Machine Learning: Predicting COVID-19 Outbreaks

Luke Beatty, Oliver Reckord-Groten, Prasidha Timsina

Table of Contents

Introduction

COVID-19 has had a huge impact around the world and is not something that will go away anytime soon. As of the 23rd of October, 2020, the US reported having the biggest one-day increase of COVID-19 cases at 83,000 new cases. As we approach the cold winter months, it is suspected that we will continue to see an increase in new cases as more populated areas spiral out of control and cases start spilling into areas that, so far, have done an adequate job of handling this pandemic.

This project is designed to collect data on known cases and deaths by county for individual states in order to predict potential outbreaks. These predictions are important because an outbreak could mean disastrous consequences for the county and lead to a lot of unnecessary deaths. To start off, we scoured the internet for data sources that are consistently updated and that have the necessary data features for prediction. We found a plethora of datasets on the official Center for Disease Control (CDC) dataset website, but found that their data was only reporting on deaths. USAFacts, a non-profit organization which offers a non-partisan portrait of the United States' population, currently maintains three datasets that contain applicable information on number of cases, number of deaths, and population. This project will involve preprocessing the collected into an adequate time series, training a Decision Tree Regressor to fit our data, optimizing our hyperparameters, and using the model to predict the spread of COVID-19 cases and subsequently deaths over time.

Problem Definition and Algorithm

2.1 Task Definition

Our goal for this project is to use data on confirmed cases, deaths, and populations of counties around the United States in order to train a model that predicts potential COVID-19 outbreaks. We define an *outbreak* as the point in time when cases start to rapidly rise and spiral out of control, subsequently leading to virus-related deaths. Our inputs, including preprocessed data, would include the following (per county): population, total number of cases, number of new cases, total number of deaths, and number of new deaths. Using these features, our algorithms will predict the number of cases in a given county in seven days. According to the CDC, the average incubation period for a confirmed COVID-19 case is five days, so we chose seven days to give a buffer that accounts for more confirmed cases.

This problem, and potential solution, is incredibly important due to its topical relevance and direct relation to public health. By accurately predicting potential outbreaks before they occur, counties would be given the opportunity to take the necessary preventative measures in order to stunt the spread of the virus. This could in turn lead to the saving of human lives.

2.2 Dataset

The data utilized in this project was collected from USAFacts (usafacts.org). USAFacts provides three public datasets: number of cases per county, number of deaths per country, and county populations. Each of these datasets are updated daily, and start on January 22nd, 2020. They are adequately labelled by county, state, and date. However, in order to properly train any machine learning algorithm, a reasonable amount of preprocessing must occur. To start, the datasets record the cumulative number of cases and deaths over time. While this is useful information, and will be utilized to train algorithms, another useful feature is the number of new cases, non-cumulative, and number of new deaths. Additionally, by removing the most recent data up to the past seven days from the training data, we can add an additional feature that describes the number of cases for any given county after seven days. This feature is crucial to being able to accurately train any model.

2.3 Algorithm Definition

In order to predict the number of cases in the next seven days, a discrete value in the form of an integer value, the algorithms used need to perform regression. Since the data collected in this project is in the form of a time series, we utilize a multivariate time series forecasting model. The initial algorithm of choice is a Decision Tree Regressor. This algorithm breaks down the data into smaller and smaller subsets while developing a hierarchical decision tree containing decision nodes and leaves. In order to measure the quality of any given split, the algorithm utilizes the mean squared error, which is equivalent to variance reduction. There are a number of relevant hyperparameters within the Decision Tree Regressor algorithm, such as the maximum depth and the minimum number of samples required to be at a leaf node. In the hopes of optimizing these hyperparameters to best fit the current problem and dataset, the cross-validation technique of a grid search is utilized.

Going forward, the natural progression from a decision tree algorithm is to generate an ensemble technique such as a random forest. In the random forest method, multiple decision trees are created without pruning which creates probable overfitting. The random forest technique will add additional randomness to the model in order to combat any potential overfitting that occurs within individual decision trees. Additionally, instead of searching for the most important feature to split a node with, the algorithm selects the most important feature from a random subset of features. This coincides with a wider range of diversity within the trees, and generally leads to a more accurate model.

Experimental Evaluation

3.1 Methodology

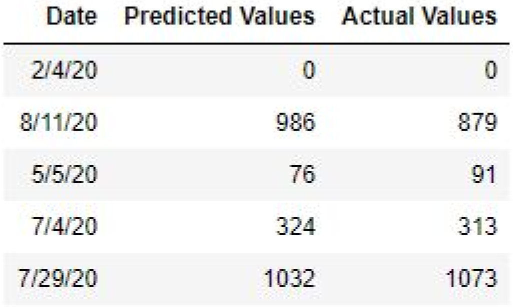
The experiments conducted throughout this project all aim to answer the following hypothesis: when provided with adequate case, death, and population data, a decision tree regressor is able to predict potential outbreaks in a given county. In order to adequately come to a conclusion about supporting or not supporting this claim, evaluation of the model must occur in order to understand how effective it is.

