MACHINE LEARNING PROJECT REPORT

COVID-19 CASES PREDICTION – A TIME SERIES PREDICTION USING MACHINE LEARNING APPROACH

Yumo Yao | Wanyi Yang
Prof. Li Guo
CSCI-SHU 360 Machine Learning
December 18, 2020

ABSTRACT

Our project aims to predict the number of cases that will break out in the future based on the number of COVID-19 cases already available. We study the impact of case base, population density, GDP level, economy, political constraints, and policies on the spread of the virus in several major countries to predict the number of future case growth. It provides information for the government to issue restraint policies during the epidemic and alleviates the fear and anxiety about the unknown of the virus. We find data sets that quantify various policies and apply multiple machine learning models to make predictions. From a broad perspective, we applied two approaches to solve the time series problem, one is to add time as features inside the attributions of the prediction, and the other is to use a neural network structure for the memory of historical data to solve the problem. The first data approach uses five models, and for the second data approach, we use two types of RNN. finally, we use the part of the divided data for validation.

Key Words: COVID-19, Public health, Prediction, Cases, Linear Regression, Ridge Regression, KNN, Random Forest, Regression Tree, LSTM, Time series

1. Introduction

The Covid-19 pandemic, also known as the coronavirus pandemic, is an ongoing pandemic firstly detected in December 2019 in Wuhan, China. The Covid-19 epidemic quickly swept across the globe, bringing tremendous impact on the economy, politics, and national life of various countries. As of today, there have been 73.6M cases worldwide, of which 41.7M have recovered. And unfortunately, there have been up to 1.64 deaths worldwide. In order to alleviate the Covid-19 epidemic as soon as possible, governments have called upon many resources and introduced relevant policies. These include requirements for social distance, the mandatory wearing of masks, restrictions on rallies, school closures, etc.

In order to be able to better control the outbreak, forecasting the number of future cases is crucial. Only then can appropriate policies be implemented and national resources be rationally deployed. In this project, we selected a small number of representative or influential countries. Based on the data such as their national status, policy implementation, past cases, and other data, we tried to predict the number of cases in the coming day.

2. Data Set and Features Explanation

Our data are primarily derived from COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University and Oxford Covid-19 Government Response Tracker. The two data-sets above provide the most important features in this project, which are the specific number of cases and deaths and the various policies per region per day. In addition to this, we added some long-term features of the countries (i.e., features that did not change or did not change much during the epidemic) from the government official websites, such as values of GDP per capita, aging, population size, and monthly temperature. We selected epidemic data from the United States, China, India, France, Italy, Brazil, and the United Kingdom because of the high representativeness or specificity of these countries.

There are around 330 observations per country. Each observation contains new cases and new deaths for that country on that day, and also includes many different features from the data-set mentioned above. We manually combined multiple data sets simply on a time-based basis. There are nearly 65 features in total. Due to the specificity of the epidemic, the data is highly time-series.

3.Data Pre-processing

We noticed that there are some missing data in the dataset. To improve the accuracy of the prediction, we did not simply fill in with averages or zeros. Instead, we went to official government websites or other reliable data sources to find news or data on the day of that missing data.

In addition to this, we also noted some data with more obvious errors, such as a negative number of new deaths. After discussion, we decided to keep this part of the data as the noise of the original dataset.

Due to the differences in the models used afterwards, we used a total of two datasets. These two datasets are from the same source, but are processed in a slightly different way. They are Combined data set for all countries and Separate data set for each country, respectively.

a. Combined data set for all countries

For a part of the model, we used a more naive approach to preprocess the dataset. This dataset includes data from the seven countries mentioned above. We added five features directly after the original dataset, which are the number of cases in the past five days.

b. Separate data set for each country

For some other models (i.e., 2 LSTM models), the dataset is country-based. So there are seven tables in total, each showing data for one country.

To filter out the insignificant features and select useful ones for the predictions, we first manually removed insignificant features, such as the flagged item after each policy. The flag is to indicate whether each policy is targeted or general, whereas, in most observations, the flag was left blank, showing there is no data. In addition, the original dataset from Oxford University has indices, i.e., different policies quantified by formulas. However, considering that the weight of different policies may be separate, using indices directly may lead to a decrease in precision, so we removed the columns of indices.

4. Model Explanation

After finishing all the preliminary data processing, we started our training on this regression problem by using five different models. Explicitly speaking, we tried Linear Regression, Ridge Regression, KNN, Regression Tree, Random Forest, multi-layer LSTM with single feature, and single-layer LSTM with multiple features to predict the number of cases for the up-coming day. For the first five models, we use the first dataset, in which we treat time (the number of cases in the first five days) as data features, and randomly shuffle all the countries we studied to make predictions. In the latter two models, we try to separate each country and predict the trend and case number of each country separately by using the memory of historical data to solve the time series problem. We used the model with the best performance in the first four models as a baseline and improved our neural network method based on that.

