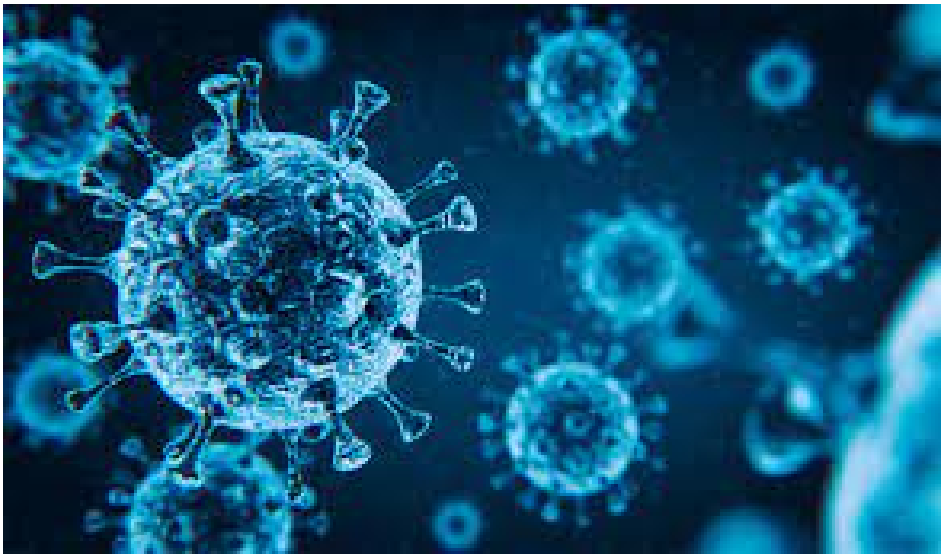


Predicting COVID-19 Infection Rates for Different Zip Codes in Chicago

Client: Real Estate Agencies in Chicago

Authors: Tongshu Wu, Mario Martino, Jacklyn Clauss, Gabriella Rub, Shuang Lin

I. Executive Summary

Since 2020, the onset of the COVID-19 pandemic has affected lifestyles, schedules, as well as decisions. Our objective is to provide information regarding disease data so that individuals are better informed before making decisions. Specifically, we focused on information regarding buying a house in a new neighborhood. For this report, we used COVID-19 data from various Chicago regions. Based on different zip codes, we analyzed the outbreak of the disease in various parts of Chicago. This data includes COVID-19 data from January 2020 to October 2022, which allows us to analyze this data in both time and location.

Our target audience is real estate agencies in Chicago, who will be relaying information to customers about rates of COVID-19 in desired locations. Specifically, we would like to predict the COVID-19 infection rate in certain counties in Chicago, and also compare the severity of the disease in the different zip codes. By using this data, customers who are worried about the disease will be able to make a decision on whether or not they would like to buy a house in that location.

We carried out the prediction and analysis of multiple models on the data. We used the Time Series model to estimate the pattern of COVID infections in each region in mid-age, monthly, and seasonally. For the prediction of the number of weekly infections, we also used several different models to make judgments. Logistic regression, Naive Bayes, decision tree, and K-NN. In each model, we used 10-fold validation to split the data to get the highest accuracy of the model. We take the data predicted by different models and compare them to see the differences. At the same time, we also used the map data of Chicago to show the data analysis in a dynamic way. The following sections will cover some of the specifics of this project in more detail.

II. Data

Data Processing

Our data, published by Data.gov, was collected from various zip codes in Chicago. Specifically, it contains data on COVID-19 cases, tests, and deaths by zip code from November 10th, 2020 to October 14th, 2022. It consists of 8,460 rows as well as 22 unique attributes that were separated by weekly vs cumulative. Based on this information, our group decided on a target variable describing the number of cases in one week in that specific zip code: Cases_Weekly. Using four models - logistic regression, naive Bayes, decision tree, and k-nearest neighbor, we hope to do an analysis to predict the COVID-19 infection rate in different Chicago zip codes and compare the severity of the disease in the different locations. By using this data, our target customers - real estate agencies - will better inform their customers who are worried about infection rates before making a purchase.

