Covid-19 Forecasting and Non-Pharmaceutical Interventions

Catherine Appleby, John Miller, and Jacob Shusko

May 14, 2020

Abstract

As Covid-19 has spread across the United States, there have been a variety of non-pharmaceutical interventions (NPIs) put into effect by counties to reduce case load and prevent the spread of infection. As states and counties begin to re-open businesses and public spaces, it is important to know which counties will be at risk for large outbreaks and to know an effective timeline for instituting effective NPIs to curb these outbreaks. Using Autoregressive Integrated Moving Average (ARIMA) models, we first forecast cumulative deaths for one, two, and three week periods on the county level. The ARIMA models appeared to perform surprisingly well for counties with higher death counts, but less well with counties with low death counts. Then, we assessed the relationship between community risk factors and a county's pro-activeness with enacting NPIs using tree-based models. The results illustrated that a county's smoker population, stroke mortality rate, and heart disease mortality rate are key factors in predicting how quickly a county enacted NPI measures.

1 Introduction

This paper outlines and evaluates models that can be used to evaluate outbreak severity for U.S. counties, by predicting deaths as well as community responses to Covid-19. This is of immediate relevance to current events and has the potential to influence decision making that could save lives. As cases of Covid-19 increased, government leaders were at odds on how to respond to the virus. On March 16th, 2020, the White House issued guidelines on how to prevent its spread. By this time, some officials had already moved to shut down their jurisdictions, while others thought that the virus would just fade away. This difference in opinion among leadership lead to confusion about what the appropriate response to the virus should be. Additionally, after instituting a non-pharmaceutical intervention (NPI), there is a long lag in the reduction of cases, making it hard to immediately see the effect it may have on outbreak severity.

Originally, our goal was to predict Covid-19 outbreaks by looking at search data from Google Trends. However, Google Trends did not provide the granularity for daily samples of county-level data. Instead, we chose to forecast deaths from Covid-19 as well as predict community responses to the pandemic using county demographics and data on NPIs. The intersection of predicted deaths and responses shows how proactive a county is to an outbreak and how they should react in the future to prevent infection and death.

2 Data Collection

The master data set we collected was joined together from an open source NPI data set provided by Keystone Strategy [1], a county level demographic data set provided by the Census Bureau [2] and the Covid-19 data repository provided by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University [3]. We will call this data

set **master one**[1] for the rest of the paper.

The Covid-19 data set has observations for each county (identified by FIPS) and a cumulative death value for dates starting from 1/22/20 up to the most current update, so we transformed this observation into a time series for each county. Then, we joined this time series data with the Census data set adding demographic information for each observation of the time series. The NPI data set gave start dates for each county for many policies such as closing of public venues, lockdown, and social-distancing; so we calculated the days in effect for every policy as features for each time series entry. All this preprocessing is contained in a python script found in our GitHub repository[2], which we ran on 5/9/2020 outputting a master data set with 35,748 rows and 179 columns.

In addition to this previous master data set, we found another data set that includes county-level demographic and health metrics from the Yu Group at the University of California, Berkeley [4]. We merged this data set with the Covid-19 data repository mentioned previously adding the death rate over the time span as a feature for each county. This data set has only one observation for each county code, which allows us to do analysis without a time series format. We will call this data set **master two**[3] for the rest of the paper.

3 Methods

We used ARIMA models to forecast cumulative deaths over one, two and three week periods for each county. This segment created 331 ARIMA models all fitted automatically using the R function auto.arima(). We used tree-based methods in Python to predict if a county was fast to act in terms of enacting NPIs.

[1] master one is named "master_5-8-20.csv" in the repository
[2] Our repository for data and scripts: https://github.com/jshusko/covid19_forecast
[3] master two is named "master_yu.csv" in the repository

3.1 Forecasting with ARIMA Models

In developing the ARIMA models, we referred heavily to the *Forecasting: Principles and Practice* [6], specifically chapter eight on ARIMA modeling. We chose to use ARIMA models considering they are the most general class of models for forecasting a time series that is stationary, and fortunately the R function auto.arima() computes d differences of the data to make it stationary. Thus, we estimate the cumulative deaths of a certain county on day t as y_t satisfying the general equation based on parameters p, d and q:

$$\hat{y}_t = \mu + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-1} - \theta_1 e_{t-1} - \cdots - \theta_q e_{t-q} \tag{1}$$

where μ represents the long-term drift, θ_i represents the moving average parameters indexed by p, ϕ_j represents the effect related to previous lags indexed by q.

