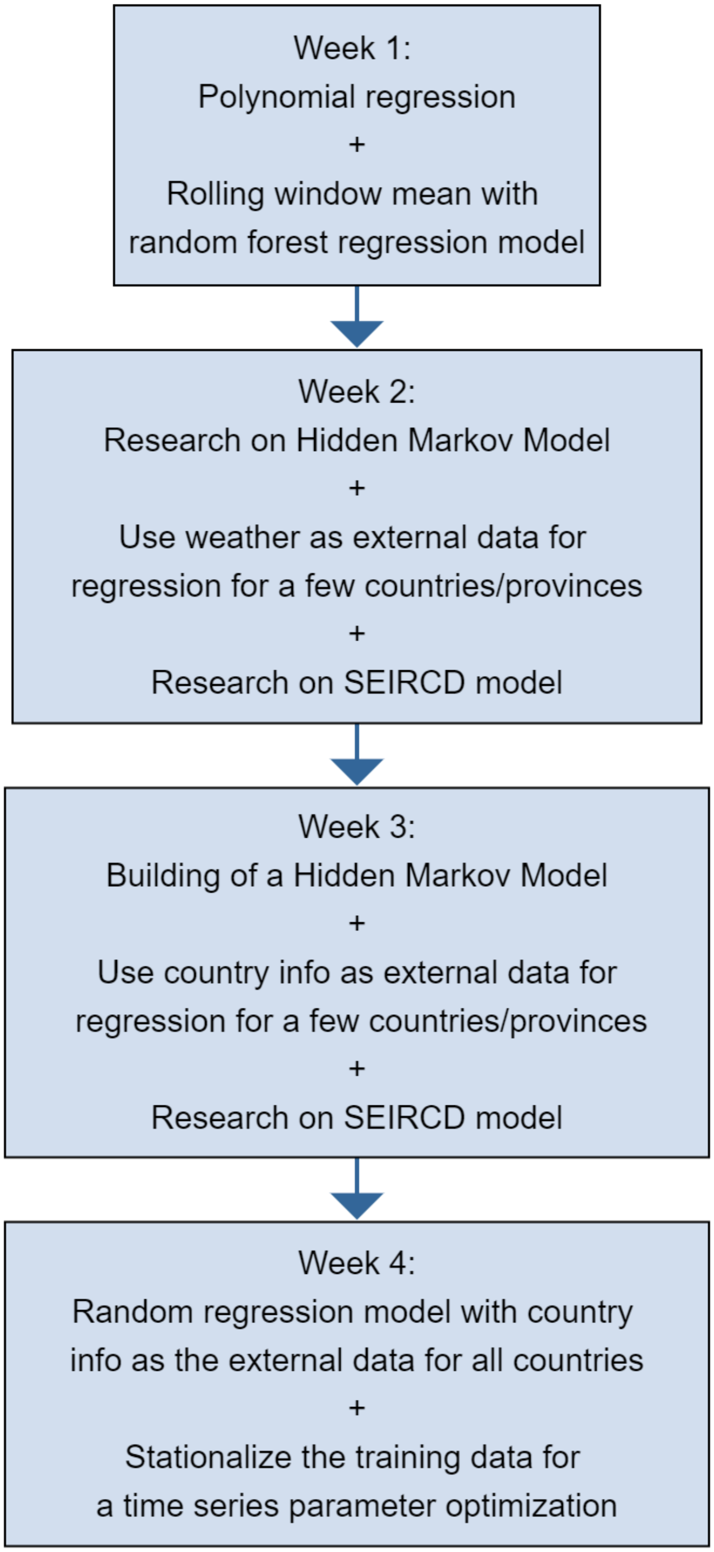
Below is a plot of the mean precipitation recorded in Germany.



Below is a plot of the mean wind speed recorded in Germany.



