Covid-19 Awareness and Covid-19 cases in Ohio

* Kaggle Challenge-1

1st Srishti Todi
MS in Data Science
University of Rochester
Rochester, USA
stodi@ur.rochester.edu

2nd Arunaggiri Pandian Karunanidhi
MS in Data Science
University of Rochester
Rochester, USA
akarunan@ur.rochester.edu

***Abstract*—The data that we used for training had 3141 rows and we had to predict the number of cases for the test data which contained 7331 rows. After we combined the data, we observed that there were 119 rows or days for each county out of which some were in test data that had to be predicted. We approached the problem in two ways: first, is a machine learning model and second is a rule-based approach, where we have used the observations from the data to predict the cases in the test data. We have explained more about this in the methods section. Since there were 144 columns in the data, feature engineering was the most important part of the modelling, for which we chose Correlation Analysis and Recursive Feature Elimination. In total, we used 11 features. We had to predict the number of cases in the county on certain days given the features. We realized that it's a regression problem and approached the problem using regression-based algorithms. For training the data, we tried several methods like Random Forest, Decision trees and XG Boosting of which we got the best results for decision trees with an R2 score of 0.98 on the training data. Both methods gave us similar R2 scores on the test data. However, the rule-based approach gave slightly better results than the machine learning model.**

I. Introduction

The training and testing data were combined to see the relation in the number of days and cases in the county. We noticed that there were exactly 119 rows for each county in the combined dataset. We also noticed that the data is not linearly dependent. There is an exponential relation between the number of days and the number of cases in each county. We also found some outlier values in the cases like the county Cuyahoga had the maximum number of cases of 13566. On more exploration, we also found that for all the counties, the value of cases up till day 70 was 0 and it did not increase at all till tis date. There were also 3 counties that had the value of 0 cases till day 100. These counties were Hocking, Vinton and Harrison. Since, we now knew that the data is not linearly related, we used rule-based algorithms like Decision Tree and Random Forest to train our machine learning algorithm. With the features we used for training the model, we found the following relations:

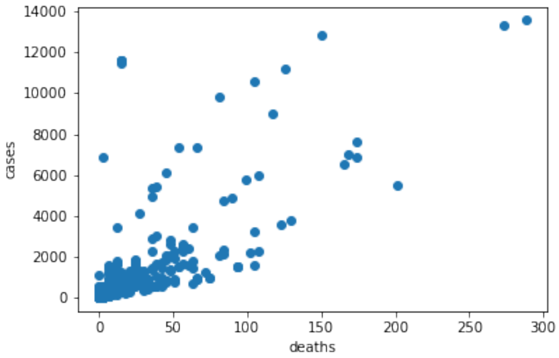

Fig. 1. Deaths vs Cases

We can see that deaths and cases have a linear relationship. They also have a high positive correlation. Next, we see that day number and cases have an exponential relation. The other graphs shown here are the features that had a high correlation with the cases. We can see from the graph that poverty rate also has a direct relationship with cases. It also makes sense as high poverty means less availability of resources. Similarly, unemployment rate also has a high correlation.

