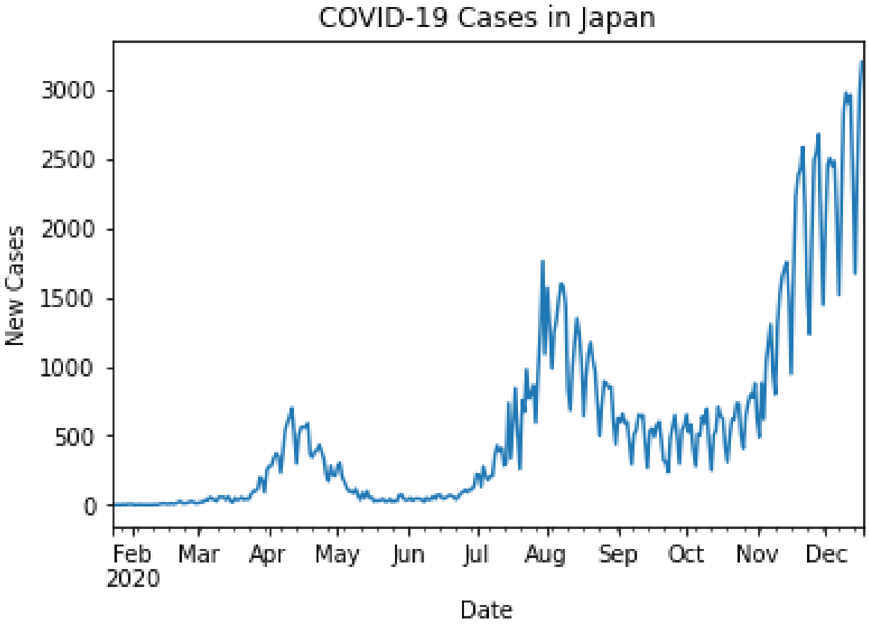
Emilio Padrón Molina

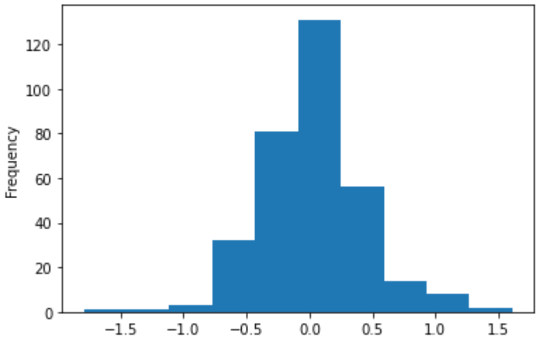
COVID-19 in Japan: Time Series Modeling

Objectives and Description of the Dataset

For this project, I decided to model the rise of COVID-19 new cases in Japan by using data from Johns Hopkins University and observe its predicting power. This data was extracted from a generalized dataset that included COVID-19 novel cases per day, from many countries. These observations start on January 23rd, 2020 and end on December 17th, 2020, making a total of 330 days of data. In the case of Japan, the rise of COVID-19 through time is:

