

MSBA Capstone Project

**Forecasting Nursing Home Covid-19 Cases**

****

**Team 13**

**Group Members:**

Yash Warty, Yashaswini Kalva, Jackson Hassell, Raymond Li, Emma Sun

Executive Overview

Our task was to find the best way to predict Covid-19 cases at nursing homes. This is obviously an extremely useful prediction to make - if we know a home is at high risk, they can start taking preventative measures like enforcing mask mandates and not allowing visits.

First we had to prepare the data, as nearly half of the values in the CDC-provided dataset were missing, and bring in some more data about each nursing home. However, we were still running into issues with modeling - it turns out most homes had no cases most weeks, and our models were struggling to predict when that first case in weeks would occur. We solved this by aggregating our data into counties - adding all of the cases for each home in the county together, and predicting that. We had much more success with this.

We tried seven models of varying complexity, but found that the best model was **Lasso** regression, which is simply linear regression with an added regularization component that forces it to only use the most important predictors. With this model we were able to predict on average within 10 cases of the true number of cases in the county that week. We were also able to find what features were most important, which was not surprisingly the number of cases among patients and staff in the previous weeks.

## Introduction

Since 2019, the Coronavirus Disease (COVID-19) has claimed the lives of more than six million people worldwide. The COVID-19 pandemic started when the coronavirus disease 2019 was first identified in Wuhan, China. The virus spread worldwide after attempts to contain it failed, causing the start of the pandemic. Over the course of the past two years, significant advancements in the fight against the virus have been developed such as quick and efficient ways of detecting the virus in an individual as well as administering vaccines that have up to a 90% efficacy rate against the virus. Despite this, there are still, on average, about 45,000 new cases of COVID-19 being reported each day. The virus has been evolving into new variants that our current vaccines are unable to combat. COVID-19 is especially deadly against those that are considered immunocompromised, specifically individuals that are of 65 years of age or older.

This age group is the exact demographic that our client, TMF, has asked us to work with. TMF is a Quality Innovation Network for Texas, Arkansas, Nebraska, Mississippi, Puerto Rico, and the Virgin Islands. They provide business value by improving care for Medicare beneficiaries with a focus on various topics, including COVID-19. Because of this, one of their main clientele are nursing homes in these regions in the United States. Nursing homes are required to report on COVID-19 data to the CDC’s NHSN platform. This data is used to assess the impact of COVID-19 through facility reported surveillance data. By analyzing this data, TMF can identify facilities that are at high risk for COVID-19 outbreaks and allow for early intervention.

TMF has come to us to help them do exactly that. Our goal is to create a predictive model that will use this data to be able to accurately predict nursing home facilities that are most likely to have a COVID-19 outbreak based on various features provided by the data. In addition, we want to be able identify which of these features will predict a high likelihood in causing a COVID-19 outbreak.

