Predicting COVID-19 Cases in Connecticut: A Neural Network Approach

Yinfei Wu

BIS557 Final Project

Abstract

Since the start of 2020, COVID-19, the novel coronavirus quietly began to spread from person to person, taking away millions of people's lives and costing millions of dollars loss. Even though scientists have united to work on the prevention and control of the virus, the virus haven't stopped its step. To help understand the current state of the pandemic and help public health professions make policies, in this article, a neural network architecture which utilized the COVID-19 time series data provided by Johns Hopkins University was built. Especially, we focused on the counties in Connecticut, since they once did a good job in containing the virus. Overall speaking, predictions were made specifically to each county and the model reached a high accuracy rate ~95% with a good fit to the changes of the time trend.

Introduction

Since the outbreak of the novel coronavirus COVID-19 or SARS-CoV-2, the whole world has changed the way it works: millions of people are suffering from pandemic fatigue, trading halts happened several times within a month, major cities were forced to shut down, around 12 million Americans may lose jobless benefits(Adamczyk). Up until December 11, there are 15.7M cases in the United States with 293K deaths in total. Every day, there are more deaths than 911 and the number continued to exploded continuously(Haltiwanger). Yet, the virus still hasn't stopped its way to spread around the world: healthcare system is nearly broken, vaccines development is time-consuming and medical resources is exhausted.

Despite that many research projects were working on the biological treatment of coronavirus, it's equally urgent to model the transmission on a macro level to prevent new cases and focus on cities with high infection rates. An accurate and comprehensive predictive model synthesize all available information and outputs reliable outcomes. Neural network is such a tool that has the ability to learn by itself and work with incomplete knowledge to produce something that is not limited to the input provided to them. Additionally, unlike traditional statistical modeling, it copes with nonlinear relationships and exploit the availability of multiple training algorithms. Given that we haven't had that much information on the table for SARS-CoV-2 yet, neural network is the safe choice for predictive modeling of new cases.

To better study the outbreak mechanism and prevent the happening of future public health crisis, knowing the underlying dynamics of the pandemic is important. Connecticut, a middle state between the hardest hit area of New York and Massachusetts, didn't let the virus go riot, rather, the virus was contained for a while in the state. Thus, in this article, we will build a neural network to explore the COVID-19 cases in Connecticut.

Methodology

1.1 Descriptive analysis of the dataset

Taking a closer look at the input data for the neural network, comprehensive trend plots for all of the counties in Connecticut were made to visualize the changing dynamics across time. Meanwhile, a heatmap was made to exhibit the density of the current state and display the concentration of the virus in Connecticut.

1.2 Data standardization

To reach a reasonable and stable learning process, normalization to rescale the input and output variables prior to training a neural network is necessary. In this case, we standardized the data by subtracting the mean and dividing by the standard deviation (Z-score normalization) for the number of cases.

1.3 Time-step determination

Since our data is time-series data, meaning we have repeated observations for each county in Connecticut every day since the outbreak of the virus, it's crucial to determine how many previous days we need to predict the new cases of the next day. An appropriately picked time-step will improve the accuracy and efficiency of the neural network prediction. If redundant number of days got picked, then too much information was inputted in the algorithm, resulting in a slow and unstable learning rate; if few information was used, it would hurt the precision of the predicted value, which is the same as strongly biased outcomes.

To find the best fitted time-step and the balance between accuracy and efficiency, a sensitivity analysis was conducted. A few candidates (2 days, 5 days, 8days, 11 days, 14 days) were applied on the training data for in-sample forecasting and the accuracy were recorded. The number of day lags with the highest accuracy rate was chosen.

1.4 Train/Test data preparation

Regular train/test data split with cross validation will not work in our dataset, due to the inherent multi-levels of our data. That means, if we randomly throw our data points into the test and train sets, the algorithm is told to model the data on a level regardless of the county information. Therefore, a generalized network across all the counties in Connecticut will be made and the algorithm won't have the ability to adapt to different regions. In order to cope with this potential obstacles, I trained our data with the best fitted time-step all within the same location and tested on the newest data. It allows our algorithm to incorporate county-specific information and output more precise results.