Data Cleaning

Our first step in the data cleansing process was to observe any patterns or information that might not be necessary, or might have a negative impact on our analysis. After importing our data, we noticed that each week has a number of records for that current year, for a total of 53 weeks each year. We also noticed that the population does not change with the year; it stays constant in its specific zip code. Utilizing these observations, we performed the following steps to clean the data:

- Removed unnecessary attributes from the data set
- Removed "unknown" zip codes
- Removed zip code 60666, because it is an airport with a population of 0.
- Convert Week.Start to R-compatible date type
- Sort the data frame by ZIP.Code, then Week.Start date, then Week.Number.

The removal method was simply converting all unwanted items to NAs and omitting them. After cleaning, the data frame was left with 7987 rows and 8 attributes. The data frame is now chronological by zip codes, and can easily implement time series.

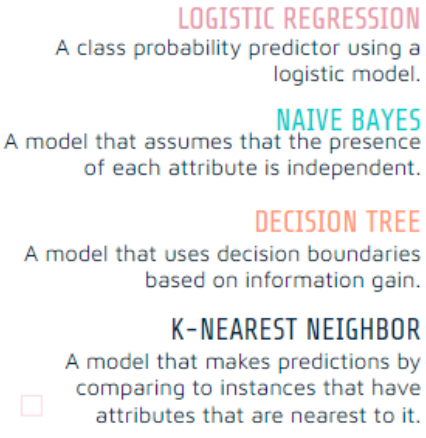
Time Series

We have added 4 new time-series attributes which allow predictions of future cases based on historically changing data. It can be inferred that infection rates change at different times of the year. Attributes of Year and Month were extracted from the Week.Start attribute, now that it is formatted. The attribute Prior.Week.Cumulative was created to hold the previous week's cumulative cases. Lastly, the Prior.Week.Rate attribute was calculated by last week's cumulative over its population. After setting up the time series, the data frame results in 7987 rows with 12 attributes. The data frame is, then, randomized to allow proper cross-validation.

III. Modeling

Overview

Four modeling techniques were used: Linear Regression, Naive Bayes, Decision Tree, and K Nearest Neighbor. Each is stored with their respective nicknames appended: glm, nB, tree, and kNN. In building the models, the same 8 attributes were used for all of the models to ensure no outstanding advantages were given to a specific model. The kNN model was manually tested to have its distance = 1 and k = 9 achieving the best results.



```
  model_glm <- glm(formula = Cases...Weekly ~ ZIP.Code + Week.Number + Week.Start + Population + Year + Month + Prior.Week.Cumulative + Prior.Week.Rate, data=train)
```

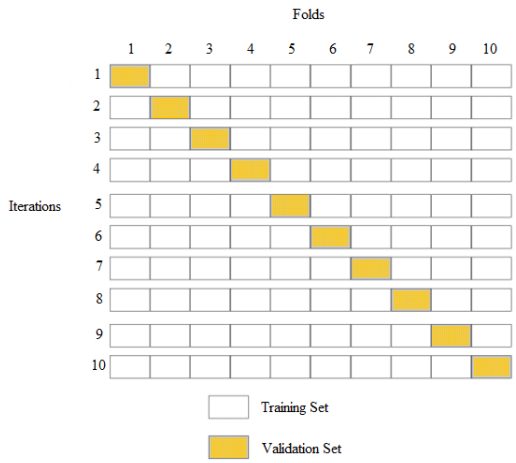
```
  model_nB <- naiveBayes(Cases...Weekly ~ ZIP.Code + Week.Number + Week.Start + Population + Year + Month + Prior.Week.Cumulative + Prior.Week.Rate, data=train)
```

```
  model_tree <- rpart(Cases...Weekly ~ ZIP.Code + Week.Number + Week.Start + Population + Year + Month + Prior.Week.Cumulative + Prior.Week.Rate, data=train)
```

```
  model_kNN <- kknn(Cases...Weekly ~ ZIP.Code + Week.Number + Week.Start + Population + Year + Month + Prior.Week.Cumulative + Prior.Week.Rate, train, test, distance=1, k=9)
```