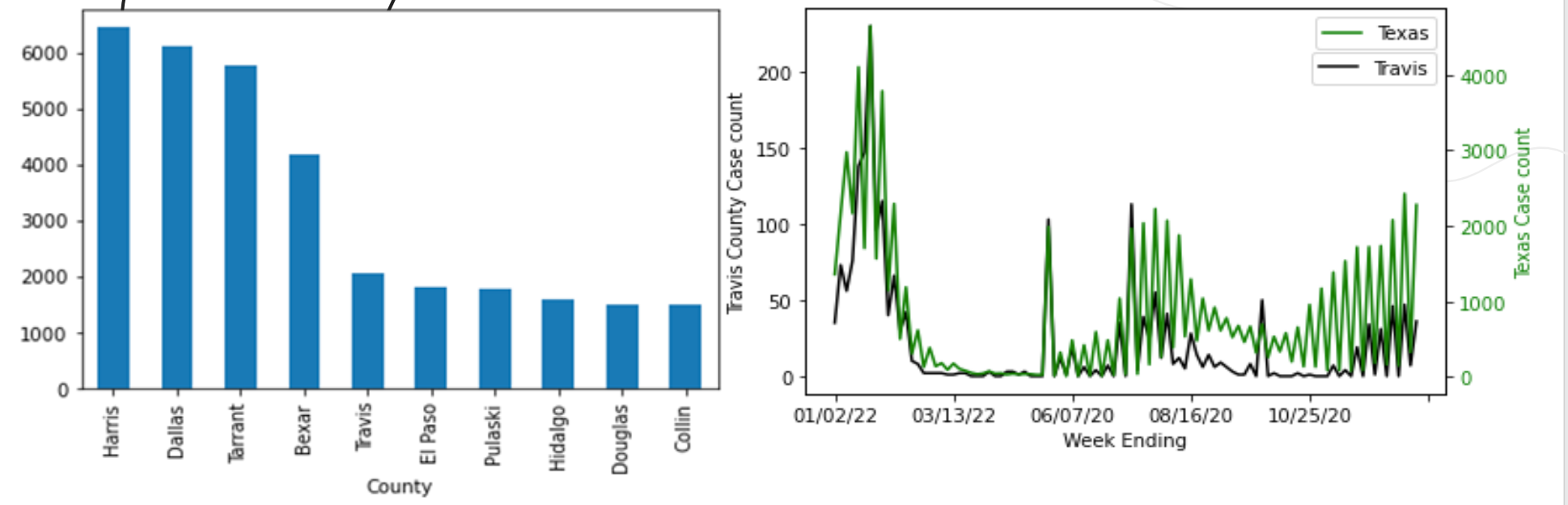
## Data Exploration

We used two data sets to predict nursing home facilities that are most likely to have a COVID-19 outbreak

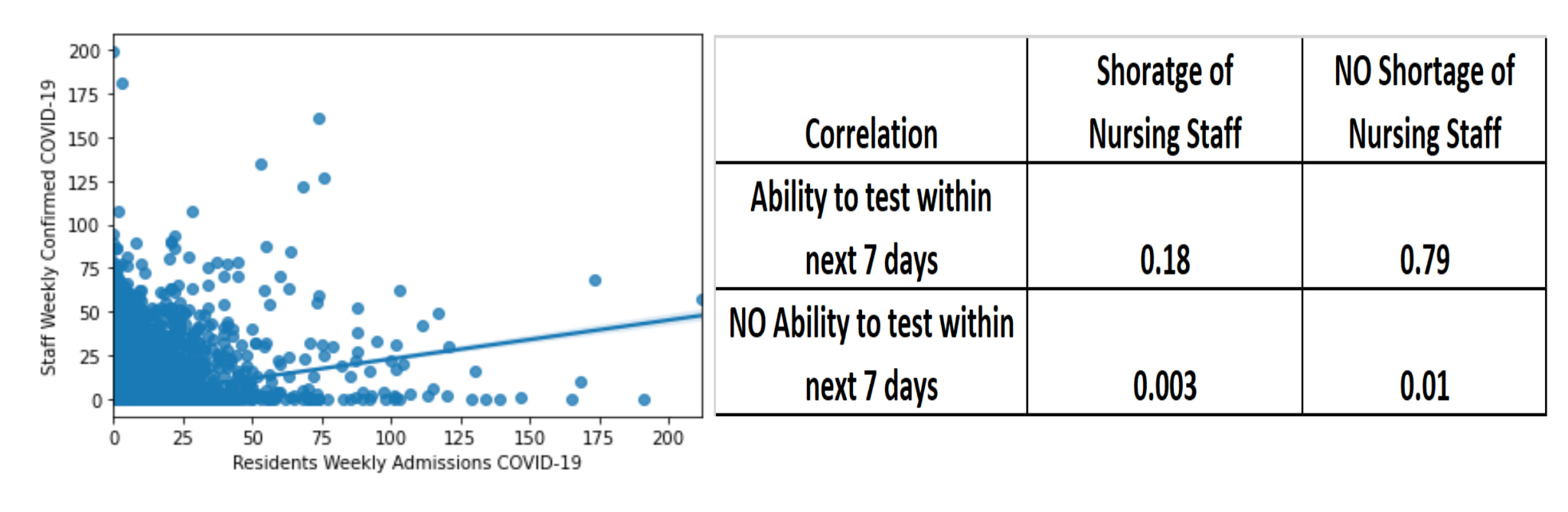
* Nursing Homes dataset which has cases at nursing home/clinic level and has close to 1.5M observations with 183 features
* Provider dataset which has localized trends to predict outbreaks has 15k observations with 96 features

Upon doing analysis on missing values on the combined data sets, 32% of the features have more than 50% of the data missing. And when we checked for the 3 features TMF Operations is interested in( Shortage in Nursing Staff, Shortage in Aides, Shortage in Clinical Staff), there are only 1.6% missing values for these features. However, nearly 20% of the observations have shortage in Nursing staff/aides and around 2% have shortage in clinical staff.

