McGill | School of Computer Science

COMP 551 - Applied Machine Learning

MiniProject 1: Analyzing COVID-19 Search Trends and Hospitalization

Authors:

- Matthew Kourlas (ID: 260686484)
- Cong Zhu (ID: 260826128)
- Haoran Du (ID: 260776911)

November 3, 2020

Abstract

In this project, we implemented two different models: KNN & Decision Tree based on the *scikit-learn* and other libraries. We evaluate these models on the COVID-19 search dataset. The data sets are retrieved from Google Search Trends, including the US search trends symptom dataset and the US aggregated open COVID-19 dataset. In the data processing stage, we cleaned up the data, performed PCA dimension reduction (including K-means clustering), and data normalization. The two models are trained separately in the two domains of the COVID-19 search dataset, namely the time domain and the region domain. By tuning the hyper parameters and applying K-Fold Cross Validation, generally speaking, we achieved a better performance in accuracy using the k-nearest neighbour regression approach compared to that of decision trees.

1 Introduction

The "open-covid-19-data" reflects the volume of Google searches for a broad set of health symptoms, signs, and conditions related to COVID-19. The data covers hundreds of symptoms such as fever, difficulty breathing, and headaches. The resulting dataset is a daily time series for each region showing the relative frequency of searches for each symptom.

We, therefore, utilized the Google Search Trends data for the weekly symptom search trend data across regions in the US to predict the newly hospitalization cases and cumulative cases. The primary objective of this project is to demonstrate K-Nearest-Neighbor Regression and Decision Tree based on the training of the symptom search trend dataset. After comparing all models, we came to the conclusion that the KNN learning approach in the region domain and time domain prediction is the better prediction strategy for this research.

2 Data Sets

We primarily used two data sets for this project. One is the volume of Google searches for a broad set of health symptoms, signs, and conditions over the period of covid-19 pandemic. The format of data is a weekly time series for each region showing the relative frequency of searches for each symptom. This data set is shown in relative frequency because it has been normalized by dividing each data by the maximum value within each region. In order to process this data, we drop the rows with less than eight input values and the columns that are completely blank. The second dataset we analyzed is the hospitalization dataset. This dataset contains the record to new and accumulative value of hospitalization cases each week. For this dataset, there are several steps we took to process this data. First, we group the daily data into weekly format according to the date value so that it can be matched to the search trend dataset. Second, we kept the rows that present the hospitalization cases in each region of US and dropped the other rows that aren't relevant to our analysis. Third, we dropped the columns that are completely blank. After doing the seperate processing for each of the two datasets, we then merged them into a combined dataset based on the date

and the region code as indexes. Lastly, we dropped the rows of data that doesn't match into both of the datasets and the combined dataset is now ready for further analysis.

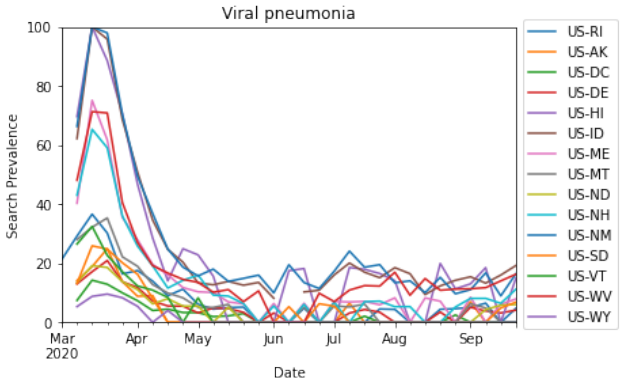

Figure 1: Distribution of the symptom: Viral Pneumonia.

Since the original symptom search trend dataset has already been normalized based on the maximum regional value, and we don't have access to the normalization factor, we decided to re-normalize the search trend data so that the data across different regions can be compared. Our strategy is to use the mean value of search trend across different regions as the baseline for scaling. This helps reduce the effect of having possible outlines as maximum on our models.

3 Results

3.1 Dimension Reduction

The PCA dimension reduction was applied on the dataset to produce a visualization of data in 2D. The following is the projections of the principal components in 2D with and without feature pruning.

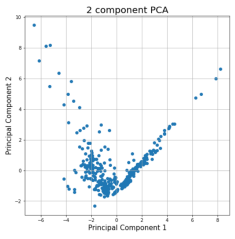

Figure 2: PCA with feature pruning in 2D.

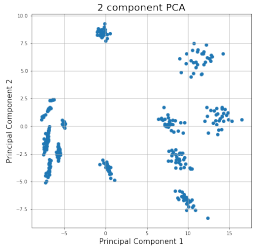

Figure 3: PCA without feature pruning in 2D.