Cross Validation

A 10-fold cross-validation method was used to randomly sample 10 equal parts of the data frame. This same method was used to train 10 models for each modeling technique, resulting in 40 total models built. Due to the uneven row numbers, the partitioning was rounded up, which created empty rows that had to be removed from the folds. Ultimately, each model was trained with 1 different training and validation set from the other models.



IV. Analysis

Evaluation

We calculated various accuracy measures for each of the models that were created. Since we used 10-fold cross-validation, each of these measures was calculated for each of the 10 models of each type. It is important to note that the exact numbers that were found will vary each time the program is run as the folds are selected randomly, and therefore the models created will differ somewhat.

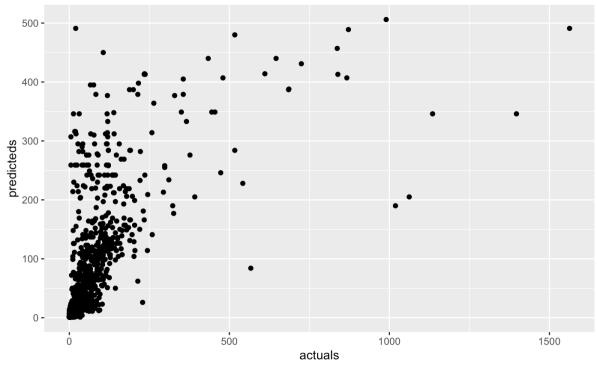
For each model we calculated the accuracy, precision, recall, mean squared error, and mean absolute percent error. These values could then be used to determine how effective each model was. While accuracy, precision, and recall are intended to be used on classification problems, our model was predicting the number of cases, and therefore was not sorted into discrete categories. This is why these values are extremely low, as they are not an accurate view of the effectiveness of each model. A better picture of how well each model worked was given by the mean squared error and the mean absolute percent error, as these are error measures intended to evaluate regression models.

Error Analysis

Our logistic regression model has a warning “prediction from a rank-deficient fit may be misleading” because some attributes are perfectly correlated. Additionally, some folds created were troublesome due to sorted order from cleaning. Our solution to this problem was to randomize our data again after adding sorted time series, varying our predictions each time. In terms of our evaluations, because our folds were created randomly the exact values for the accuracy measures will be slightly different every time the program is run. We also did not implement a ROC graph as an evaluation metric because our target variable is not used for classification.

V. Results

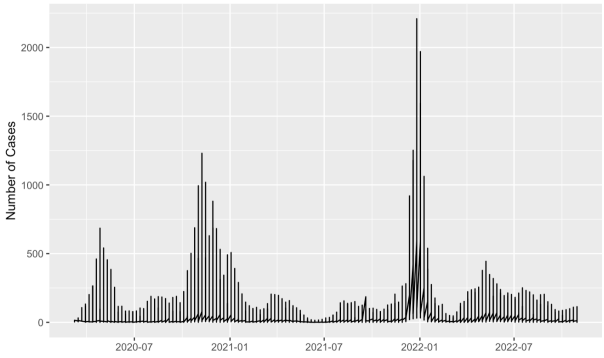
Using the accuracy, precision, recall, mean squared error and mean absolute percent error calculated for each model, we were able to compare the performance against each other. Although accuracy may seem like the strongest indicator of a model’s performance, that is not the case for our analysis. We decided on the best accuracy measure for our situation. Instead, we ranked our models mainly by mean absolute percent error. This result is Naive Bayes having the least error. This could be true for a number of its characteristics, including being highly scalable with the number of data points and not sensitive to irrelevant features. Below is the scatterplot of the actual values versus the predicted values. As you can see, most of the values fall toward the origin, where the error is smaller. Overall, we were able to take a large and rather unattractive dataset, and transform it into a predictive model with acceptable accuracy using the Naive Bayes model.