## Design Evolution



Week 1:
Polynomial regression
+
Rolling window mean with
random forest regression model

Week 2:
Research on Hidden Markov Model
+
Use weather as external data for
regression for a few countries/provinces
+
Research on SEIRCD model

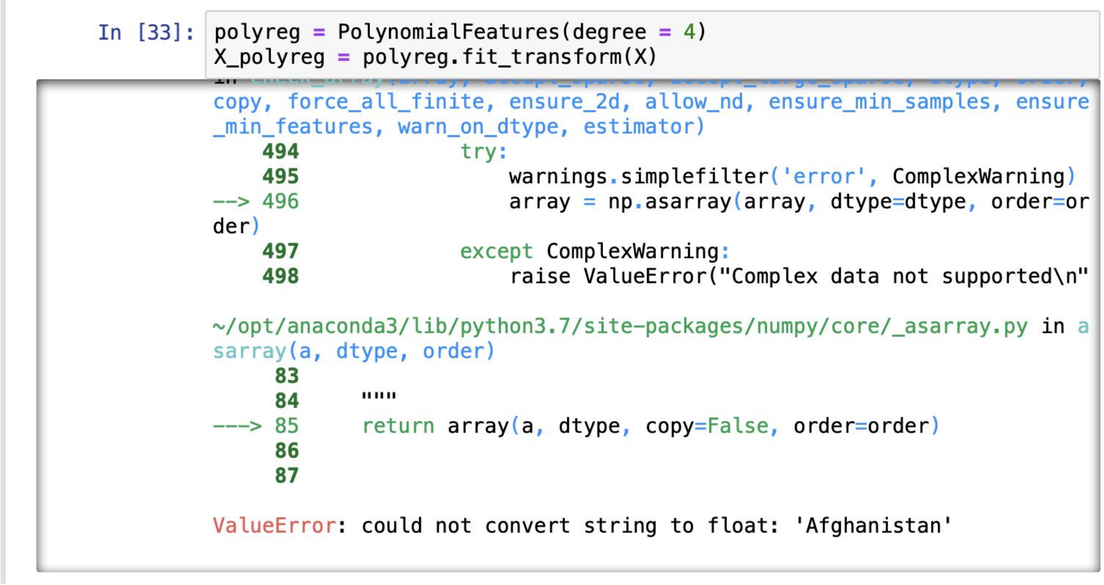
Week 3:
Building of a Hidden Markov Model
+
Use country info as external data for
regression for a few countries/provinces
+
Research on SEIRCD model

Week 4:
Random regression model with country
info as the external data for all countries
+
Stationalize the training data for
a time series parameter optimization

## Week 1

In the beginning, we weren’t sure on which model to begin with. We were familiar with fitting linear regression lines, so we wanted to start with that. After accessing the data, Felicia started with exploratory data analysis. She read into articles to decide other variations of linear regression and nonlinear regression that could be applied onto the dataset. In the meantime,  she created bar plots and made analysis on the results to determine which model to fit. Since the logarithmic scaled cumulative cases seems to have a nonlinear trend, we decided to give polynomial regression a try.

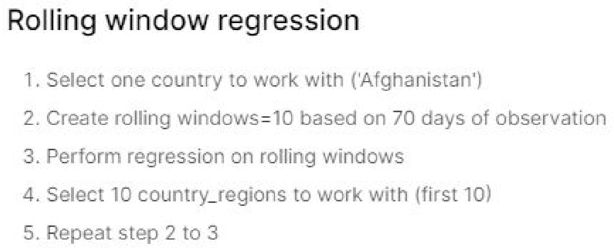
Lily and Deja were working on polynomial regression, based on the exponential trend shown by Felicia’s plots. Cumulative cases (*cumCases)* and deaths *(cumDeaths*) are features aggregated from the training data to be used in the model. One other issue that we ran into with creating the polynomial regression is that it was trying to convert the names of “Countries” into floats, so we were not able to get any predictions from that model either. We were all surprised to find out that there’s not many variables to work on, so the polynomial regression model was crossed off the list. The below screenshot is of an issue we had with trying to create the polynomial regression.



```
In [33]: polyreg = PolynomialFeatures(degree = 4)
         X_polyreg = polyreg.fit_transform(X)
```

```
in check_array(array, accept_sparse, accept_large_sparse, dtype, order,
copy, force_all_finite, ensure_2d, allow_nd, ensure_min_samples, ensure
_min_features, warn_on_dtype, estimator)
    494                 try:
    495                     warnings.simplefilter('error', ComplexWarning)
--> 496                     array = np.asarray(array, dtype=dtype, order=or
der)
    497                 except ComplexWarning:
    498                     raise ValueError("Complex data not supported\n"

~/opt/anaconda3/lib/python3.7/site-packages/numpy/core/_asarray.py in a
sarray(a, dtype, order)
     83
     84     """
---> 85     return array(a, dtype, copy=False, order=order)
     86
     87

ValueError: could not convert string to float: 'Afghanistan'
```

Then we attempted to use the rolling window mean as an explanatory feature for a simple random forest regression model. Unfortunately, the result was undesirable as the model predicts a very small number of cases for the first few countries that went under testing. Below is an outline of how we planned to approach the problem with the method.