1.5 Prediction model

In this article, I choose to build a neural network that is classical and have all the layers connected with each other. Building the architect of the neural network required the choice of the number of hidden layers as well as the activation function. For the purpose of optimizing the performance of the algorithm, hyperbolic tangent activation function was selected. It allows that the negative inputs will be mapped strongly negative and the zero inputs will be mapped near zero in the tanh graph. Then, for the number of layers, it comes down to trial and error. A few reasonable numbers of hidden layers were used to test on the performance of the network and the one with highest in-sample forecast accuracy was picked. To help better understand the neural network architecture, **fig 1** visualized the inputs and output.

- Hyperbolic tangent(tanh) activation function

$$T(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$T'(x) = \frac{1}{\cosh^2 x} = 1 - \frac{(e^x - e^{-x})^2}{(e^x + e^{-x})^2}$$

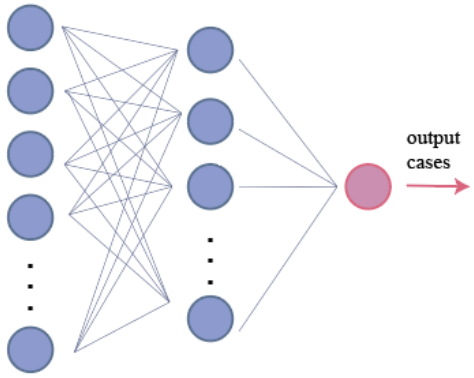

Fig 1: Neural network architect

1.6 Quantifying the behavior of the model

Assessing the legitimate of the model is of great importance. To have a better sense of how the model is behaving, out-of-sample forecasting and comparison strategy is adopted. Additional data was gathered from the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University(Dong, et al.). And the accuracy plots comparing the up-to-date COVID-19 cases in Connecticut against the out-of-sample predicted values were made to capture the deviations of the neural network.

Results

Fig 2 gives us an explicit description of the daily COVID-19 cases bar plots separated by the geographical locations. From the plots, we can see most of the counties shared a similar trend as time goes by. They first increased with a decreasing rate from April to around the beginning of October and then increased with an increasing rate when fall came. Some exceptions are Windham, New London and Tolland which always increased with an increasing rate. Referring to Table 1 and Table 2, where I tested on the sensitivity of the choice of day lags and the hidden layers on the error rate of the neural network and the output showed that a 5 day-lags and 6 hidden layers is the best amongst all of the candidates of mine.