VI. Imaging

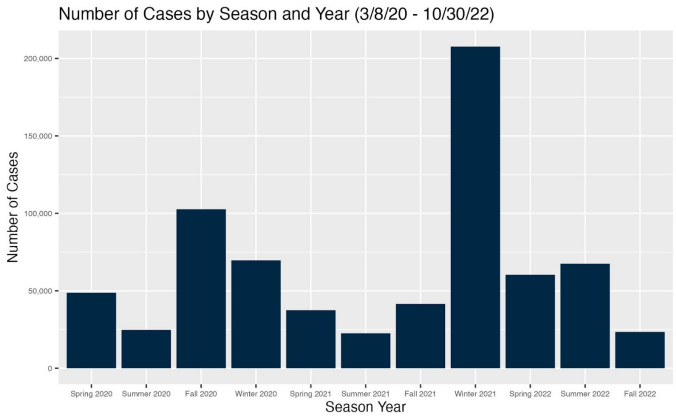
Time Series Analysis

Being sorted by week, one of the first ideas we had for our data was to create a time series. A time series gives a good depiction of how the data varies over time. In its simplest form, we created a time series for weekly cases over time, which can be seen below. As you can see, the values are fairly low with a few spikes over the given time period. With a disease like COVID-19, this makes sense as it is a highly contagious disease that is usually effective for one to two weeks.



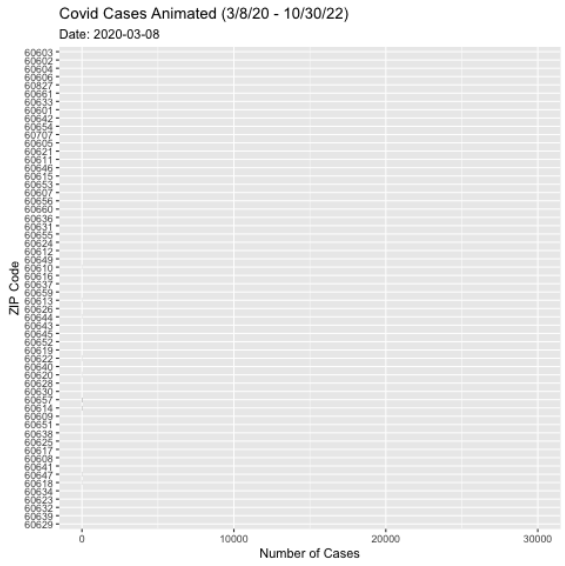
Seasons

Many diseases can be attributed to seasonal change. For example, the flu season is largely regarded as the fall and winter months. Next, we wanted to find out if it was the same case for COVID-19. Using the created time series, we were able to group data by season. We then graphed it using a bar chart. Below is a bar chart that shows the number of cases grouped by season year. Analyzing the chart, there is no obvious trend, but there does seem to be a greater amount of cases during the fall and winter seasons.