Rolling window regression

1. Select one country to work with ('Afghanistan')
2. Create rolling windows=10 based on 70 days of observation
3. Perform regression on rolling windows
4. Select 10 country_regions to work with (first 10)
5. Repeat step 2 to 3

## Week 2

Due to a lack of explanatory features given in the original training data, we were unable to do any meaningful model building. Reflecting on our lack of progress in the first week, we decided that we should explore other methods besides regression models. Deja has looked into the usage of Hidden Markov Model in time series analysis, and Fecilia has looked into the popular SEIRCD model which many researchers use to model contagious diseases while Lily would continue to work on the regression model using the weather data as external explanatory variables.

In regard to the start of the HMM for our project, this week Deja spent primarily on research for the Hidden Markov Model and the theory behind it. This included watching videos and reading articles that discussed how one might implement an HMM on their data. Since we wanted to predict future cases and deaths, those were our hidden states for the model. Deja also was trying to figure out what the possible transition probabilities might be as well. While Deja was working on resolving this, she also looked into other areas of the project to assist with.

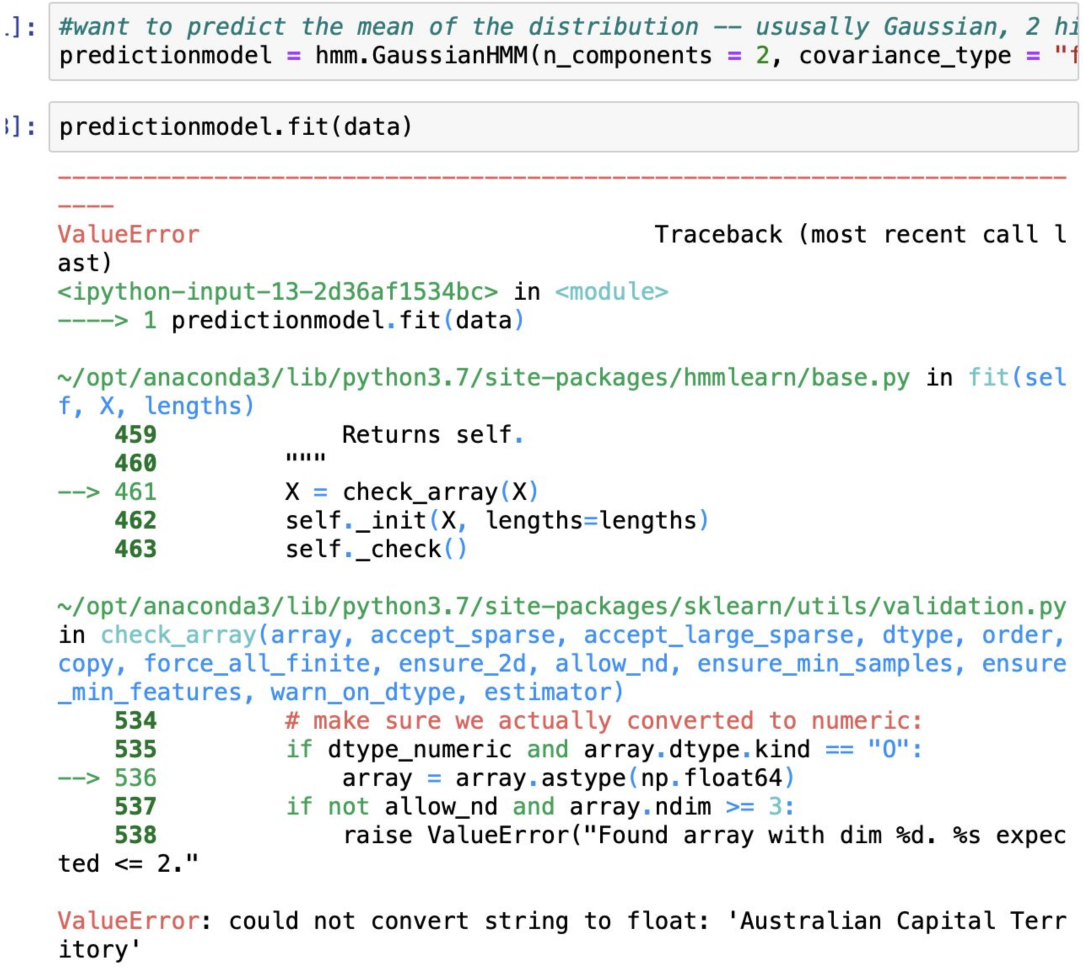
Felicia began reading into SIRS to get herself familiarized with SEIRCD. She explored packages for SEIRCD and mathematical formulas (coding manually). She spent some time considering how to transform the data so that the package may run. She theorizes that the SEIRCD might not even work because of the difference in parameters. She also worked on finding packages for HMM and ruled out possibilities of what other variables may help the polynomial model work better. The SEIRCD model is thought to be the ideal one to simulate the spread of a disease. The model is more commonly known as SEIR with vital dynamics.