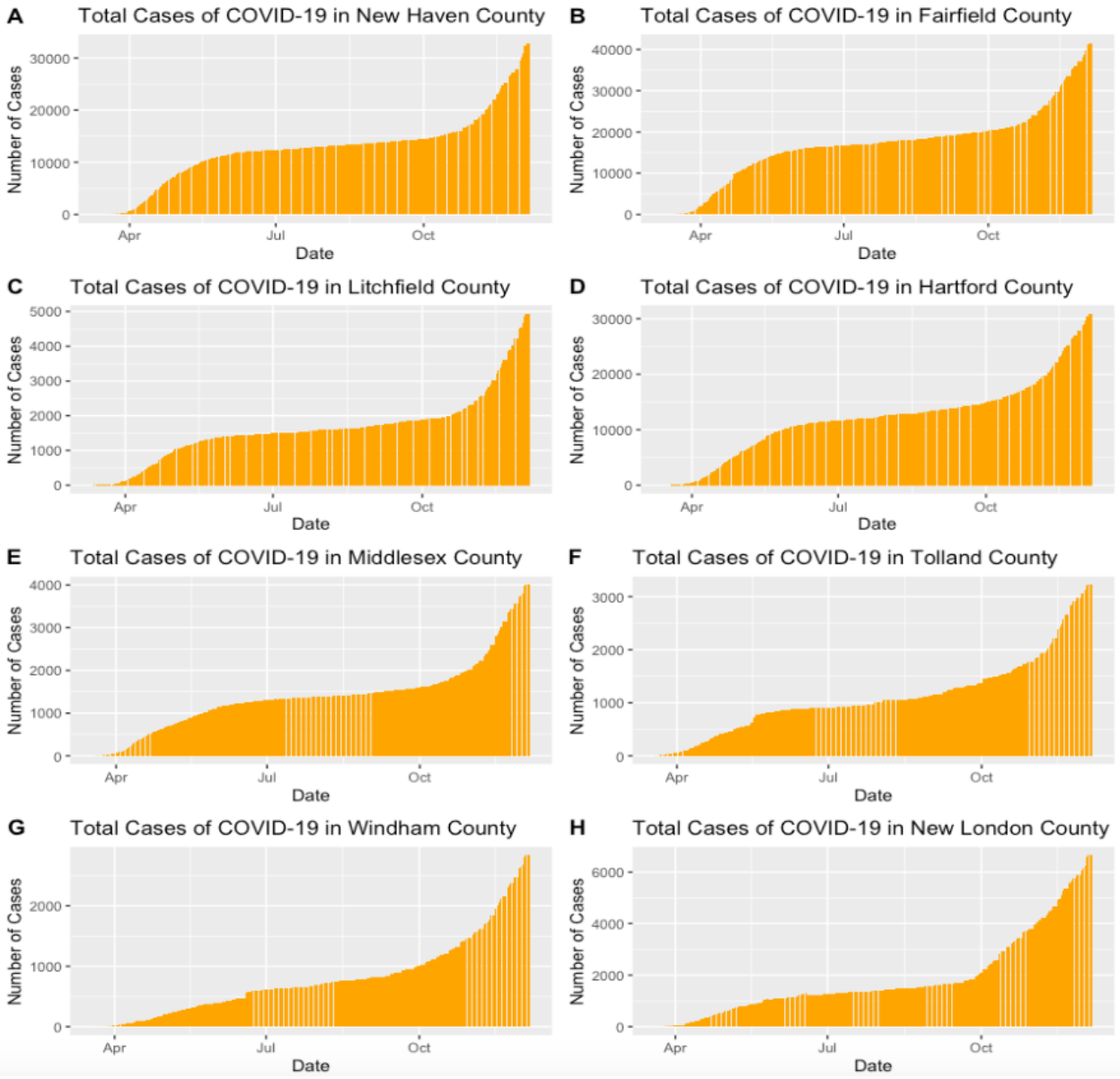

Fig 2: Total number of COVID-19 cases in Connecticut counties across time

Time-step	Error
2 days	0.0917
5 days	0.0766
8 days	0.0772
11 days	0.0784
14 days	0.0875