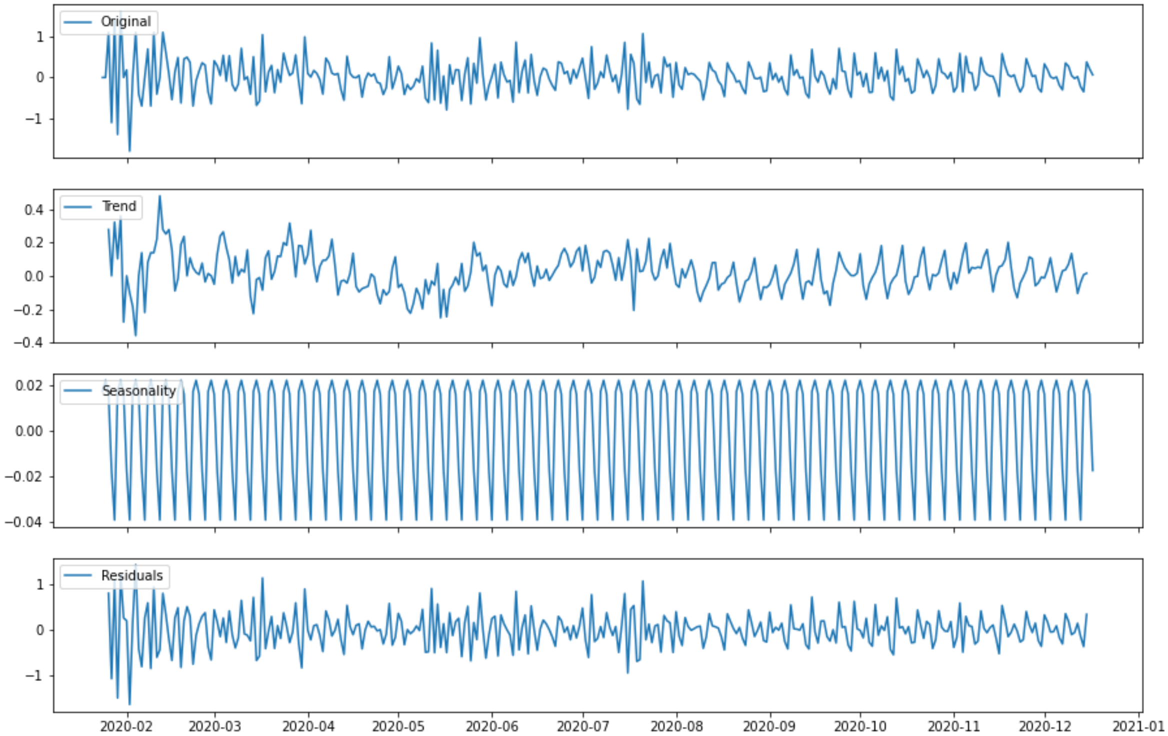


As it can be seen from the previous graph, the series doesn't appear to be stationary; however, this can be proved through several tests. For starters, one can check for trend and heteroscedasticity by dividing the entire dataset into chunks (in this case 10) and comparing the mean and variance of each. The results of this test show that neither the mean nor variance are constant and therefore one must apply a "log1p transformation" to fix this problem. Moreover, given the lagged nature of this data, one can assume that autocorrelation is present in this set; as a consequence, one can transform it by taking the difference between data points. After these two transformations have been performed, it is relevant to check the distribution of the set to see if it is now stationary:



This figure shows a somewhat normal distribution and therefore stationarity; nonetheless, to be sure, one can perform the ADF Test. After applying this test, a *p-value* of 0.0624 is obtained; this means that with an α=0.1, one can reject the null hypothesis and assume the data as stationary.

Furthermore, one can decompose this set as an attempt to understand its trend and seasonality; nevertheless, as one can see from the graphs below, it is not easy to extract the mentioned patterns from this transformed set (especially the trend). Such issue is dealt with later on when selecting models.

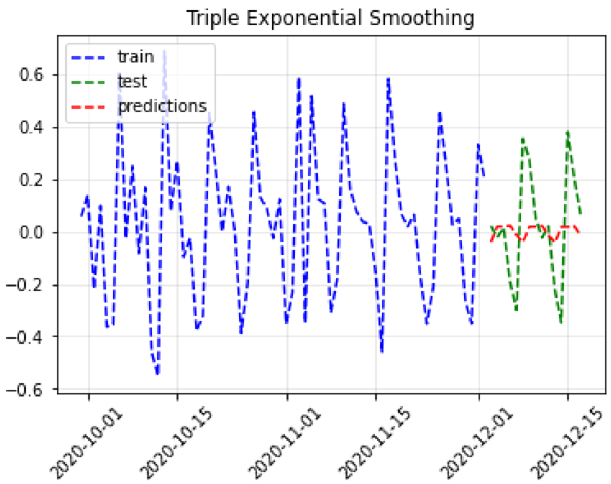


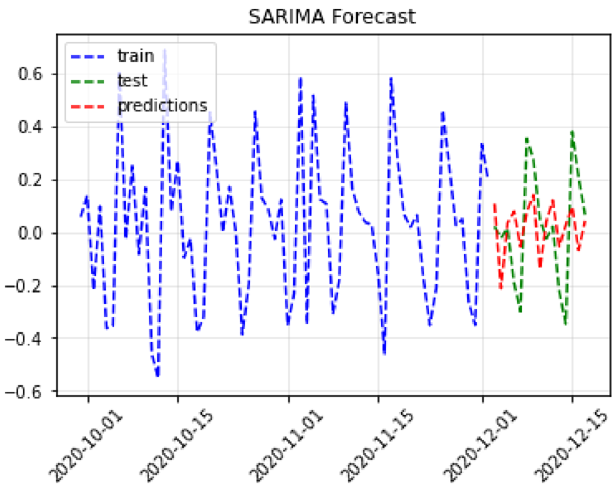
As a final step before modeling, it is necessary to divide the set into its training and testing splits. For the nature of the problem, the last 15 observations are assigned to the testing split while rest is assigned to the training split.