Lily researched the discussion posts in Kaggle about available external datasets and potentially useful features discovered by other competitors. She found a weather dataset contributed by one of the Kaggle competitors. The weather data required cleaning regarding the missing data filled with dummy values and the NA values. Previous cases *(prevCases)* and deaths *(prevDeaths)* were created from the training for model training.

## Week 3

We continued to work on the respective models from last week.

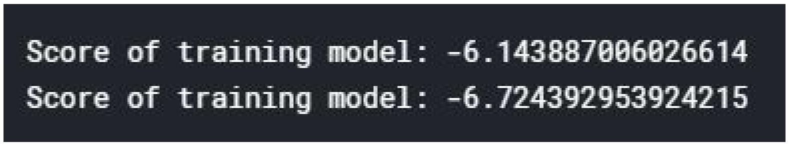
Deja's HMM model building: Deja continued trying to work on creating an HMM. A Gaussian prediction model was the plan, but Deja also looked into a Multinomial prediction model as well. Some starter code was created, but there were a couple roadblocks in trying to get the code to run. One of the issues included an issue recognizing one of the variables we wanted to use and running into a similar issue with our polynomial regression where names of "Countries" were not being converted to floats for the model. Due to these early roadblocks, Deja wasn't able to train the data. She reached out for outside assistance, but was not able to make much progress as she had hoped. An example is provided below of one of the errors from the HMM code.



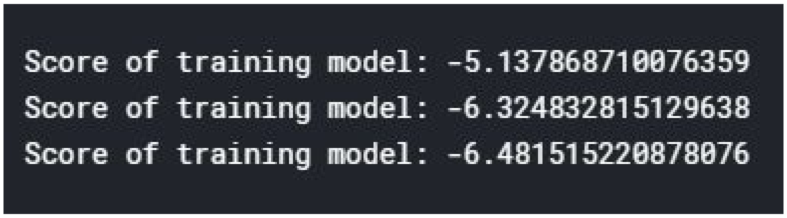
```
.]: #want to predict the mean of the distribution -- ususally Gaussian, 2 hi
    predictionmodel = hmm.GaussianHMM(n_components = 2, covariance_type = "f

]: predictionmodel.fit(data)

    ---------------------------------------------------------------------------
    ----
    ValueError                                Traceback (most recent call l
    ast)
    <ipython-input-13-2d36af1534bc> in <module>
    ----> 1 predictionmodel.fit(data)

    ~/opt/anaconda3/lib/python3.7/site-packages/hmmlearn/base.py in fit(sel
    f, X, lengths)
        459             Returns self.
        460         """
    --> 461         X = check_array(X)
        462         self._init(X, lengths=lengths)
        463         self._check()

    ~/opt/anaconda3/lib/python3.7/site-packages/sklearn/utils/validation.py
    in check_array(array, accept_sparse, accept_large_sparse, dtype, order,
    copy, force_all_finite, ensure_2d, allow_nd, ensure_min_samples, ensure
    _min_features, warn_on_dtype, estimator)
        534         # make sure we actually converted to numeric:
        535         if dtype_numeric and array.dtype.kind == "O":
    --> 536             array = array.astype(np.float64)
        537         if not allow_nd and array.ndim >= 3:
        538             raise ValueError("Found array with dim %d. %s expec
    ted <= 2."

    ValueError: could not convert string to float: 'Australian Capital Terr
    itory'
```