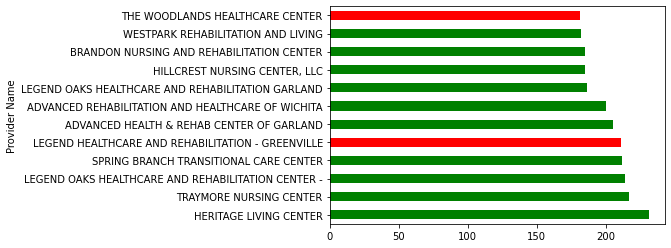
We have done feature analysis from the combined data sets and below are some of the interesting observations



* County's with more covid cases are from Texas state followed by arkansas. Harris county has the highest number of covid cases
* If we compare the covid cases in Texas vs Travis county, there is a similar trend although at a different scale at county and state level

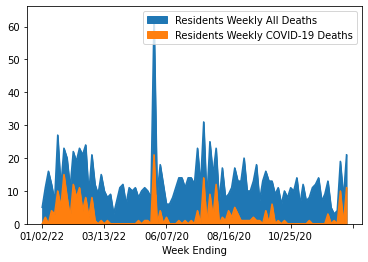


* We observed a weak Correlation between Residents and Staff affected with COVID. And the amount of correlation between these 2 features is only 0.15.
* When we checked the trend between shortage of staff and ability to test, interestingly we found out that even though there is shortage in nursing staff in some clinics, it did not hinder covid testing and 97% of the times, clinics were able to test irrespective of availability of nursing staff.



(The red bar represents shortage ;green bar represents no shortage and the bars themselves represent the number of covid cases at clinic level)

Furthermore on shortage of staff and aides,approximately 1 in 5 clinics have a shortage of nursing staff and aides



We considered only Travis county data for the above graph and there is an observable trend for the number of deaths caused by covid to the overall number of deaths, however the number of deaths due to covid has reduced a lot for a prolonged period after a rise again from Dec 2021.

## Data Cleaning

As mentioned above, this is very messy data - over half of the values in the dataframe are missing. We can solve this to an extent with data interpolation - filling in each missing value with a guess of what it might be. But interpolation doesn’t work very well if most of the values for a given column are missing, there’s just too many unknowns to have the results be meaningful.

So we have to throw out the columns that have the most missing values, but what the cutoff should be is a hyperparameter. We tried several different values to find the optimal cutoff point, and it turns out our models do the best when we throw out all columns with more than 50% missing values, and interpolate the rest. There are a few different options for interpolation, but we found that the nearest neighbor method worked the best for us.

We also threw out every row that was missing the weekly covid cases value, and every row that didn’t meet the CDC’s data quality check (encoded in the column “Passed Quality Assurance Check.”) We then dropped all data from homes that were extreme outliers - those that never had a covid case and those that had a weekly average of more than 4 cases (which was very few homes). Many models get confused when trained on data with heavy outliers and will overfit, doing worse on the majority of the data to do better on a small handful of data points.

As an extra level of feature engineering, we clustered all 15,000 homes into 50 categories, and added those one-hot-encoded cluster labels to our dataframe. This should help our models learn which homes are similar to each other, and we did see a modest boost in performance with many of our models after including this.

## Data Encoding

For most of this project, we were trying to predict how many cases each home will have each week, because more granularity is usually more useful. However, we ran into a problem. Unlike the U.S. population at large, nursing homes didn’t experience much in the way of successive waves of covid cases. They had a large initial spike, then died down to relatively low numbers, with small bumps for Delta and Omicron. Each home had weeks and weeks of no cases, and we’re asking our models to predict when it might suddenly have one or two cases. That’s largely random, and predictably our models didn’t do very well - only a few did better than simply guessing zero cases every week for every home!

