CS230

A Deep Learning Approach to Population Based COVID-19 Case Prediction in the US

Sameer Sundrani*
Department of Biomedical Computation
sundrani@stanford.edu

Amy Zhang*
Department of Computer Science
ayzhang@stanford.edu

All code is publicly available at this GitHub

1 Introduction and Related Work

Over the past two years, COVID-19 has caused over 700,000 deaths with 44 million cases in the United States alone (1). In an effort to curb the spread of the illness during the peak of the first wave, 42 US states and territories issued mandatory stay-at-home orders (2). As these lockdowns and stay-at-home orders became the new normal during the past two years, the US has seen unprecedentedly low rates of travel, with travel spending decreasing by 42% (nearly $500 billion) from 2019 to the end of 2020 (3). Decreasing travel became synonymous with preventing the spread of COVID; however, surveys show that there are major disparities between the percentage of people staying home in each state with some state percentages as high as 80% and others as low as 5% (4). Machine learning techniques have the potential to gain insights from these disparities and their resulting correlations with COVID cases. As such, we aim to investigate the predictive capabilities of Neural Networks and several machine learning baselines when applied to US travel data in order to predict the quantity of COVID cases at a county level.

Machine learning can also aid in the quantification of COVID reporting inaccuracies. A study in Science Translational Medicine suggests that upwards of 80% of COVID cases in the US during March 2020 went undetected (5). These massive reporting inaccuracies are further exacerbated by the vastly different testing capabilities of each state's healthcare systems. As such, we hope to train our neural network on data from states that have robust COVID testing and reporting systems (6). We then intend to use this predictive model to generate a model-specific quantification of reporting inaccuracies in low-testing states using travel data from these states. Given that there are no ground truth labels for these states (accurate COVID case numbers), we will be using a combination of domain knowledge and other proxy economic and social variables to evaluate the performance of our model for this task.

2 Datasets and Features

Our dataset consists of travel data (trips by distance) from the US Department of Transportation Bureau of Transportation Statistics, economic indicators from the U.S. Department of Agriculture Economic Research Service, and COVID case data aggregated by the New York Times (7; 8). Both the travel data and COVID case data are daily time series at the county level while the economic indicators are time invariant and taken from the most recent year. We use as input into our model trips by distance and the economic indicators. We use COVID case counts as our labels. Our dataset spans from January 21, 2020 to October 9th, 2021. We preprocessed the data by dropping any rows

*Both authors contributed equally

CS230: Deep Learning, Fall 2021, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

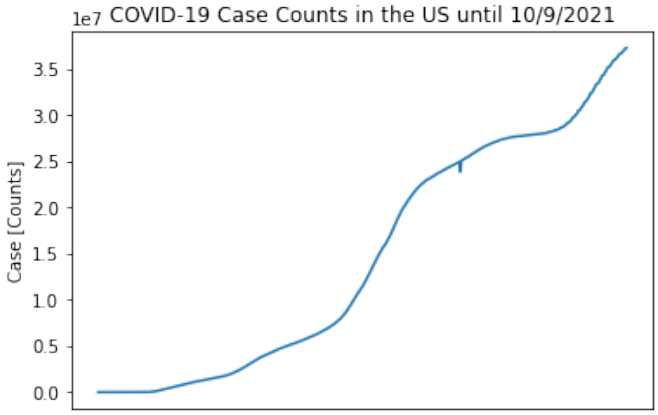


Figure 1: COVID-19 Case Counts until October 2021

All Features		
County	Number of Trips 10-25	Number unemployed annual average
State	Number of Trips 25-50	Unemployment rate
Number of Trips	Number of Trips 50-100	Estimate of median household Income
Number of Trips <1	Number of Trips 100-250	Median household income percent of state total 2019
Number of Trips 1-3	Number of Trips 250-500	Estimate of people of all ages in poverty 2019
Number of Trips 3-5	Number of Trips >=500	Estimate of people age 0-17 in poverty 2019
Number of Trips 5-10	Civilian labor force annual average	
Number of Trips 10-25	Number employed annual average	

Table 1: Model Features

with NaN input feature values and/or labels and normalizing our entire dataset to have zero mean and unit variance. Our final dataset consists of 1,573,394 data points.