The evaluation metrics utilized for the decision tree regressor are root mean squared error (RMSE) and R-squared, the coefficient of determination. Since the RMSE indicates the absolute fit of the model to the data, lower values indicate a better fit. This metric is an important measure of fit since the main purpose of the model is prediction. R-squared is a goodness-of-fit method that indicates the distance between the data and the fitted regression line as a percentage. A high value of R-squared, 0.7 to 1.0, indicates that the model can account for the flexibility of the response data around its mean. The combination of these two metrics provide us with a deeper understanding of the performance of our model, which can then be used to draw conclusions about our hypothesis.

The training and testing data utilized in this project was split from the original dataset using Scikit-Learn's train_test_split method. This method splits the data randomly into two subsets with user-defined percentages; in this case the testing data made up twenty percent of the original dataset, while the training data made up the remaining eighty percent. Since this data represents the only source of COVID-19 cases and deaths, it is clearly the optimal and most realistic choice when it comes to predicting COVID-19 outbreaks.

3.2 Results

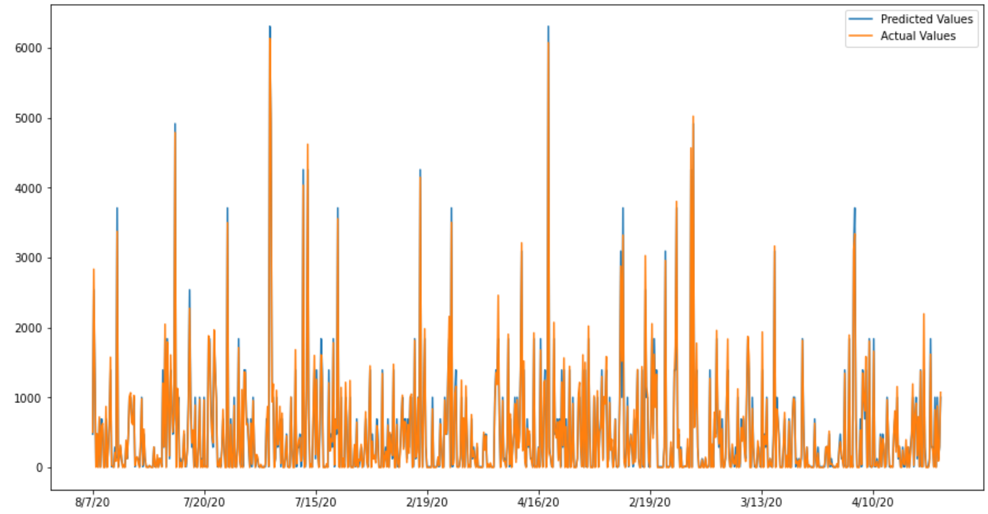
The figure below shows five random dates with their predicted values of cases based on all data up till 7 days before, as well as the number of actual cases on that date. This sample of predictions, although a small sample, indicates a high level of prediction accuracy.



Date	Predicted Values	Actual Values
2/4/20	0	0
8/11/20	986	879
5/5/20	76	91
7/4/20	324	313
7/29/20	1032	1073

After training the decision tree regressor on training data, the previously mentioned evaluation metrics were calculated. The RMSE came out to be 112.36 which indicates that on average, the model's predicted values were 112.36 units (cases) away from the observed data points. Since the dataset has some instances with over a thousand cases, and some instances with zero cases, this is a relatively low RMSE. The R-squared value was recorded as 0.986, or 98.6%. This indicates that the model had a very high prediction accuracy, which is expected when evaluating training data.

Hyperparameter optimization using grid search with cross-validation found the following optimal values: maximum tree depth of 5, maximum leaf nodes of 20, and the minimum number of samples required per leaf node at 0.001. Using these hyperparameters to define a new decision tree regressor, and predicting values using the test data resulted in an RMSE of 98.41 and an R-squared value of 0.985 or 98.5%. These results are encouraging because the R-squared value stayed almost the same, at a high percentage of accuracy, and the RMSE value decreased by 13.95 units. This indicates that the model's predicted values were on average 13.95 cases closer to the observed data points than they were with the training data.



The above graph shows the predicted values that the algorithm produced against the observed data points. As shown on the x-axis, the dates were shuffled because of the randomized selection process when splitting the data into a training and a testing set. While this could certainly be amended in future experiments, it does not overshadow the main function of the graph which is to emphasize the accuracy of the model. The congruence between predicted values and actual values is blatant, and represents a strong accuracy in prediction. Additionally, the balance of blue peaks and orange peaks indicate that the model is not constantly overfitting or underfitting.