To fix this, we tried aggregating all homes in each county. Now, there were on average 6 cases each week in each county! There is a lot of variation, and a lot of counties with zero cases weeks in a row, but we’re no longer trying to predict the first case in weeks for each home. This gives our models a lot more room to work with, and predictably they do much better. This doesn’t come without drawbacks - it would be more useful to be able to point to individual homes and say they have a high chance of having an outbreak next week. But our models simply weren’t able to do that with a high enough degree of accuracy, and due to community spread the number of cases in different homes in the same county will likely be very similar.

Next we needed to create lag variables of our predictors. This is a time series dataset, and as such it’s probably very useful for our models to know how many cases each home has had for the past several weeks. If cases went up last week, there’s a good chance they’ll go up again this week. This is a little tricky for our dataset though - each home is a separate time series, and we don’t want to encode lag variables between homes. As such, we must create lagged dataframes for each home separately, and then stack them on top of eachother, lastly sorting by week so they’re in the right order.

For the final data encoding step, we want to standardize all of our predictors. Some models are sensitive to differences in scale between predictors - if variable A goes from 1-10 and variable B goes from 10-100, some models will assume B is ten times more important than A, even though it simply is on a different scale. We can fix this easily with standardization, which transforms each predictor into a vector of mean 0 and standard deviation of 1.

## Data Modeling

For a large dataset like ours, we were fortunate to have many different Data Modeling options. We chose four key modeling techniques to make future predictions of COVID-19 cases. They were: SIR Model, Linear Regression (Ridge, Lasso), Neural Networks, and Ensemble Methods.

The SIR Model is a compartmental model that uses a system of differential equations to predict the spread of infectious diseases like COVID-19. The model intends to predict future trends for time series data by taking three main inputs:

* **Susceptible:** The number of susceptible individuals. When a susceptible and an infectious individual come into "infectious contact", the susceptible individual contracts the disease and transitions to the infectious compartment. In our case this is the total number of patients in the nursing home.
* **Infectious:**The number of infectious individuals. These are individuals who have already contracted the disease and are capable of infecting susceptible individuals. In our case, these are the total number of active COVID-19 cases in the nursing homes for any given week.
* **Removed:** These are the number of recovered (and immune) or deceased individuals. In our case, these are the nursing home residents who have fought the disease or passed away from it in any given week.

While the model has been very successful at modeling infectious diseases for decades, we were unable to get good results for our data. The model doesn’t take re-infection into account, which is common with covid, and doesn’t handle peaks and troughs in the data well. The R-squared value was extremely low at ~9% and hence this model was dropped.

We then moved on to machine learning methods. We tried seven unique models - ridge regression, lasso regression, a decision tree, a bagging regressor, a basic feed-forward neural network, a recurrent neural network, and an LSTM recurrent neural network. These were evaluated by their root mean squared error (RMSE), which is how many cases off the true value they were each week. For example, an RMSE of 10 implies that on average that model’s prediction is within 10 cases for any given week and county.

These models were trained with 5-split time-series cross-validation. In the first split the models were trained on the first 20% of the data and evaluated on the second 20%. For the second split they were trained on the first 40% of the data, and evaluated on the third 20% of the data, and so on. Their RMSE for each split is then averaged, which we compiled into the below table.

|  | Ridge | Lasso | Decision Tree | Bagging Regressor | Feed-  Forward Neural Net | Recurrent Neural Net | LSTM Neural Net |
| --- | --- | --- | --- | --- | --- | --- | --- |
| RMSE | 9.74 | 10.05 | 14.63 | 9.96 | 15.35 | 163.97 | 36.16 |

On evaluating our model results, we first notice that sometimes simplicity is better than complexity. Our linear models (Ridge and Lasso) performed much better than the complicated models as proved by their lower RMSE. This is great news, because linear models are much more explainable than more complex models, and with these models we would be able to say which features are most important..

We decided on Lasso as our model of choice. Though it has a slightly higher RMSE (10.05) compared to Ridge (9.74), it comes with certain in-built advantages. While both are explainable, Lasso only uses the most important features (only 8 features in our case were used), while Ridge makes use of all several hundred predictors, which makes understanding how it works more difficult.