2.1 Trips by Distance

The trip by distance data is produced from anonymized mobile data using a weighting procedure that "expands the sample of millions of mobile devices, so the results are representative of the entire population in a nation, state, or county". It is also important to note these following details about the trips by distance data: "Trips are defined as movements that include a stay of longer than 10 minutes at an anonymized location away from home. Home locations are imputed on a weekly basis. A movement with multiple stays of longer than 10 minutes before returning home is counted as multiple trips. Trips capture travel by all modes of transportation. including driving, rail, transit, and air." More details on the travel dataset can be found publicly (7). We broke down trips by distance into the 10 features described in Table 1 that reference the Number of Trips.

2.2 Economic Indicators

In addition to the travel data, our dataset also includes numerous county-level, economic indicators (unemployment statistics and poverty statistics). This data was included given the close relationship between these indicators and travel as well as healthcare resources. The full list of features included in our dataset can be seen in Table 1 (9).

3 Modeling Methods

3.1 Evaluation

To evaluate our models we utilize the Pearson correlation coefficient, r:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{1}$$

as a proxy for the linear correlation between our model's outputted case counts and the true case counts. We also utilize mean absolute error, MAE, which has been used in prior economic works (10):

$$MAE = \frac{\sum_{i=1}^{n} |\hat{y}_i - y_i|}{n} \tag{2}$$

The following baseline machine learning models were tested and trained: L1 regularized regression, L2 regularized regression, and XGBoost decision trees. We then compare the r and MAE of each model on a validation set with the r and MAE of 2 layer, 3 layer, 10 layer, 4 layer and 7 layer neural networks (with varying number of hidden units). We train and test all models using existing machine learning and deep learning packages (11; 12; 13; 14).

L1 Regularized Regression: Our objective for L1 regularized regression is

$$min \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \tag{3}$$

where β is our learned coefficient vector, whose L1-norm is penalized by hyperparameter λ. By default, we set $\lambda = 1$ to assess baseline performance.

L2 Regularized Regression: Our objective for L2 regularized regression is

$$min \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \tag{4}$$

where β is our learned coefficient vector, whose L2-norm is penalized by hyperparameter λ. By default, we set $\lambda = 1$ to assess baseline performance.

XGBoost: XGBoost is a popular model based on decision tree ensembles, or a set of classification trees where leaf values are summed to give a final prediction for a particular respondent (13). The objective function for XGBoost is

$$obj = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k}^{K} \Omega(f_k) \tag{5}$$

where $\hat{y}_i$ is our model output, l is our loss, f is a function over the functional space of possible sets of trees, $\Omega(f_k)$ represents the complexity of a tree, and model training is performed by learning one tree, f_k, at a time. For training, we use the following default parameters: with colsample_bytree = 0.3, learning_rate = 0.1, max_depth = 5, alpha = 10, n_estimators = 10.

Neural Networks: We train each of our neural networks using the Adam optimizer with MAE as our loss function and a step-size of 0.01 and for 100 epochs (Note: the default step-size was 0.001, but we noticed that training for our models was too slow as characterized by a loss curve that did not exhibit flattening at the final epoch.).

3.2 Training and Dataset Splits

We hypothesize that partitioning our training dataset by state would be an appropriate partition given the major disparities in testing capabilities. Using the previously mentioned article (6), we took the data from 'VT', 'ME', 'NY', 'RI', 'MA', 'NH', 'CT', 'HI', 'MI', 'WA', 'MD', 'NJ', 'CA', 'DE', 'VA', 'CO', 'FL', 'IL', 'NC', 'LA', 'NM', 'WV', 'OR', 'SC', 'AK', 'GA', 'OH', 'AR', 'PA', 'MN', 'IN', 'NV', 'NE', 'UT', 'OK', 'KY', 'MS', 'AZ', 'MO', 'TN', and 'TX' as our training dataset (1,273,308 entries) and tested on data from 'SD', 'IA', 'WY', 'ID', 'KS', 'AL', 'MT', 'WI' which were reportedly the states that were testing the least (split into validation and test sets each with 136,613 entries).