We use the mean absolute error to calculate the error of the prediction. Since our project is intended to solve the regression problem, we cannot use the accuracy rate to judge the goodness of the model. We have tried squared absolute error and found that this calculation method is too tolerant of errors. The error calculated in this way is too large and thus tends to lead to a wrong judgment of the model.

Before starting to apply the model, we need to train test split the data set. We considered using cross validation, but since the LSTM model used subsequently contains temporal information, it is not possible to group observations randomly. Therefore, it would be difficult to compare the results of different models if we use cross validation.

Combined data set for all countries

Integrating all the data together allows us to compare the impact of different features on the virus transmission rate between countries.

Linear Regression : We first used a linear regression model to make predictions about the number of future cases. Linear regression is a relatively simple model that we can use to make a rough prediction of how well the data was processed. Considering the low complexity of linear regression, we did not use principal component analysis to prevent overfitting. The result is better than random guessing and the MAE is 0.01307.

Ridge Regression : We then applied Ridge Regression since we thought there would be some unimportant factors and correlated attributes that need to influence the outcome. Ridge Regression penalty the parameters of more important attributes that can solve this problem. The result turns out to be slightly better than just implying Linear regression, where the MAE is 0.01607.

KNN(Regression) : Before we used the KNN model, we screened all the data with PCA for the 24 best attributes, sifting out the ones with strong correlations. then we implemented the KNN model and we found the best results corresponding to a number of neighbors of 7. The result of training this model is an MAE of 0.01339. This is the best result we got from training all the models, so we use this result as the baseline.

Random Forest : We also used a random forest model with different nodes and tried to find a smaller error value by changing the node more or less by this. We think this may be a good way to predict since one random forest contains many small decision trees, and we consider it robust and accurate. We tried many values, but the results were somewhat unsatisfactory, the best node resulting in 0.04415, a larger error than even the previous simple linear model. We guess it is because our dataset is small and does not reach enough volume to make the random forests run similar trees, resulting in a large error. So, we then tried Regression Tree with less complexity and trying to see which one has better results.

Regression Tree : Then we applied the Regression Tree model. By trying different tree depths, we found that the max depth of the original data and the PCA selection data are 6 and 10, respectively. The error of this Regression Tree model is better than the Random Forest model, which is 0.1076 and 0.01602, respectively to original data and PCA selection data. However, we find that the max depth for the data after the PCA-picked attributes is larger, so we suspect that the model is overfitting with the PCA selected data.

Separate data set for each country

Long Short-Term Memory (LSTM) is a type of recurrent neural network that can learn the order dependence between items in a sequence. LSTMs have the promise of being able to learn the context required to make predictions in time series forecasting problems, rather than having this context pre-specified and fixed. Considering the ability of LSTM to remember and learn longer past inputs, we believe that this model can solve the problem better.
multi-layer LSTM with a single feature : We used a relatively simple univariate LSTM forecast to try to estimate the number of cases in the coming day. In this model, the only data is the number of cases per country per day. For each country, we first divide the data into a train set and a validate set. Then we can "slice a window" in each of the two datasets to obtain a certain number of observations. In this prediction, we took the number of cases in the first 9 days as the X value to predict the number of cases on the 10th day. We generated a series of observations from the already divided train set and validation set, respectively.

After this, we generate an LSTM model. We used a multi-layer model, which is intended to handle existing data better. The model structure is showing in **Figure 1**. However, considering the small amount of data for each country, using such a complex model may lead to overfitting, we dropout a portion of nodes after each layer to prevent potential overfitting. The exact structure of the model is shown in Fig. This LSTM performs extremely well, with a significant reduction in the error corresponding to each country. This is probably because LSTM is able to learn the past data well. And it happens that in the present prediction, the change of data over time is very obvious.
single-layer LSTM with multi-features : Then we used single-layer LSTM with multi-features so that the model can contain more information, such as changes in the prevention policy. The data processing required for the LSTM model is a bit different again. Like other models, we first rescale the data and perform a train test split in a certain proportion of temporal order, except that we also make the data set sequential so that it can be put into the LSTM. article caption graphicx, subfig Then we use the training set to train a one-layer LSTM and use it to predict the validation set. We found that when using this method, the error of the data predicted by each country varies greatly, and the error is much larger than that of other methods. We believe that the former is mainly due to the fact that the outbreak situation varies from country to country. For example, in China, there are more cases at the beginning of the outbreak and the number of new cases is almost single-digit in the later stages, while in the United States, there are fewer cases at the beginning and the number of cases increases all the way afterward. To solve this problem, we adjusted the model parameters, such as the number of nodes and epochs, separately for each country in order to avoid overfitting or