Felicia continued to work on SEIRCD but has removed it from the possibility of getting it modeled. She made short reports on why the model wouldn't work on the given dataset and decided to look for alternatives. She decided to model time series. She tried Holt-Winter model (exponential time series) and univariate time series. She spent some time reading on time series and how it works since she has never had it modeled before. This week, the progress was slow due to a cramped school schedule.

After testing the first five countries with the cleaned weather data as explanatory features, the scores for these models turned out to be negative. Lily aimed to train a regression model for each country/province, and train separate models for predicting the cases and deaths. There was also a problem with the availability of the weather data. Since the actual weather data is only up-to-date, there is no accurate future data for predicting the cases and death in the feature. Even if we use the weather forecast data, we could only obtain the data for a week in advance. Moreover, the uncertainty in the weather forecast data would introduce more bias and error in the regression model.



```
Score of training model: -6.143887006026614
Score of training model: -6.724392953924215
```



```
Score of training model: -5.137868710076359
Score of training model: -6.324832815129638
Score of training model: -6.481515220878076
```

Lily researched Kaggle discussions again and found a dataset which contains information for countries, including population, continents, area, density, longitude and latitude. The country info data is cleaned to have “North_America” and “South_America” to represent “Americas”. Then the data is combined with the training data to prepare for the prediction next week.

### Week 4

Lily continues to implement the country info data to the training data. She developed a pipeline which predicts the log-scaled cumulative cases and deaths day by day and feeds the predicted data as the previous cases and deaths for the next day. The pipeline also converts the logarithmic scale data back to linear scale. This pipeline is coupled with a random forest regression model. The pipeline and the model is tested for the first country, but the predicted cases do not seem to follow the general increasing trend. The pipeline and the model is then tested for all the countries/provinces but takes a very long time to run. It is unsure if the model runs into an infinite loop or the model actually requires a long process time due to the large number of unique countries/provinces.

Felicia decided to pursue working on time series and close the project for SEIRCD. Progress was slow this week due to her  cramped schedule. She found out that the SEIRCD has parameters which values may differ between countries. These parameters are:

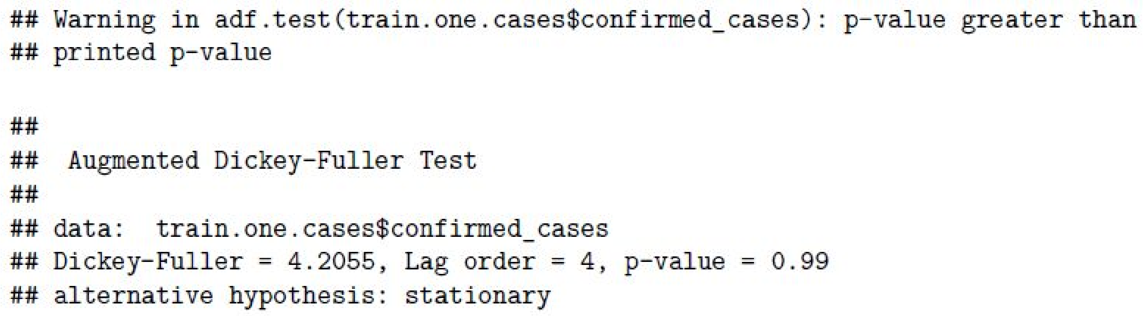
- The incubation period which will then affect sigma, the 1/incubation period.
- Gamma which is 1/the duration of the contagiousness.
- Ro which is the reproduction number.
- Beta which is infectiousness rate that can be derived from Ro*gamma.