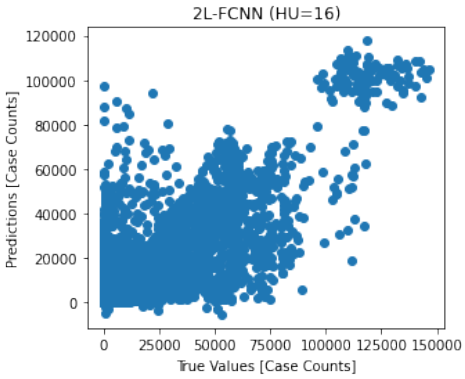
4 Results

4.1 Validation Set

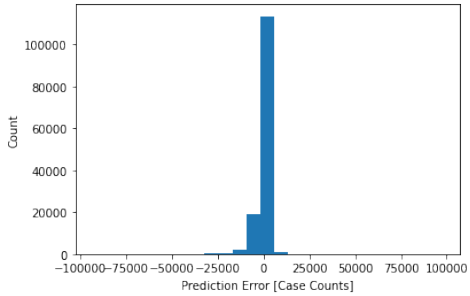
We evaluated each of our baseline and deep learning models first on a validation set to select a top performing model as seen in Figure 2. Although the best performing model on MAE was a 2-Layer

Model	Pearson's Correlation	MAE
Lasso	0.827212234	2238.494855
Ridge	0.824789774	2312.516492
XGBoost	0.832711927	2036.22094
2 Layer (ReLU) (W=64)	0.850874764	1493.148057
3 Layer (ReLU) (W=64)	0.813584249	1653.832987
10 Layer (ReLU) (W=64)	0.018013689	2659.621369
2 Layer (ReLU) (W=16)	**0.8583108982**	**1502.351529**
3 Layer (ReLU) (W=16)	0.84777755	1536.046116
3 Layer (ReLU) (W=9)	0.855034424	1561.403353
7 Layer (ReLU) (Exp-Con)	0.8200906687	1664.354521
4 Layer (ReLU) (Exp-Con)	0.8282978918	1567.432037

(a)



(b)



(c)

Figure 2: (a) Pearson's r and Mean Absolute Error (MAE) on our validation set across all linear, non-linear, and deep learning models, (b) Predicted case counts against true case counts for the model highlighted in red, and (c) Plotted error distribution of that same model.

Data	Pearson's Correlation	MAE
Test Set	0.8559363283	1514.085948
Recent Data	0.9572112649	5800.219339

Table 2: Test Set and Current Data Results

network with 64 hidden units per layer, we chose the 2-Layer network with 16 hidden units per layer to avoid potential high bias on the test set (and considering that the 16 hidden unit model performed slightly better on Pearson's r).

4.2 Test Set and End-to-End Performance

Our chosen model performed similarly on the original test set as it did on the validation set, as seen in Table 2. In addition to testing our model on the test set created from initial data splitting, we pulled the most up-to-date (as of November 29, 2021) data from the NYT and BTS sources (7; 8), which included only case counts from the last 30 days, none of which were included in the initial dataset. Using the same model, we evaluated performance using the same metrics for each prior test and found that we predicted current case counts quite well, with a Pearson's $r \approx 0.96$. These results are also shown in Figure 3.

5 Conclusion

Due to time limitations, we could not explore all avenues surrounding COVID reporting inaccuracies. For future work, we would like to consider the following:

1. Exploring the performance of other models such as LSTMs and transformers given their success in the task of multivariate time series forecasting (15).