For ease of explainability, we also presented our client with the data in an easy to understand Tableau dashboard that showed them the raw data provided, predicted cases from our model and the actual cases that occurred based on the data they shared. Several filters were added to improve accessibility and to enable them to visualize the data in many different ways so that our analysis had useability across their diverse internal departments.

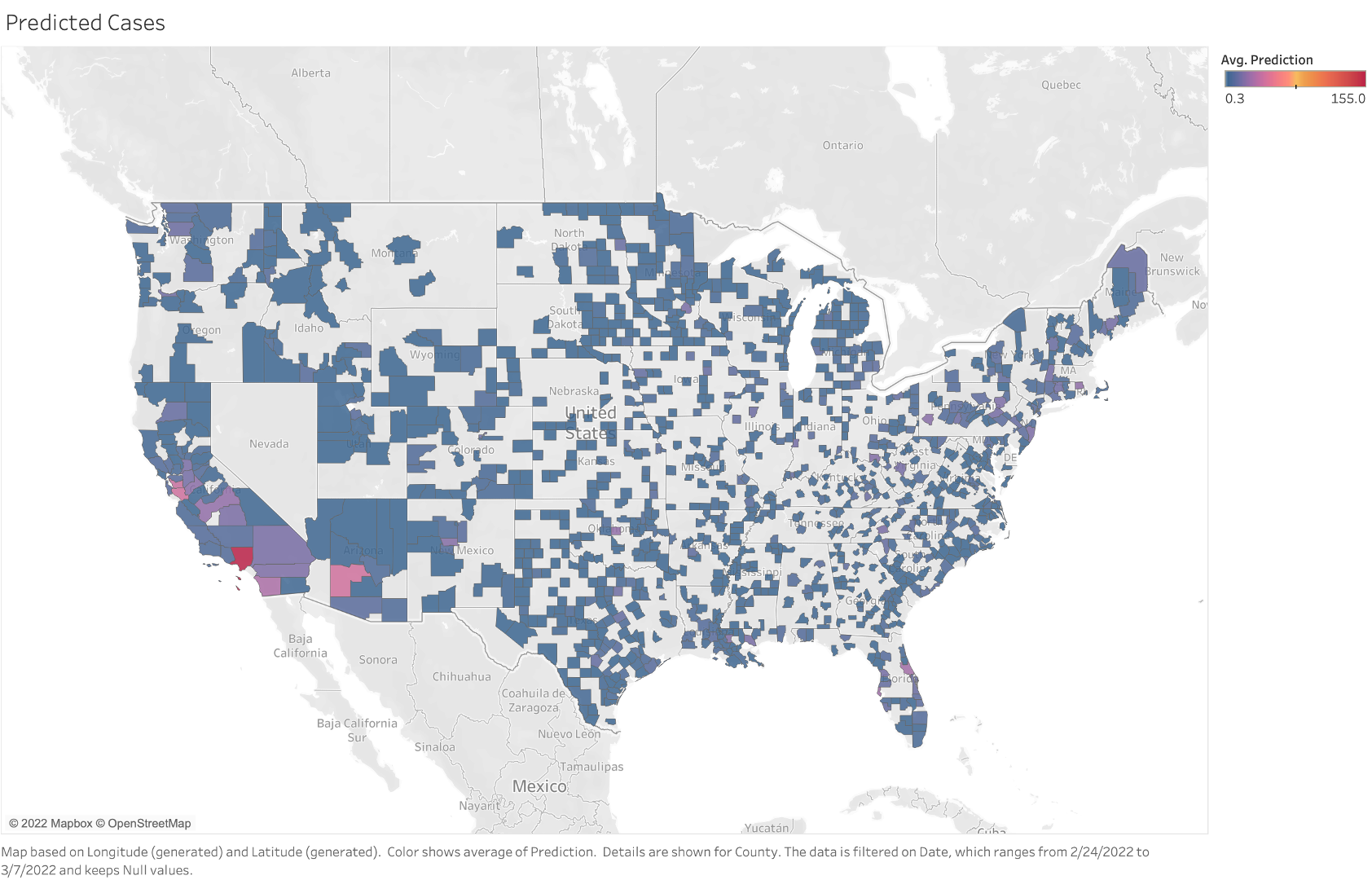
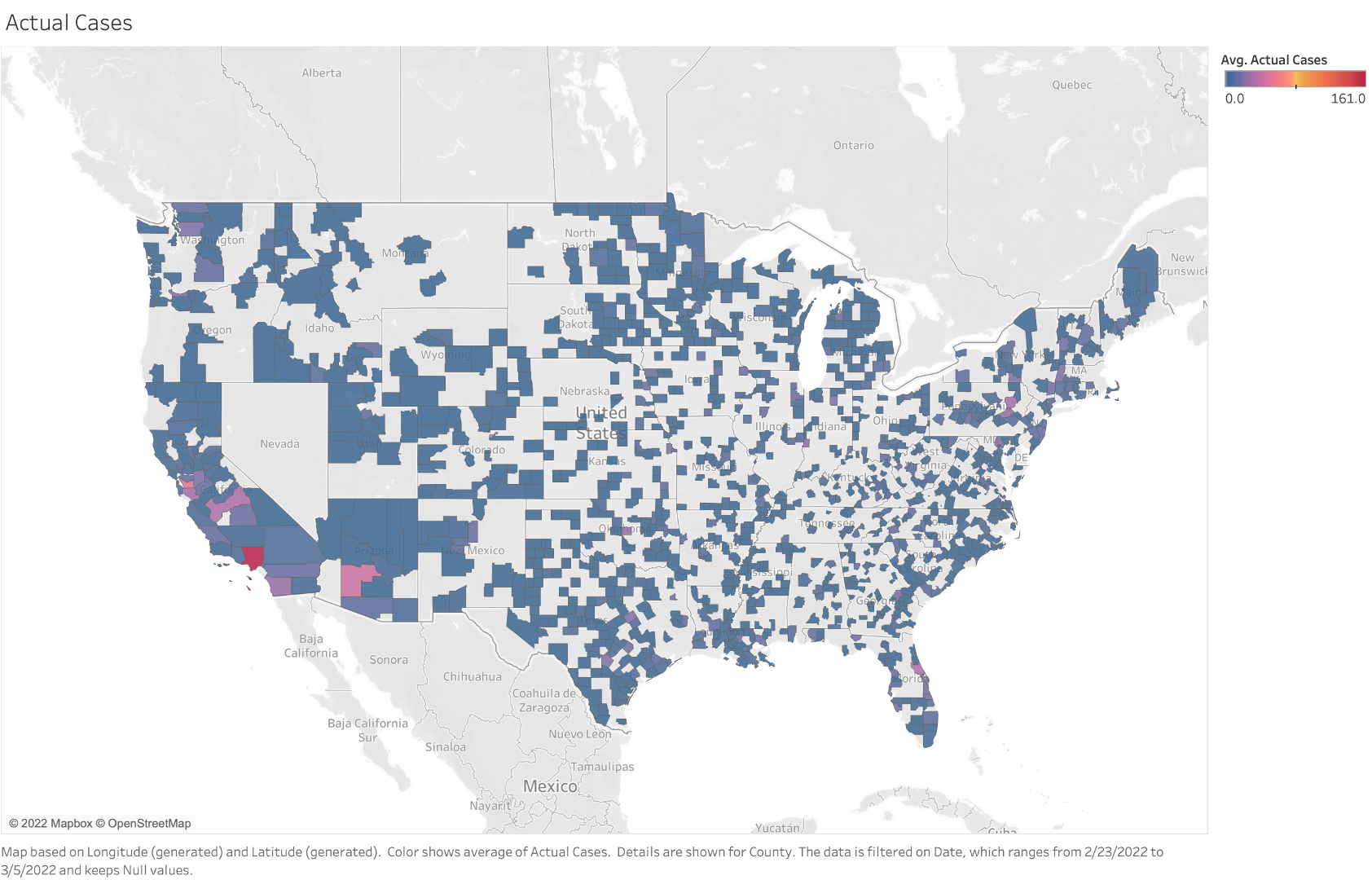
## Conclusion