To work with a smaller sample at first, we created a subset of nine key counties distributed around the United States: Los Angeles County, CA, New York County, NY, Cook County, IL, Gwinnett County, GA, Harris County, TX, King County, WA, Suffolk County, MA, Hennepin County, MN, and Hillsborough County, FL. We then check the data before and check the model after fitting to an ARIMA to ensure statistical properties such as stationary and constant variance hold (more on this later). After running this procedure and evaluating the best forecast interval, we ran this procedure on all the counties in **master one**. See section 4.1 for results.

3.2 Predicting County NPI proactivity with Tree-based Methods

To transform the data in **master two** to fit a classification problem, we developed a scoring rubric such that we could label counties that proactively enacted NPIs. A county was labeled proactive if it enacted an NPI before the White House issued its guidelines for social distancing on March 16. In our data set of 2,825 counties, 825 counties were

classified as proactive and 2,000 counties were classified as non-proactive. All numerical demographic features ranging from median age to number of ICU units for each county were selected; the proactivity scores were used as labels. We made a 70%/30% split for the training and testing data, respectively.

We performed a randomized cross validation search algorithm on different hyperparameter values to fit the classifier to the training data; we used a 3-fold cross validation splitting strategy. After running the algorithm multiple times we identified a small range of the best hyperparameter values to use, and performed an exhaustive grid search 3-fold cross validation algorithm on this range to find the best combination of features. See Appendix D for more on the hyperparameters.

We created a Random Forest classifier with the optimal hyperparameters, and once more trained and tested. After calculating the classification accuracy score on the test set and viewing the confusion matrix for the model, we looked at the node purity values to asses which predictors were the most important. To confirm that we had identified the most important demographic estimators, we trained a simple Random Forest classifier (without specifying hyperparameters) on this subset of features and once again predicted county scores. See section 4.2 for results.

4 Results

Using the previously mentioned methods, we forecast the cumulative deaths of counties for short periods in section 4.1 and we use county demographic factors to predict if a county was proactive in enacting NPI in section 4.2.

4.1 ARIMA Forecasting Results

As stated in section 3.1, we first ran our time series analysis process with a subset of key counties. After plotting the cumulative death data we noticed the time series was non-stationary and there is no evidence of heteroscedasticity so a Box-Cox transformation was not necessary. Thus, we were ready to run auto.arima() on these nine counties. After fitting these models, as seen in Table 1, most models had an acceptable Aikaike Information Criterion (AIC) value including many differences and higher order smoothing parameters to fit the data.

	LA	**NY**	**Cook**	**Gwin**	**Harr**	**King**	**Suff**	**Henn**	**Hill**
(p,q,d)	(2,2,0)	(0,2,5)	(2,2,3)	(0,2,3)	(3,2,2)	(4,2,1)	(0,2,2)	(2,2,2)	(0,2,4)
AIC	550.34	777.26	578.48	235.45	243.29	482.24	534.27	231.14	159.61

Table 1: ARIMA parameter values and AIC for subset fits.

Now, before forecasting using these ARIMA models, we ensured that the residuals from the model have no correlations using ACF plots which can be found in Appendix C. The residuals seem to behave like white noise after viewing the ACF plot. Now, we can finally forecast over the remaining time series data. The forecast for the two week period can be seen in the figure in Appendix B. We followed this same procedure with all of the counties as well predicting over one, two and three week intervals. Then, we summarized MSE of the models as seen in Table 2, and it seems the two week forecast performs the best while the one week forecast performs poorly likely due to a high sensitivity to local trends.

	7 Days	**14 Days**	**21 Days**
Subset Max	598,553.7	29,721.93	33,945.17
Subset Mean	67,781.48	5,134.886	4,975.358
All Max	598,553.7	53,299.65	98,533.29
All Mean	78,043.51	33,251.59	28,008.11

Table 2: Max and average MSE for both the subset and all of the counties over forecast periods 7, 14 and 21 days.

4.2 Random Forest with NPIs Results

The Random Forest classifier with hyperparameters chosen by cross-validation had an accuracy score of 77.5% on the test data and 100% on the training data. We calculated the true positive rate (TPR) and true negative rate (TNR) from the confusion matrix in Table 3 and got values of 37.6% and 93.7%, respectively.

Predicted/Actual	Non-proactive	Proactive
Non-proactive	565	38
Proactive	153	92

Table 3: Confusion matrix for the cross-validated Random Forest model.