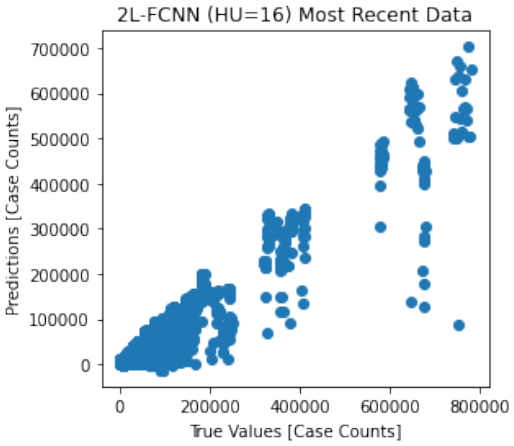


Figure 3: Predicted COVID case counts against true case counts for the month of November 2021

2. More hyperparameter and model tuning surrounding things like batch normalization, dropout, initialization schemes, (interlayer) activation functions, learning rate decay to decrease training time, etc.
3. Developing a better way to measure how well our models can account for misreporting or poor data collection. The fact that we have no ground truth about COVID cases in states with poor testing (and to some degree, all states) poses a major limitation to evaluating our approach.
4. Inputting more time-dependent features into our model such as weather, amount of stores open in a particular county, number of hospital beds available, etc. This may allow for greater non-linear modeling by deep learning methods and could lead to more robust predictions.
5. Examining our results on the county level to determine potential county- or state-specific disparities. While we see that our model error's are visibly normally distributed with $\mu \approx 0$ (see Figure 2, it may be true that we are not capturing specific locations as well as others.

In conclusion, we have shown that transportation data and basic economic factors are useful indicators to predict COVID case counts on the county level across the United States. Deep learning allowed for nearly a 35% reduction in MAE and 3% boost in Pearson's r over regularized regression models (lasso and ridge) as well as nearly a 20% reduction in MAE from boosted decision trees (XGBoost). In addition, more complicated neural network architectures that project the data into higher dimensional spaces seem to not contribute much predictive power, both in terms of Pearson's r and MAE, over simpler two and three fully-connected architectures.

6 Contributions

Both Amy and Sameer contributed equally to this project.

References

[1] Nov 2021, page Version ID: 1053774941. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Template:COVID-19_pandemic_data&oldid=1053774941

[2] A. Moreland, "Timing of state and territorial covid-19 stay-at-home orders and changes in population movement — united states, march 1–may 31, 2020," *MMWR. Morbidity and Mortality Weekly Report*, vol. 69, 2020. [Online]. Available: https://www.cdc.gov/mmwr/volumes/69/wr/mm6935a2.htm

[3] ncarpenter@ustravel.org, "Covid-19 travel industry research," Mar 2020. [Online]. Available: https://www.ustravel.org/toolkit/covid-19-travel-industry-research

[4] N. V. . March 4 and 2021, "Which states are staying home. and is it working?" [Online]. Available: https://quotewizard.com/news/posts/stay-at-home-orders-and-covid-cases

[5] "What we're reading: Missing covid-19 cases." [Online]. Available: https://www.ajmc.com/view/what-were-reading-missing-covid19-cases-sanofi-ramps-up-vaccine-work-hand-sanitizer-warning

[6] K. Collins, "Is your state doing enough coronavirus testing?" *The New York Times*, Jul 2020. [Online]. Available: https://www.nytimes.com/interactive/2020/us/coronavirus-testing.html

[7] [Online]. Available: https://data.bts.gov/Research-and-Statistics/Trips-by-Distance/w96p-f2qv

[8] [Online]. Available: https://github.com/nytimes/covid-19-data

[9] "Usda ers." [Online]. Available: https://www.ers.usda.gov/data-products/county-level-data-sets/download-data

[10] M. D. Hssayeni, A. Chala, R. Dev, L. Xu, J. Shaw, B. Furht, and B. Ghoraani, "The forecast of covid-19 spread risk at the county level," *Journal of Big Data*, vol. 8, no. 1, p. 99, Jul 2021.

[11] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[13] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 785–794. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939785

[14] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[15] J. Grigsby, "Multivariate time series forecasting with transformers," Oct 2021. [Online]. Available: https://towardsdatascience.com/multivariate-time-series-forecasting-with-transformers-384dc6ce989b