She has the reason to believe so because each country's parameter may differ based on the Gross Domestic Product (GDP). The GDP provides an economic snapshot of a country, used to estimate the size of an economy and growth rate.Theoretically, a country that's well off will handle the virus better due to better healthcare facilities. The population density may also contribute to the spread of the virus. After many hours surfing the Internet, she could only find parameters for Wuhan, China, but not for any other countries. Furthermore, the data provided by Kaggle lacks the record for susceptible, exposed, infectious, and recovered. Thus, further data aggregation is needed. Last but not least, the package provided by R, EpiDynamics, is limited to SIR, SIRS  and SEIR. Thus, if she were to model SEIRCD, she needed to use its original mathematical model and its derivation, plus a method of choice to simulate the spread as a function of time such as Newton's Method. There was also an unseen caveat using the SEIRCD model, which was the fine tuning had to be done manually. Since each country has different parameters and starting population, those numbers had to be entered one by one as creating an autonomous pipe would take too much time.

After finding the deadend, she decided to work on a time series model, which she thought was impossible to work on at first due to not having enough statistical background. She found a package in R that specifically for time series, *tseries* and *timeseries*. Before fitting the model, she had to check for stationary assumptions.

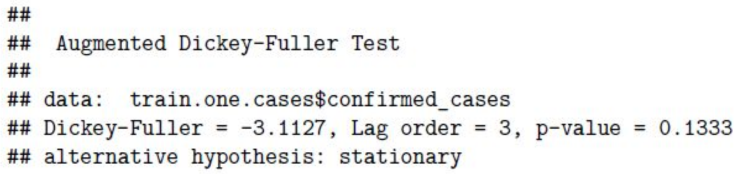
Dickey-Fuller Test of Stationary  
Ho: The data is not stationary (explosive).  
Ha: The data is stationary.

She tested the train data for Afghanistan to find that the Dickey-Fuller Test of Stationary yielded a p-value of greater than 0.99, thus she failed to reject the null hypothesis



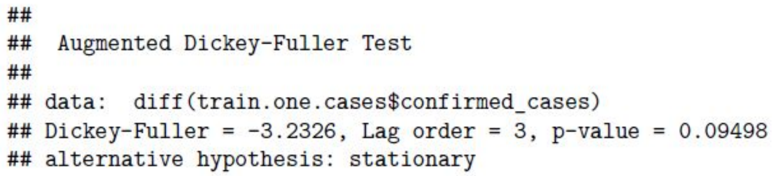
```
## Warning in adf.test(train.one.cases$confirmed_cases): p-value greater than
## printed p-value

##
##  Augmented Dickey-Fuller Test
##
## data:  train.one.cases$confirmed_cases
## Dickey-Fuller = 4.2055, Lag order = 4, p-value = 0.99
## alternative hypothesis: stationary
```

She performed log transformation and difference transformation on the training data to handle the problem. The log transformation yielded a p-value of 0.133. Again, it's greater than default alpha of 0.05, so she failed to reject the null hypothesis. However, the p-value was a lot smaller so she had the reason to believe that the transformation did good for the data.



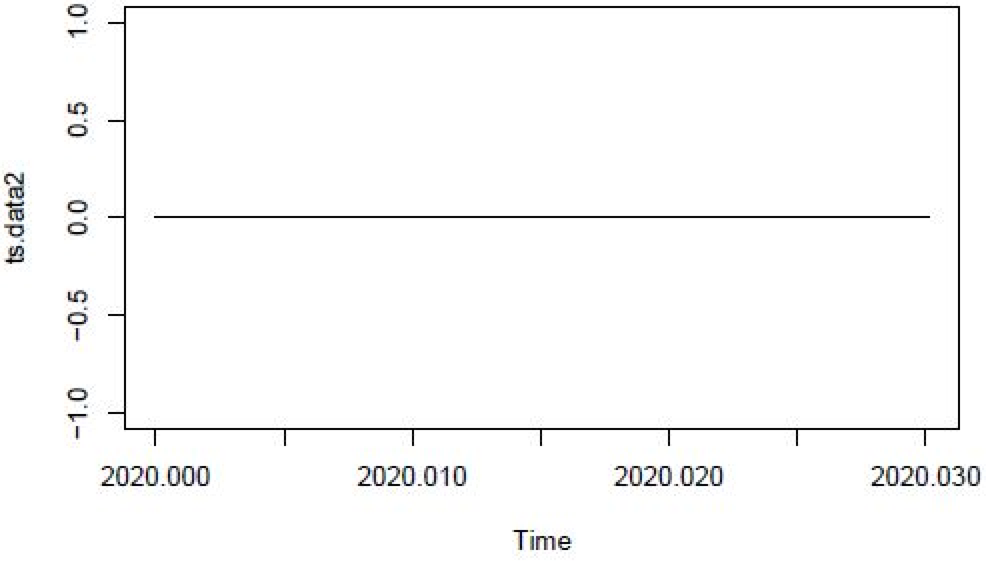
```
##
##  Augmented Dickey-Fuller Test
##
## data:  train.one.cases$confirmed_cases
## Dickey-Fuller = -3.1127, Lag order = 3, p-value = 0.1333
## alternative hypothesis: stationary
```