We found the most important predictors for decision tree branching by looking at the model's node purity values as seen in Table 4. Our results offer a glimpse at how county officials assess their county's risk-level, mostly focusing on health condition data excluding factors such as gender proportion and partisan split.

Feature	**Purity**	**Feature**	**Purity**
Smokers Percentage	0.1264	Population Estimate	0.0559
Stroke Mortality	0.0731	Diabetes Percentage	0.0518
Heart Disease Mortality	0.0711	Population Aged 65+	0.0485
Population Density	0.0684	3 Year Mortality Aged 85+	0.0415
Respiratory Mortality	0.0681	Number of Doctors	0.0394

Table 4: Node purity values for the ten most prominent features in the model.

The simple Random Forest model trained on the top five most prominent predictors yielded an accuracy score of 74.8% on the test data and 100% on the train data, with a TPR and FNR of 32.7% and 91.9%, respectively. It seems that Random Forest models trained on this data have a tendency to overfit, and produce a low TPR, likely since demographic data is not the exclusive decision factor in choosing to enact an NPI; travel rates, neighboring county's actions, and the complete political environment are not considered. The fitting was also likely biased due to under sampling of the proactive class in the data.

5 Conclusion

In this paper, we focused on two aspects of the COVID-19 pandemic: deaths and non-pharmaceutical interventions. We applied an ARIMA model to forecast deaths for individual US counties using reported death data as a stationary timeseries model. We saw that this method was fairly accurate at predicting deaths from a two and three week period, but forecasting for the one week period resulted in significantly less accurate predictions. We applied tree-based methods to NPI, demographic data and some political data to see which kinds of counties were responsive to Covid-19. Despite seeing a tendency to overfit in Random Forest, we saw that counties with higher populations of people with known health risk factors were more proactive. These were notably counties with high smoker percentage and high incidences of stroke mortality, heart disease, respiratory problems, and diabetes. One interpretation of this result is that policy makers used community risk factors to help make decisions. Another interpretation could be that counties with higher population densities had higher smoker populations and earlier outbreaks, resulting in earlier responses.

The models we used could potentially be applied to countries outside of the US, giving a global view of responses to the pandemic. The same methodology can also be applied to future pandemics or epidemics as they emerge, identifying areas with better responses to the disease. To expand on this project, it would be beneficial to compare case or death numbers from counties with similar population risk factors but different timelines in instituting interventions. This will provide more information on effective responses and where an outbreak is worse. Additionally, oversampling techniques could be used to improve the true positive rate and overall test rate of the tree-based methods. Further researchers should try to incorporate these demographic and NPI results to assist with forecasting the cumulative deaths for a county.

References

[1] Keystone Strategy. (2020). *Keystone-Strategy/covid19-intervention-data*. Retrieved from Covid19 Intervention Data: https://github.com/CSSEGISandData/covid19-intervention-data

[2] U.S. Census Bureau. (2019). *Counties 2019 Totals Alldata.* Retrieved from United States Census Bureau: https://www2.census.gov/programs-surveys/popest/datasets/2010-2019/counties/totals/

[3] Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. (2019) *CSSEGISandData/COVID-19*. Retrieved from COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University: https://github.com/CSSEGISandData/COVID-19

[4] Altieri, Nick; Barter, Rebecca; Duncan, James; Dwivedi, Raaz; Kumbier, Karl; Li, Xiao; Netzorg, Robert; Park, Briton; Singh, Chandan; Tan, Yan Shuo; Tang, Tifany; Wang, Yu; Yu, Bin. (2020). *Curating a COVID-19 data repository and forecasting county-level death counts in the United States* Berkeley: University of California, Berkeley

[5] Nau, R. (2019). *Introduction to ARIMA: non-seasonal models*. Retrieved from Statistical Forecasting: notes on regression and time series analysis: https://people.duke.edu/ rnau/411arim.htm

[6] Hyndman, R.J., Athanasopoulos, G. (2018) *Forecasting: principles and practice, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2.* Accessed on May 15, 2020