Table 1: Time-accuracy table

Hidden layers	Error
4	0.11
5	0.097
6	0.084
7	0.089
8	0.090

Table 2: Layers-accuracy table

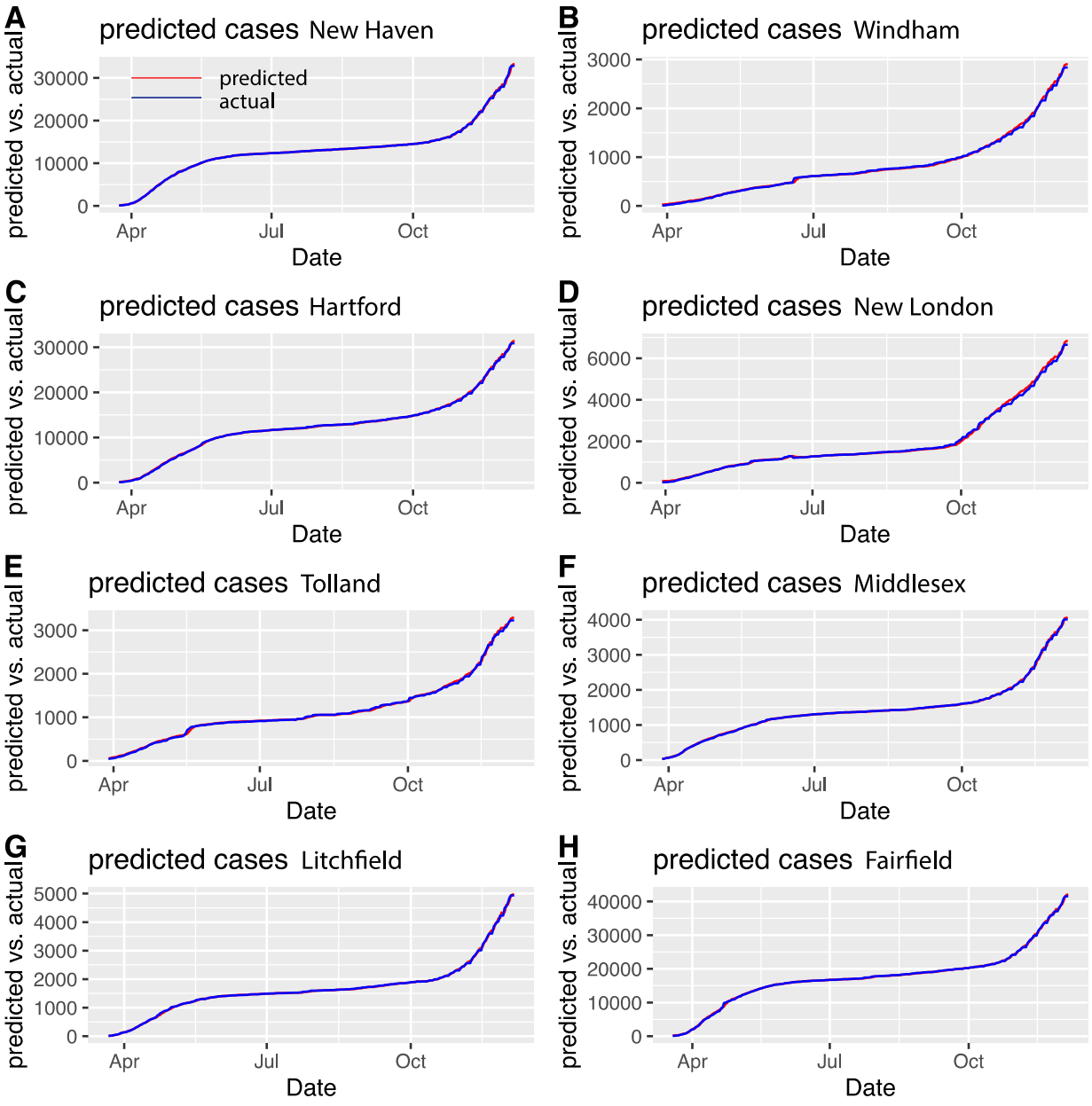

Fig 3: Predicted vs. actual COVID-19 cases in Connecticut counties across time

The predicted number of cases adjusted by counties were all displayed in **Fig 3**. From the plot, we can see the comparison between predicted versus the actual value. On a general perspective, it's shown that our neural network did an excellent job in catching the time trend on a macro level. The predicted lines almost perfectly fit the dynamics of the curvature, which means increase with a slower rate and then with a faster rate. Additionally, the results are very stable. As we can find on the plot, there isn't any sharp jumps or surges as time goes by, rather

all of the counties shared a similar S-shaped predicted trend with a steadily changed speed. Thus, in terms of the overall trend capturing and time series changing dynamics, our neural network is outstanding.