3.3 Discussion

The results collected throughout this project confirm our previous hypothesis; a decision tree regressor is able to predict COVID-19 outbreaks in any given county when given an adequate amount of data related to cases, deaths, and populations. As shown in section 3.2, training a decision tree regressor with historical COVID-19 data enabled the model to predict the number of cases in a county relatively accurately. While there may be other regression algorithms that would also fit the given data, we were limited by our knowledge of different algorithms and chose to stick to our strengths in order to better understand how the algorithms were running, and subsequently understand how the data interacts with the model. This particular dataset was challenging to work with because of its relation to a period of time. We had no previous experience working with time series, and it required a lot of research and testing in order to get to a point where we were able to accurately train a model.

Our goal by the end of the semester is to improve upon the accuracy of our predictions by utilizing ensemble learning. For more on this, reference the “Next Steps” section. Additionally, we intend to use this project as a means of furthering our understanding about how these specific models work and how we are able to evaluate model performance using various techniques. One technique that we have discussed using is creating learning curves in order to optimize parameters in a different way and evaluate the tradeoff between bias and variance.

Related Work

Due to the fact that COVID-19 is a recent occurrence, there are not many documented machine learning projects related to it. COVID-19 projections change rapidly because of incoming external factors, and therefore require programmers to spend copious amounts of time keeping up with.

However, one of the most common time-series forecasting problems comes from the field of economics. Predicting stock prices over time is a closely related problem that also benefits from the use of decision tree regressors and ensemble predictors. One particular project on the popular dataset website Kaggle (Oliveira, 2017) shows a plethora of commonalities with this current project. Both projects require a regression technique in order to predict discrete values and have a consistent amount of incoming data. The main difference between the two projects is that COVID-19 growth shows similarities between states and counties while stock prices often vary more aggressively between neighbors due to the sheer number of external factors that affect them.

Next Steps

The next step in this project is to use the predicted cases in one county to attempt to predict how they will affect neighboring counties. This is a natural progression because as people move around their states, COVID-19 cases can follow them and a county that was once doing well, or had only a few cases, could have a new outbreak due to the travel of individuals. We plan on using the random forest ensemble prediction method because our initial model uses a decision tree regressor. The natural next step to make the model more accurate is to combine multiple decision trees into a random forest. Random forests will be especially useful in this scenario because of the variation it adds to the model, which fits hand-in-hand with the varied behavior of humans around the nation.

We also plan on cleaning up our original model and making robust functions so that users can create graphs and heatmaps for their desired state or county. This is important because a critical part of this project is about directly predicting the number of cases within the next seven days and we have a working algorithm but not a functional user experience.

In a similar way that we built this proposal, the next steps will involve each team member wearing multiple hats. We will collectively come up with the concepts that are needed to fit our project and then one team member will execute them, with the other members reviewing their code and debugging. Once the model is up and running, the teammates that reviewed the model code will create visualizations and interpret results.

Code and Dataset

- Datasets: https://usafacts.org/visualizations/coronavirus-covid-19-spread-map/
  - This dataset is shared under a Creative Commons license.
- Code: https://drive.google.com/file/d/1-gEfT4bbG3Aq2M5WAk-jwjiR0g8qqAMb/view?usp=sharing
- How to reproduce - Run the code in a Jupyter notebook. Links for sources of data are included and all files are generated automatically.

Conclusion

The results gathered from this project indicate that the decision tree regressor we trained is able to accurately predict the number of cases within the next seven days in any given county. This is a promising start to the project, however there is always room for improvement. Moving forward we will be enhancing our current algorithm by organizing the way the data is fed to the model which will in turn clarify any visualizations. We also plan on enhancing our current method, decision trees, by utilizing the random forest ensemble method, and extending these projections to predict the influence of cases in a specific county to its neighboring counties. These revisions to the model will allow for us to better see the cases spread throughout counties and states, as well as be able to predict more aspects of our dataset.

Bibliography

Oliveira, D. O. (2017). *Deep Learning for Time Series Forecasting* [Dataset]. https://www.kaggle.com/dimitreoliveira/deep-learning-for-time-series-forecasting/notebook

Singh, D. S. (2020, May 18). *Machine Learning with Time Series Data in Python* [Dataset]. https://www.pluralsight.com/guides/machine-learning-for-time-series-data-in-python

USAFacts. (1/22/20). *US Coronavirus Cases and Deaths* [Track COVID-19 data daily by state and county]. https://usafacts.org/visualizations/coronavirus-covid-19-spread-map/