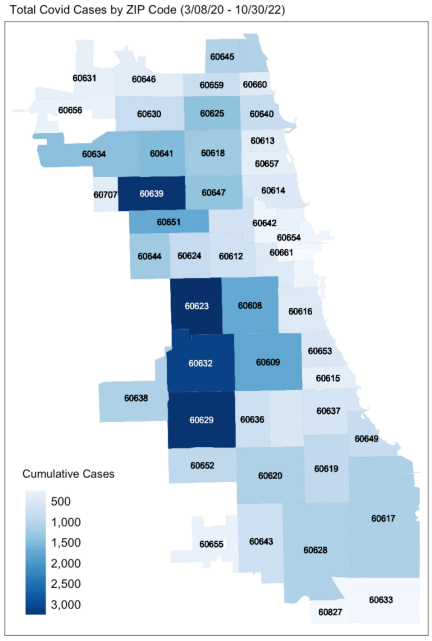
Bar Chart/Animation

As a more customary visualization, we decided to create a bar chart that shows the total number of cases per ZIP code. To give us a better understanding, we animated the bar chart to update for values of cumulative cases for each week. The bar chart updates for each week over the given time period. As you can see, ZIP codes such as 60629 and 60639 are hot areas in terms of cumulative cases.



Heat Map

As our last main visual, we decided to create a heat map based on ZIP codes in Chicago. Using the same data to construct the bar graph above in conjunction with a shapefile that also includes ZIP code classification. After merging the two datasets, we came up with the heat map below. Darker areas correspond to higher case totals over the given time period.



VII. Conclusion

The goal of our project was to be able to predict the number of covid cases in different zip codes within Chicago. We first had to clean and modify the data by removing unnecessary attributes, and rows with N/A or unknown values, and converting the start and end dates from strings into the date format in R. We then added a time series component by adding the previous week's cumulative cases and case rate. This data was then split into 10 different random folds to perform 10-fold cross-validation. We then built four different types of models: Naive Bayes, logistic regression, k nearest neighbor, and a decision tree. Various accuracy measures were calculated for each of the models, but the mean absolute percent error was given the most weight in deciding which of the models should be used. In the end, it was decided that Naive Bayes should be the model that is used, as it generally had the best performance when evaluating the accuracy measures.

Additionally, although we were not able to make a ROC curve, we made several other visualizations of our data and results, including a heat map of covid cases, as well as an animated bar graph that shows how cumulative cases changed over time. To get a visual of our accuracy, we plotted predicted cases vs. actual cases on a scatter plot. This introduced an interesting visual that let us get a better idea of what the accuracy looked like.

Our target audience for this project was real estate agents, who would want a resource to be able to reference when discussing with clients who want to avoid high-risk areas for covid-19. Since we were able to create a model successfully modeling the data, this would be of interest to our target audience.

With more time to work on this project, it would be beneficial to add more models, including some more complex techniques such as the random forest in order to get the most accurate model possible. It would also be interesting to go without the cross-validation and see how a model would work if it was fed in the chronologically first 80% of data as the training set, and then tested on the most recent data.

VIII. Appendix

Jacklyn Clauss: Helped clean and organize data. Calculated and compared accuracy measures. Wrote the model description, testing and comparing accuracy slides, and helped build the presentation. Presented slides regarding testing and comparing accuracy, and our project's goals. Wrote the evaluation and conclusion sections of the final report.

Gabriella Rub: Helped clean and organize data. Graphed the predicted values against the actual values for each model. Helped build slideshow and edit final presentation. Presented slides regarding our data and its characteristics, k-nearest neighbor, and graph results DT and KN. Wrote the data processing/cleaning and error analysis sections of the final report.

Mario Martino: Helped clean, organize data, and merge datasets. Created map and time series visuals. Helped build slideshow and presented slides for Mapping and Season Time Series. Wrote the Results and Imaging sections of the final report.

Tongshu Wu: Helped clean and organize data. Helped build the logistic regression part of the data and analysis. Built slides for the introduction part, intended audience and summary part of the presentation slides. Presented the introduction part and logistic regression part of the presentation. Wrote the executive summary part of the final report.

Shuang Lin: Aid in data studying. Data cleaning, sorting, and setting up time series. Built the cross-validation method, the four models, and their predictions. Calculated accuracy, precision, and recall. Presented 4 slides: Data Preparation, Time Series, Building Models Cont., and Graph Results: DT. Wrote Data Cleaning and Time Series sub-sections of the report. Wrote section 3, modeling, and drew its 10-fold validation diagram.