However, if we take a closer look of the data on a micro level, some potential problems have surfaced. Due to the inherent scale of the y-axis, the deviations at each time point is hard to view clearly. A small shift in the plot can represent a huge deviation in the prediction, even though it's hard to tell from the mini-plots. With regard to geography, some special cases are like New London and Windham where there are some noticeable deviations near the end of the time trend, meaning our network did have some flaws in catching the outbreak in those locations. As for the time period, our neural network didn't fit very well at the beginning of the pandemic. Part of it was due to the small number of cases at the beginning. Even if the deviation is small, when the case number is low, the relative percent of error will be greater.

As a whole, the algorithm performed well in the in-sample forecasting. To examine the out-of-sample performance, I collected the up-to-date(Dec 16, 2020) COVID-19 cases (10 more days than our data) in Connecticut for comparison. No surprise that the neural network modeled the spread of virus in a reasonable way with on average only 1.2% percent of absolute bias in the number of cases.

Discussion

Generally speaking, our model is excellent with a small in-sample error rate as well as out-of-sample accuracy. We have shown that the predicted results are stable, consistent across regions and easy to implement. It's also found that due to the insufficient data at the beginning of the pandemic, the predications are biased.

However, more work can be done in the future to improve the network. To begin with, only the tanh activation function was used on the data, even if there are some other options we can adopt, for example, the ReLu activation function which will output the input directly if it is positive, otherwise, it will output zero. Additionally, we should also take into account many other factors influencing the shape of the curve of the increase in infections. Some examples are government behaviors in those regions, seasonality changes in Connecticut, the availability of medical resources and equipment and the efficiency in healthcare delivery. These can be major contributing factors and adds on to the performance of the network. Checking the external validity is also important. Training the model on the data in other states and checking the homogeneity may help us to extrapolate the network to a more generalized version.

In sum, we have discussed both the advantages and the disadvantages of the solution on predicting the COVID-19 cases in Connecticut. And the neural network architecture we built fit the data on a general picture with some lack of precision at the beginning of the outbreak of coronavirus. Further research can be done to play with the algorithm and expand the accuracy with more predictors as well as the fit when there is insufficient data points.

References

Alazab, Moutaz, et al. "COVID-19 prediction and detection using deep learning." *International Journal of Computer Information Systems and Industrial Management Applications* 12 (2020): 168-181.

Adamczyk, Alicia. “As Congress breaks for Thanksgiving, 12 million Americans may lose jobless benefits on December 26”, *CNBC*, Nov 19, 2020

Dong E, Du H, Gardner L. An interactive web-based dashboard to track COVID-19 in real time. Lancet Inf Dis. 20(5):533-534. doi: 10.1016/S1473-3099(20)30120-1

Dhamodharavadhani, S., R. Rathipriya, and Jyotir Moy Chatterjee. "COVID-19 Mortality Rate Prediction for India Using Statistical Neural Network Models." *Frontiers in Public Health* 8 (2020).

Dandekar, Raj, and George Barbastathis. "Neural Network aided quarantine control model estimation of global Covid-19 spread." *arXiv preprint arXiv:2004.02752* (2020).

Haltiwanger, John. The US is likely to have more daily COVID-19 deaths than 9/11 for the next 60 to 90 days, CDC director warns, *Business Insider*, Dec 10, 2020.

Iwendi, Celestine, et al. "COVID-19 Patient health prediction using boosted random forest algorithm." *Frontiers in public health* 8 (2020): 357.

Pal, Ratnabali, et al. "Neural network based country wise risk prediction of COVID-19." *arXiv preprint arXiv:2004.00959* (2020).

Wieczorek, Michał, Jakub Siłka, and Marcin Woźniak. "Neural network powered COVID-19 spread forecasting model." *Chaos, Solitons & Fractals* 140 (2020): 110203.

Wang, Linda, Zhong Qiu Lin, and Alexander Wong. "Covid-net: A tailored deep convolutional

neural network design for detection of covid-19 cases from chest x-ray images." *Scientific Reports* 10.1 (2020): 1-12.