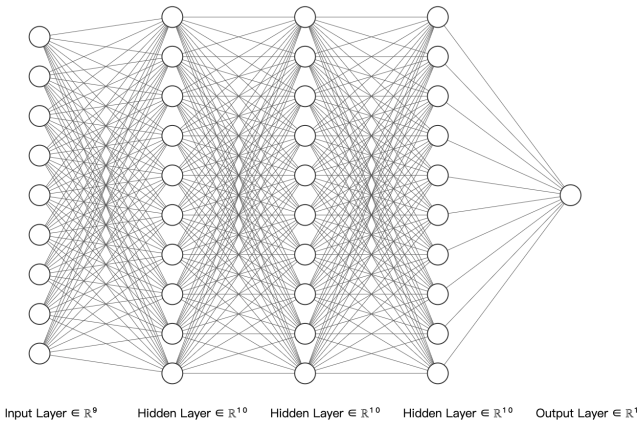

Figure 1: LSTM

underfitting as much as possible (we can judge the training situation based on the plotted train loss and validate loss). For the latter, i.e., the overall error is much larger than that of other models, we believe it is due to the different pre and post epidemic outbreak situations. We considered using the "slide a window" method, but considered that 1) previous models have used this method; 2) "slide a window" may emphasize local data and thus lead to overfitting; 3) the loss of the train set is convergent and thus can be used for the training. loss is convergent and thus can be optimized, so this method was finally abandoned. In this method, the performance of Italy and the UK is very poor, which means that the data sets of these countries are not suitable for this model. We considered the use of more advanced neural networks such as wide and deep neural networks, but unfortunately, this was not possible in the end. This issue will be explained in more detail in the reflection and feature improvement section of this paper.

Discussion

After trying two different data-sets and many different machine training models, we found that the best model is the multi-layer LSTM with a single feature. the following table shows the average error of each model. Without specifying width for last column:

	LR	Ridge	KNN	RF	Tree	LSTM(single)	LSTM(multiple)
Training MAE	0.0111	0.0113	0.0098	0.0858	0.0149		
Validate MAE	0.0131	0.0161	0.0134	0.0442	0.0160	0.0218	18.972

In this table, we can see that KNN and Linear Regression get better results, and the single feature LSTM with only time also significantly outperforms the multiple feature LSTM model. Due to the limited amount of data and a large amount of missing data for each country, applying overly complex models may easily cause overfit, and other models have their own problems. However, we can see that simpler models may give more satisfactory results when facing these complex predictions.

Reflection and Future Improvements

It means a lot to us to be able to apply what we have learned in class in our project and try to use it to solve a problem that concerns all of humanity. The process of completing the project had many twists and turns, from the initial decision on the project topic to the final conclusion, we struggled with it over and over again, finding problems and trying to solve them. For the prediction of the covid-19 epidemic, we have only completed a very naive part of the project. We are also well aware that there are many variables that affect the number of cases but are not taken into account, and that there are more complex and appropriate models and optimization methods. In addition, epidemiological projects are not just about analyzing and predicting data, but also require a certain amount of biomedical theoretical support. The following are some of the points that we think the current project needs to be improved.

The first major concern lies in exactly how many days of data should be used to predict the cases of a new day. Unfortunately, when we started the project, we did not delve into this issue. So for using the first dataset, we simply added the number of cases for five days to the dataset, while for using the multi-layer LSTM, we used the number of cases for the first nine days to complete the prediction. We believe that we can control for other variables and decide the best number of days to add to the dataset by changing the number of days in the prior sequence and observing how good the model is.

Another problem for us is how to divide the data set into a training set and validation set when using LSTM to predict time series problems. Since the number of cases does not always follow the same model (e.g., continuously increasing or continuously decreasing), the model trained with the training set will not exactly match the data pattern of the validation set. Therefore, the error of this model will be much larger than that of other models. A possible approach is to find the factors that cause the model to change (i.e., which features are responsible for the inflection point where the number of cases goes from zero to a large increase). This approach is theoretically very feasible because inflection points do not just

appear for no reason. However, to find the decisive features, huge amounts of data are needed for observation. Another approach is to use more complex neural networks. If this neural network can learn the model in different time periods at the same time, perhaps the error will be much reduced.

REFERENCES

1. "COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University", Github, https://github.com/CSSEGISandData/COVID-19/blob/master/README.md

2. "Oxford Covid-19 Government Response Tracker (OxCGRT)", Github, https://github.com/OxCGRT/covid-policy-tracker

3. "Covid-19 Forecasting using an RNN", Kaggle, https://www. kaggle.com /frlemarchand/covid-19-forecasting-with-an-rnn