She added the difference transformation on top of the log transformation to lessen the p-value.

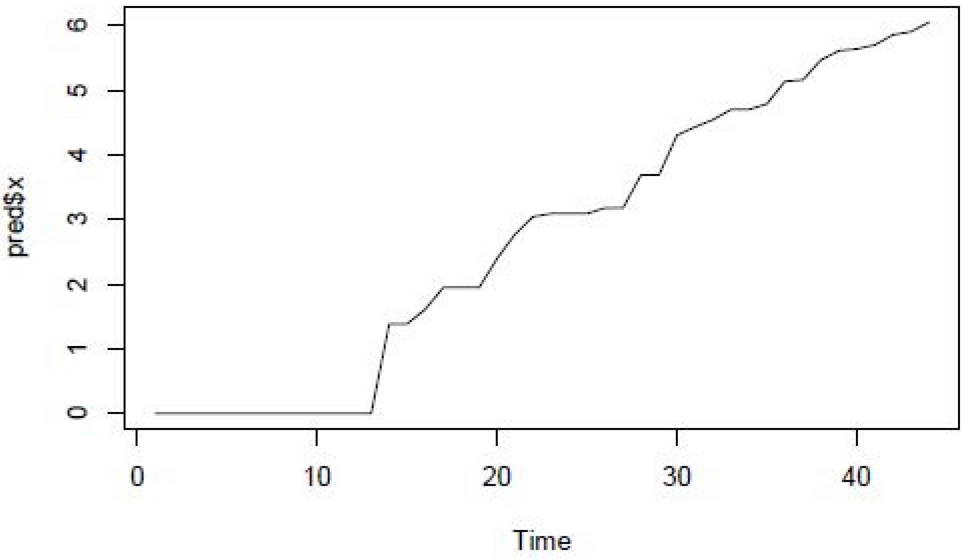


```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(train.one.cases$confirmed_cases)
## Dickey-Fuller = -3.2326, Lag order = 3, p-value = 0.09498
## alternative hypothesis: stationary
```

The p-value yielded from the test is 0.09498, which was still greater than the default alpha of 0.05. Thus, she failed to reject the null hypothesis. In the end, after two transformations, the data wasn't stationary enough for her to fit a model. She believed that the reason behind this nonstationary issue stemmed from the lack of data. For the log transformation to work, she had to remove observations with zero values, otherwise it'd yield infinity error. Thus, she concluded that the model for each country wasn't always going to be reliable, especially with countries that have more zeros in its observation. She tried to predict it anyway using the best model possible, but there was an error that made the line flat.



She moved on to creating a prediction using Holt-Winter's method. She did the same transformations and plotted the prediction.



However, it couldn't predict over 10 predictions. Thus, she led to believe that the model wasn't appropriate or she had lacked data. The main concern for these two time series models was that it wouldn't be able to capture the drastic changes in number of cases or deaths like SEIRCD would.