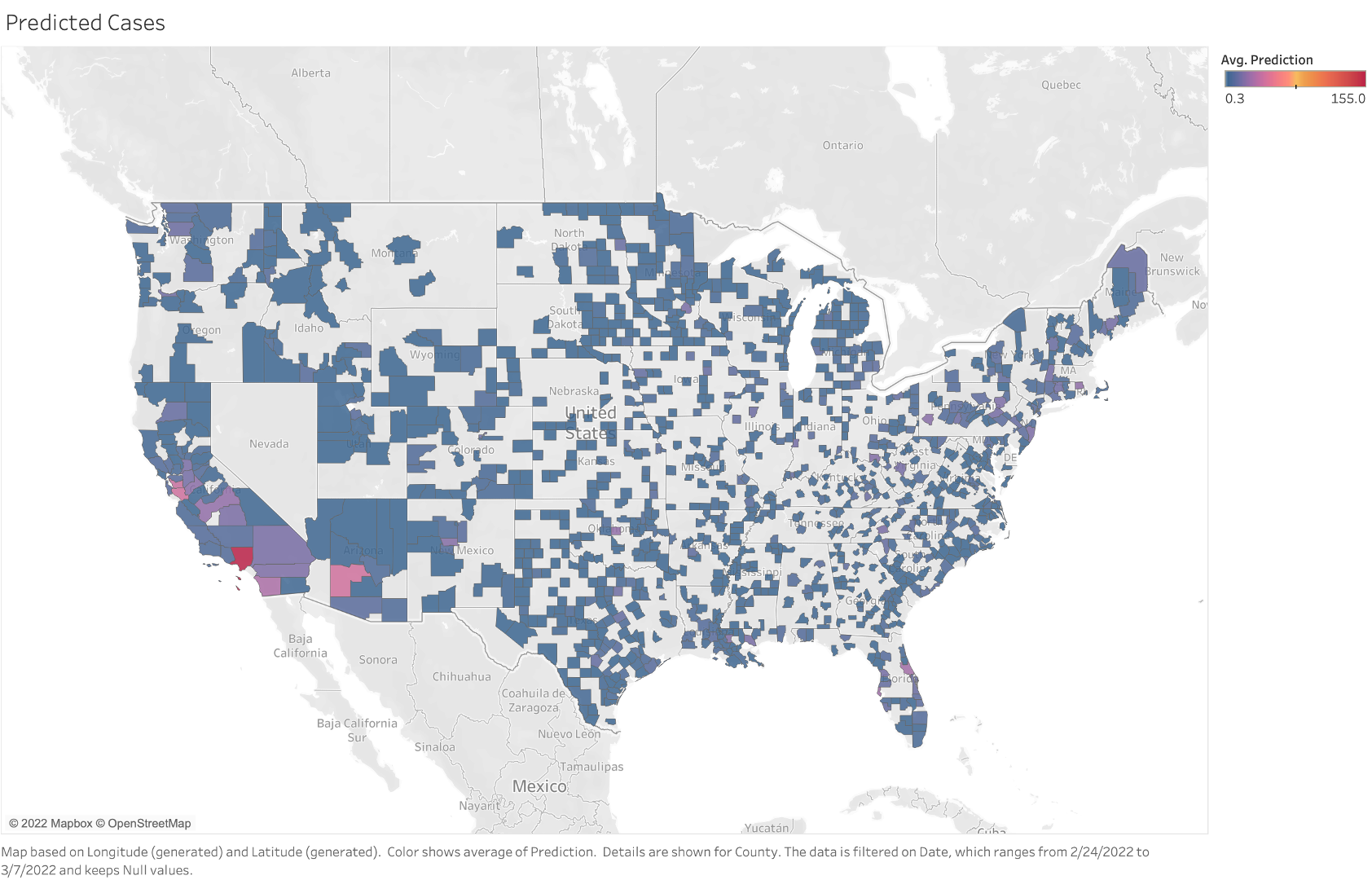
One of the best things about linear models is their explainability - we can look at the weights of each predictor to see what is most important for predicting Covid cases. And Lasso is particularly useful thanks to its strong regularization, forcing it to choose very few predictors to use for its prediction. Below are the features our Lasso model chose, as well as their coefficients.