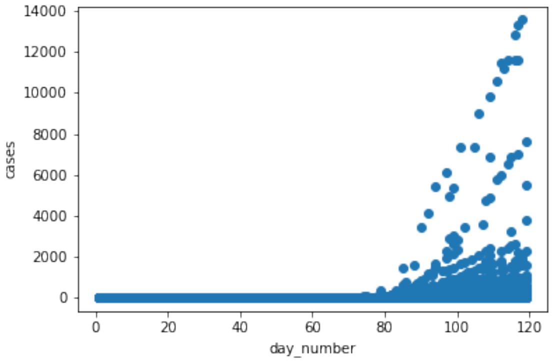

Fig. 2. Day Number vs Cases

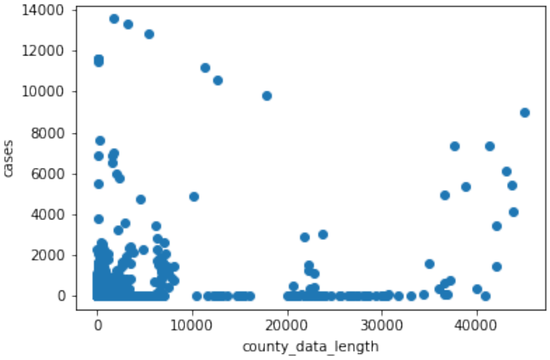

Fig. 3. County Data Length vs Cases

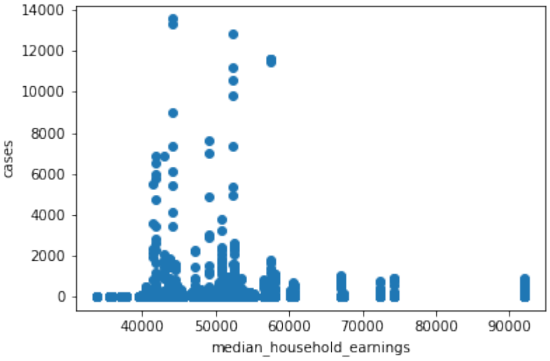

Fig. 4. Median earnings vs Cases

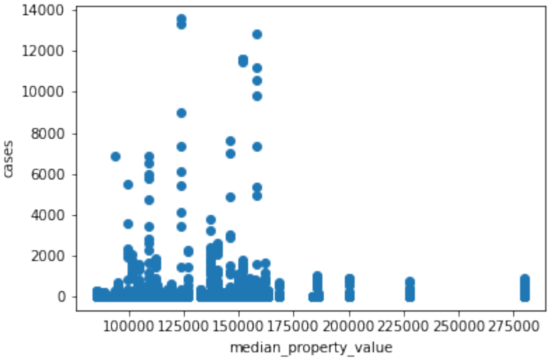

Fig. 5. Median Property value vs Cases

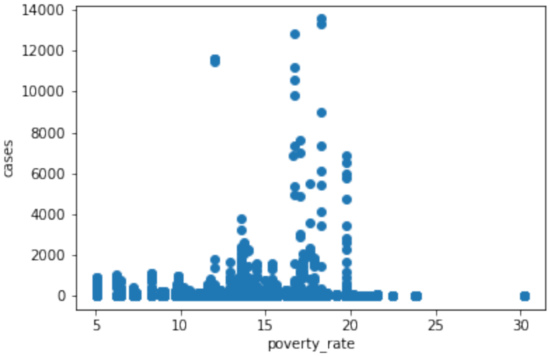

Fig. 6. Poverty Rate vs Cases

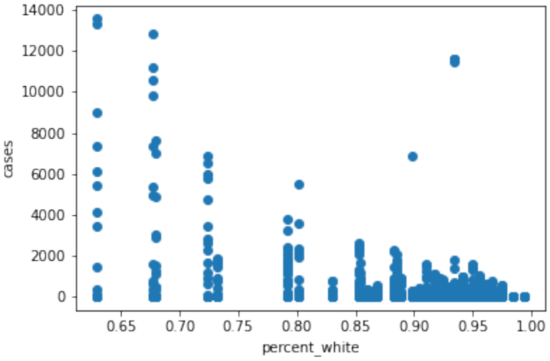

Fig. 7. Percent white vs Cases

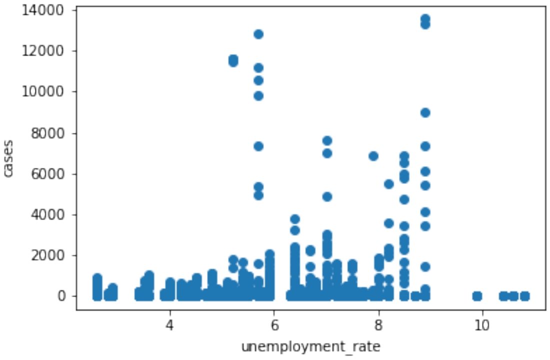

Fig. 8. Unemployment Rate vs Cases

II. Data

We found that Sports and entertainment have the highest values of Jaccard similarity based awareness, indicating a higher level of awareness in these areas. In contrast, health, ideology, and health technology have the lowest values, suggesting lower awareness levels. The remaining topics, such as education, religion, and economy, have moderate awareness levels. It's worth noting that political topics show a slightly higher awareness for Democratic and Republican love compared to hate, but overall, the awareness levels for politics are not as high as sports and entertainment. Delaware has the highest mean awareness with a value of 0.038 followed by Richland and Perry and Paulding has the lowest mean awareness. Therefore, it had the highest level of awareness amongst its residents. This is also reflected by the number of cases which remain 0 till day 75. However, after that there is a sharp increase in the number of cases. On the pother hand, even though the mean awareness of Paulding is the least, it has the 0 cases and 0 deaths till day 94. One of the reasons for it could be the population and the size of the county. After analyzing the line chart of the evolution of awareness levels for different topics, we drew the following conclusions:

race_jaccard_normalized received the highest awareness on day 20, followed by gender_jaccard_normalized on day 68 and sports_jaccard_normalized on day 115 reflecting the highest level of similarity in terms of race-based discussions, showing the greatest similarity in conversations about sports, and demonstrating the highest level of similarity in gender-focused discussions respectively.