Deja continued working on the HMM this week. She was able to find some documentation, but the main issue this week was the realization that transition probabilities would need to be defined before trying to train the model and get a prediction. Extending from the prior week, there  was also the issue of trying to figure out how to implement our time series data into the model as many of the examples online during the research phase showed arrays that were being created from scratch rather than developing an HMM using a dataset. As we wanted to focus on predicting for individual countries, that would result in separate transitional probabilities needing to be calculated and made, but there is not enough data to determine that with accuracy. There is still the issue with the code wanting to convert the countries to floats, but it was determined that the transitional probabilities would need to be made first before proceeding on to try to train the model. Similar to the reliability issue mentioned in the SEIRCD model, incorrect transitional probabilities in the beginning of the model could lead to incorrect predictions if the model could run and make predictions.

## Implementation

We explored various models from regression models to Hidden Markov Model, and from SEIRCD model to time series data analysis. We met deadends and had to manage the limited resources, research, and data for the project. Although suboptimal, we decided to use the regression model with external data as our final implementation. The implementation of this method includes data transformation, pipeline building and testing of several regression models.

The external data used is the country information which includes population, area, population density, continent, longitude, and latitude. We chose this dataset because the data does not change over a short period of time whereas the weather data changes rapidly everyday. The train and test datasets are combined with the country info datasets.

The explanatory variables used in the current model are as follows:

- DaySinceFirstCase: the number of days since the first case occurred in the area
- log_prevCases, log_prevDeath: the logarithmic scaled cumulative previous cases and deaths from the day before.
- lat: latitude
- lon: longitude
- continent_code: continents of the country/region which is transformed into integer code
- population: population of the country/region
- area: area of the country/region
- density: population density of the country/region

The exploratory variables used are as follows:

- log_cumCases: logarithmic scaled cumulative cases of the day
- log_cumDeath: logarithmic scaled cumulative deaths of the day

```
predictors     =['DaySinceFirstCase','log_prevCases','log_prevDeath',
'lat', 'lon', 'continent_code', 'population', 'area', 'density']
```

```
targets = ['log_cumCases','log_cumDeath']
```

The targets are predicted separately for each country/region day by day using two models of the same kind. A pipeline is constructed to perform this task. The pipeline trains on the days where all the explanatory features are available and predicts the targets for the next day. The pipeline then updates the predicted results of that day as the *log_prevCases* and *log_prevDeath* data of the next day. The updated data becomes part of the train dataset. The models are trained again on the updated train dataset and predict the targets of the next day. This process repeats until the models predict the targets for all the days in all countries/regions. The logarithmic scaled cumulative cases and deaths are then transformed to linear scale. The difference is taken to find the daily new cases and deaths.

We tried three types of regression models, including linear regression, random forest regression, and ridge regression.

The result of this approach is not optimal because there were difficulties building a functional pipeline. The previous attempts took a long time to run, and was unsure if it was due to the complexity of the model or the possibility of an infinite loop in the pipeline.

## Evaluation

The project was challenging to begin with. We were inexperienced with building models from scratch, much less machine learning algorithms. We didn't know what we were dealing with until we did further read on the data's characteristics. One of the characteristics of the data is its stochastic time series nature, meaning that the current data point depends on the data point in the previous time step. We spent a considerable amount of time reading articles and how-to's about ways to deal with stochastic time series data from the Internet. After thorough exploration, we believed that SEIRCD was the appropriate model to simulate COVID-19. The only issue was finding the correct parameter for each country.

In this project, we aim to investigate explanatory variables which could affect the spread of the virus, as claimed by the scientists and health organizations. Since there was not enough time or resource to study the virus thoroughly, the published studies may have flaws which were not detected in the hasty peer-review process. Therefore, we were interested to see the effectiveness of the proposed explanatory variables in predicting future COVID-19 cases and deaths.

Despite our incomplete solution, we were able to determine that the up-to-date weather data would not be a suitable variable in predicting future cases because the weather forecast may not be accurate. It would increase the uncertainties in our model which already does not have enough explanatory variables to work with. We also study how the country information, such as population density, would affect the model prediction.

Unfortunately, we were unable to get an output from our regression model due to potential errors. Several test runs with a small subset of the countries did not return results coherent to the general trend of the data. We conclude that our regression model is not optimal. Since the regression model is the only model which uses external data as explanatory variables, there is no comparison of performance between different models.

Given more research done on this topic, we might be able to find more explanatory variables to test our model. We would also like to improve the complexity of our model. This project gave us more insight on what’s required out of a data scientist and we will continue to learn from this project as an individual to sharpen our skills.