Appendix

A One Week Forecasts

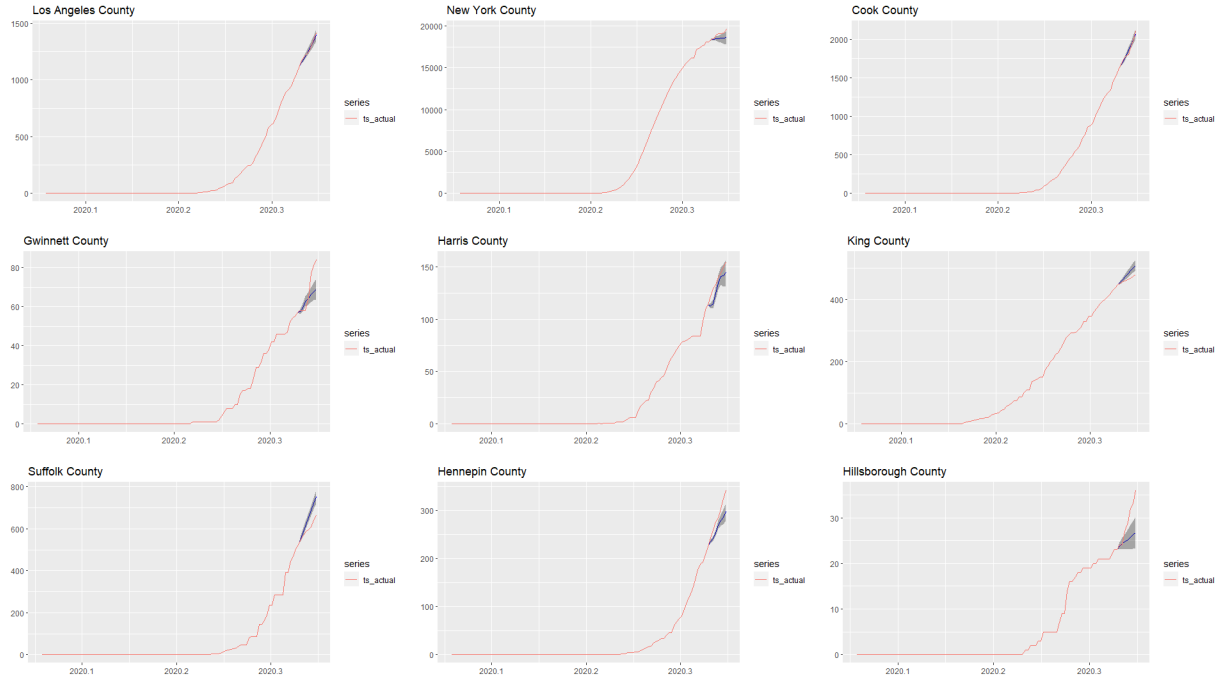

Figure 1: One week forecast for the subset counties. The red line is the actual data, blue line is the point estimate, and blue shading is the 95% and 80% confidence intervals.