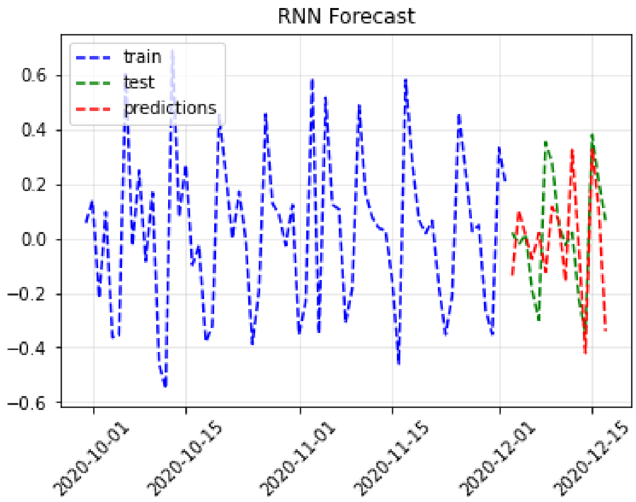
Models and Errors

In this section, three different models where implemented with the training data to then determine its performance with its testing data. Notably, these models were selected based in the fact that there is not a clear tendency nor seasonality. The first and simplest model is a *Triple Exponential Smoothing* with an additive trend and seasonality, and a seasonality period of 5 days. The second model is a *SARIMA* without a bias, and order of (2,0,4), and a seasonal order of (1,1,1,6). It is worth mentioning that the order hyperparameters where selected based on the data's autocorrelation and partial-autocorrelation plots. The third and final model is an *RNN* with a MSE loss, an Adam optimizer, and a batch size of 64, that was trained for 10 epochs.

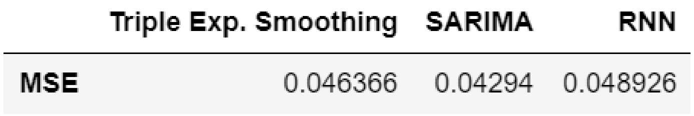
After running these models, it is worth to pass the testing set within the them to see its predictive power. The results are the following:







Furthermore, one can actually measure the performance of these models with the Mean Squared Error. The obtained MSEs are presented in the following table:

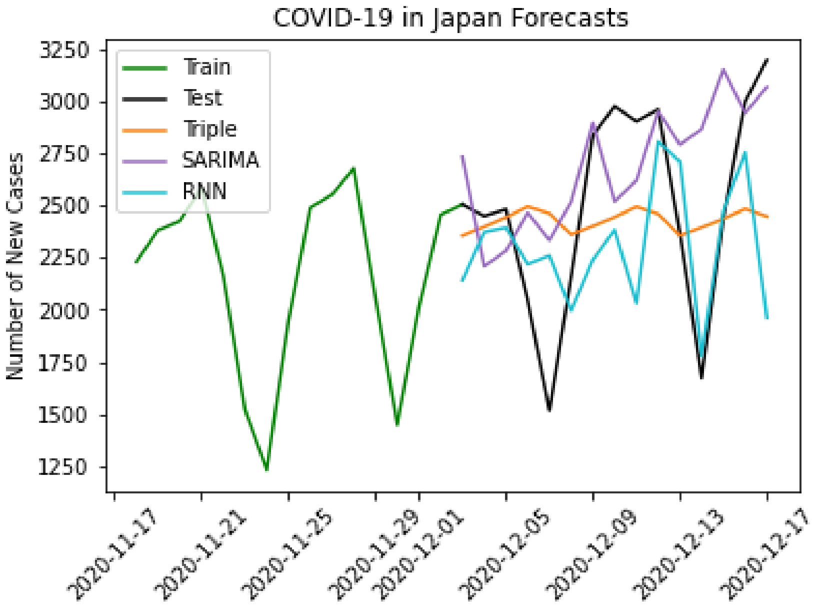


	Triple Exp. Smoothing	SARIMA	RNN
MSE	0.046366	0.04294	0.048926

Insights and Analysis

As it can be seen from the previous results, the Triple Exponential Smoothing oversimplifies the model while the two others actually have some prediction power. Also, given the MSE scores, it appears that the second model predicts better than the third one. However, it is worth mentioning that all these values are not significantly different from one another so one may argue that more evidence is needed to actually point to the best model.

Additionally, using the predicted values from the models, one can apply the inverse transformation from the one mentioned in the first part of this project, to these values in order to observe the true forecast. This results in the following plot:



This plot shows once again that while the first model oversimplifies the true behavior of the virus spread, the second and third model succeed in more or less predicting the future behavior of COVID-19 in Japan.

Finally, it is worth mentioning that there are infinite ways to improve these models. Perhaps one could try changing the order and seasonal order parameters of the SARIMA model, or add more layers to the RNN model. Also, one could try implement an LSTM model to see if there are further improvements. All these is worth doing because it can help the Japanese government take further actions to try to modify the current tendency of spread and fight against the pandemic.