| **Coefficient** | **Feature** |
| --- | --- |
| 8.74 | Resident Cases 1 Week Ago |
| 6.70 | Staff Cases 1 Week Ago |
| 3.56 | 3 Or More Total Cases 1 Week Ago |
| 1.46 | Resident Positive Covid Tests 1 Week Ago |
| 1.13 | Initial Covid Cases 1 Week Ago |
| .66 | Resident Positive Covid Tests 1 Week Ago (PCR Only) |
| .33 | Initial Covid Cases 2 Weeks Ago |
| .06 | Initial Covid Cases 3 Weeks Ago |

As expected, having high Covid cases last week means high cases this week. What’s really interesting are all the features that aren’t here - our model was given hundreds of predictors, but it threw out all but these 8. That means features that TMF was particularly concerned about - such as shortage of staff or inability to test - weren’t useful for predicting future cases. With these 8 features alone, our model was able to get within 10 cases on average for each county.

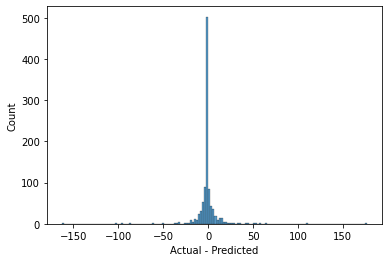
To give you a better understanding of how good this is, here are two maps of the US - one showing actual cases for a particular week, and one showing predicted cases. As you can see, they are nearly identical, with only minor differences.





Actual Cases Predicted Cases

We also visualized the residuals of the test set predictions of our model, and found that it does not systematically overestimate or underestimate, and almost all the residuals are very close to zero.

. 

Overall, we can say with confidence that our model predicts county-level Covid cases well. However, that may not be enough. There are situations where someone will need more granularity in Covid predictions. Even after extensive experimentation, we were unable to create a model that met our expectations for home-level predictions, but that does not mean this is impossible. We used various CDC datasets, including provider-level information and county-level cases and vaccination rates, but there is still so much more data out there. It is possible that by bringing in more outside datasets and through extensive feature engineering, a model that can predict home-level cases reliably could be built.

## References

#### Datasets:

* Nursing Home Dataset: <https://data.cms.gov/covid-19/covid-19-nursing-home-data>
* Provider Dataset: <https://data.cms.gov/provider-data/dataset/4pq5-n9py>

#### Papers Referenced:

Cooper, Ian, et al. “A SIR Model Assumption for the Spread of COVID-19 in Different Communities.” *Chaos, Solitons & Fractals*, vol. 139, Oct. 2020, p. 110057, 10.1016/j.chaos.2020.110057.

Dama, Fatoumata, and Christine Sinoquet. “Time Series Analysis and Modeling to Forecast: A Survey.” *ArXiv:2104.00164 [Cs]*, 25 Sept. 2021, arxiv.org/abs/2104.00164. Accessed 14 May 2022.

“Coronavirus Papers: Viral Proteins Point to Potential Treatments.” *Nature*, 27 Mar. 2020, www.nature.com/articles/d41586-020-00502-w, 10.1038/d41586-020-00502-w.

Alger, Charlotte, and Kaitlin Todd. *The SIR Model of Disease Spread*. 2015.

Liu, Meng, et al. “Forecasting the Spread of COVID-19 under Different Reopening

Strategies.” *Scientific Reports*, vol. 10, no. 1, 23 Nov. 2020, 10.1038/s41598-020-77292-8.

## 

## Acknowledgements

We as a team would like to express our great appreciation and gratitude to Prof.Daniel Mitchell and Prof. Robert Hammond of the University of Texas at Austin for their valuable and constructive suggestions during the planning and development of this research work. Their willingness to give their valuable time and guidance was critical to the success of this project.

We would also like to thank our client TMF for allowing us the golden opportunity to work on solving such a highly-relevant yet complex problem. Additionally, we would also like to extend our gratitude to Ms. Aradhana, Ms. Bethany and Ms. Piyusha for keeping us on track and providing us with all the requested resources for the successful completion of this endeavor.