B Two Week Forecasts

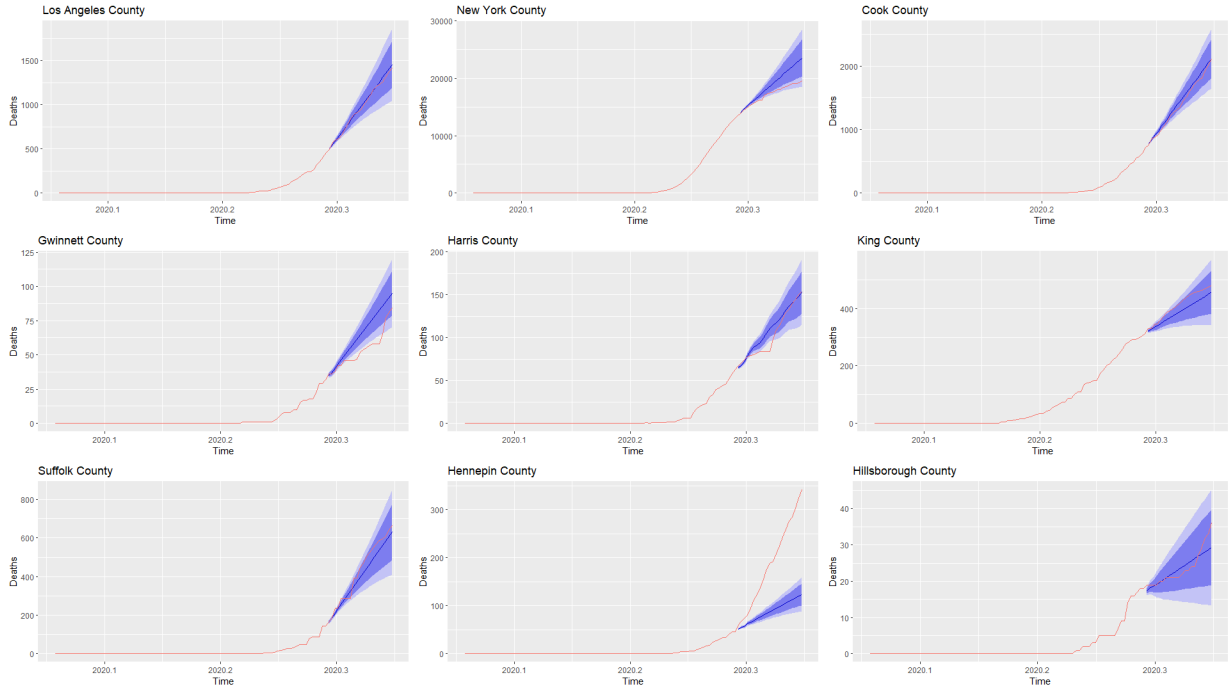

Figure 2: Two week forecast for the subset counties. The red line is the actual data, blue line is the point estimate, and blue shading is the 95% and 80% confidence intervals.

C ACF Plots

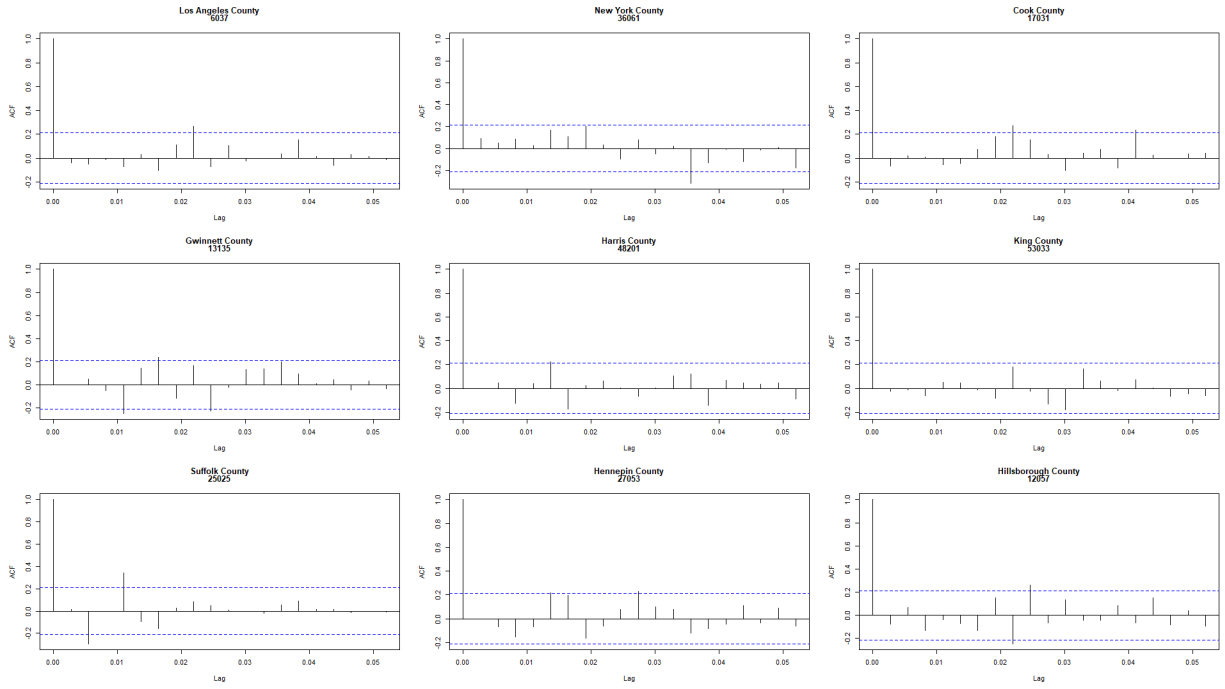

Figure 3: Autocorrelation plots for the subset of counties.

D Optimal Hyperparameters for Random Forest

The specific parameters we investigated were as follows: number of trees used, maximum features considered for node-splitting, maximum depth of decision trees, minimum data points placed in a node before splitting, minimum data points allowed in leaf nodes, and whether or not to use bootstrapping when sampling. Parameters for RandomForestClassifier from sklearn.ensemble library in Python: (bootstrap: False, max_depth: 17, max_features: auto, min_samples_leaf= 1, min_samples_split: 5,n_estimators: 100)

Note: "max_features = auto" means that square root of estimators was taken