We also found that the top five counties with highest number of per capita cases are : Pickaway, Marion, Lucas, Columbiana, Mahoning
The top five counties with highest number of per capita cases are: Miami, Darke, Columbiana, Portage and Mahoning

III. Methods

A. Method 1

Initially, we tried to find whether there is any linear relationship between the features using a scatter plot, but unfortunately, very few features had a slight non-linear relationship with the target (cases). So, from the plots, we could not make much inference. Then, using the correlation between the features, I chose the top 4 positively correlated features, namely "deaths", "total_pop", "day_number" (extracted from "date_index_converted" feature), "county_data_length", and the top 2 negatively correlated features, namely "percent_married", "percent_white". These were selected after many trials and errors, and during the process, adding "percent_25_34" to the correlated features made the ML models perform better. So, we included that as well for training the models. To further improve the model's performance, we employed Recursive Feature Elimination to find a few other features which are of importance, but adding just those and adding those features along with the features selected through the correlation method did not improve the model's performance by giving out a poor R2-value. Then we went ahead and used Random Forest to find the feature importance of each feature to the target, to find a few other features other than the features selected from the correlation method, using trial and error, and found that adding "poverty_rate", "median_property_value", "unemployment_rate", and "median_household_earnings" to the training phase improved the model's performance significantly and gave out an R2-value of 0.9802 to the Decision Tree model. Since the data did not have much linear dependency, we went ahead with models that make predictions based on decision rules. That is one of the reasons why we believe our decision tree model performed better than the other ML models.

Training:
The train data has been preprocessed by scaling the data using the StandardScaler() method and has been tested on 4 different models, namely Linear Regression, Decision Trees, Random Forest, and XGBoost, and the R2 values for each model obtained were 0.4759, 0.9802, 0.8883, and 0.9746, respectively. Since the Decision Tree model performed well, we used that to test on the test set provided for the challenge and submitted our predictions to the Kaggle platform.

B. Method 2

In the second method, we tried the rule-based approach. We combined the two datasets together.

First, we converted the date index converted column to day number with just the day numbers. When we were trying to understand the relation between the day and the number of cases in a county, we noticed that there is an exponential relation between them. When we were understanding the data county-wise and day-wise, wed noticed that for all the counties the number of cases up till day 70 was 0 or NANs. So, we imputed the cases for all the counties till day 70 as 0. It left us with just 3045 empty values to be imputed. For the rest of the days in each county, we decided to make a regression slope between two known points and impute them using the regression. For the next two known points, a different slope would be formed till the last known value. This way, each county had a distribution that comprised a fair number of regression slopes one after the other, giving it an exponential distribution kind of shape. To achieve this, we used back filling and forward filling of the cases in two new columns and then we found the days for which the cases were known in a new column. We did the back filling and forward filling for these days again. Using these values, we found the slope for each two points. Where the value for the back fill and forward fill were same, we imputed the slope values to be 0 there as it means that it has the known value for the cases. Using these values, we predicted the value for each of the cases.

We also observed that all the cases in the training data were a multiple of 7. So, once we predicted the cases, we rounded them off to 7 and converted them to integers. This gave us a score of 0.927 on the test set. The only problem we found with this method is that it's not generalizable. This method is giving us slightly better results than method 1 where we have used machine learning.

IV. Results

A. Method 1

Since the data didn't show strong linear relationships among the features, we decided to focus on models that make predictions using decision rules instead of linear assumptions. This strategy helped us capture the complex, non-linear relationships found in the data more effectively. One of the main reasons we think our decision tree model performed better than other ML models in our method-1 is its natural ability to deal with non-linear patterns.

Decision tree models work by splitting the data into smaller subsets recursively, and then using these subsets to form a hierarchy of decision rules. These rules allow the model to make accurate predictions even when the relationships between the features and the target variable aren't linear. On the other hand, linear regression models assume a direct linear relationship between the features and the target, which might not be true for our dataset. As a result, the decision tree model was better suited to handle the complex relationships in the data, leading to improved performance compared to other models that rely on linear assumptions. For this method, we obtained an R2 score of 0.98 on training data and a score on 0.925 on test data.

B. Method 2

Using the rule-based method, where we modelled our data with the regression slopes at the the known values of the cases,we got the R2 score of 0.927 on the test which is a little higher than the machine learning algorithm model. Because, the data is not linearly distributed, we have tried to give it an exponential shape using the continuous regression slopes and imputing the values between the these values using the slope. However, the problem is that it is not generalizable. The imputation of 0 value till day 70 predicted around 4300 values.

C. Critical Review

Even though our decision tree model performed better than the other models we tried, there's still a chance that different models could give us even better results. We could look into more advanced models like neural networks, support vector machines (SVMs), or ensemble methods (like bagging, boosting, or stacking) to see if they can capture the non-linear relationships and interactions between features better, leading to more accurate predictions.

Another aspect we should think about is our feature selection process. Right now, we used correlation and trial-and-error to choose important features, but there are more systematic ways to do this. Techniques like LASSO, Ridge Regression, or Principal Component Analysis (PCA) could help us find better features or reduce the number of features we need to use. By trying these techniques, we might find new feature relationships or lower the dimensionality of our dataset, which could make our model even better.

In short, although our decision tree model showed good performance, it's important for us to think about trying different models and feature selection methods to make our model even more accurate and able to work well with new data.