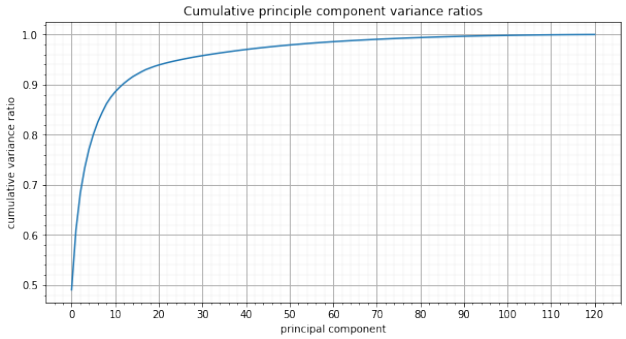

Figure 4: Cumulative explained variance of PCA over different numbers of components.

In our project, we proceeded with the non-pruning version. The upgrade to the pruning-first data was discussed and attempted to implement.

Additionally, the cumulative explained variances were examined over different number of Principal Components. 2-component PCA was selected as it yields relatively high variance.

3.2 Clustering

In order to gain an understanding on possible groupings in the search trends dataset, we performed K-means clustering on the raw data and the PCA-reduced data. Trying values from 1 to 20 for the number of clusters K, we observed 1) the remaining cost, 2) visualization of the PCA-reduced clustering, and 3) the degree of similarity between the raw and PCA-reduced clustering sets, determined using the Rand index. After performing the clustering, we found that in general increasing k would result in lower remaining cost, due to the larger number of cluster centers. As such, it was not a large consideration when choosing a value of K to analyze. Instead, we focused primarily on visualization of the clustering of the PCA-reduced data, and secondarily on Rand index so that the PCA-reduced clustering was more consistent with the raw data. Ultimately, the value of K we focused on was K=14, which resulted in both the highest Rand index (0.880) and the best visualization.

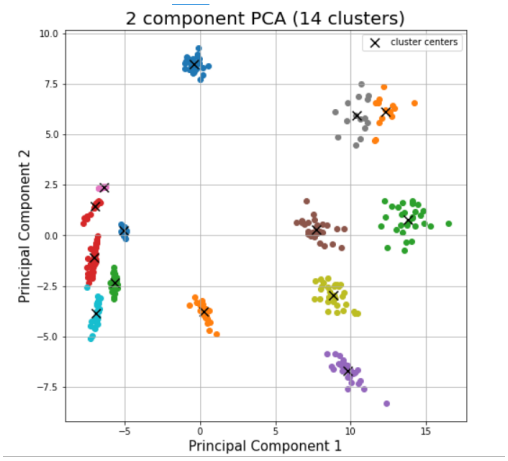

Figure 5: Clustering PCA-reduced data with 14 means

The relatively high Rand index indicated that despite only using 2 primary components, the dimension-reduced data still maintains the bulk of its trends. Additionally, the visualization appears with very distinct clustering, which indicates strong groupings. However, to explain this degree of grouping, we looked at the memberships of each cluster, considering the dates, regions, and means of symptoms (for raw data clustering). What we found was that for the most part, each cluster contained the data points from one or two regions. We speculate that region heavily affects data grouping because adjacent states might share infection rates, while some individual/groups of states provide sufficiently different conditions to result in strong symptom trend groupings. A notable example is the cluster at around (-7,

-1), which contains all of the data points from adjacent regions US-ME (Maine) and US-NH (New Hampshire).

3.3 KNN

For the region based data split, the result we get from our KNN model approximately ranges between 4000 and 6000. When we set the hyperparameter k = 1, we get the RMSE value of around 70.3 for new hospitalization cases and around 6331.7 for cumulative hospitalization cases. The RMSE value for new hospitalization cases then gradually decreases and reaches a low point at around k = 72, after which the RMSE value gradually increases. However, for the cumulative hospitalization cases, after k = 1, the RMSE value keeps decreasing and reaches a low point at around k = 80.

3.4 Decision Tree

The decision tree model is constructed using the "DecisionTreeClassifier" of the "sklearn" library. The models are trained on the scaled version of the original dataset. After training the model using k-fold cross validation, we have the following performance (predicting the new hospitalized number):

	Decision Tree (time)	Decision Tree (region)
Score	0.3523809523809524	0.24217941858391295

3.5 Comparison of Model Performance

In general, the Decision Tree models resulted in lower prediction accuracy. The reason behind such result is that the lack of pruning of symptoms in the training dataset. The agility of adjusting the hyper-parameters of KNN yields better results. The selection and pruning of highly variable symptoms observed in the PCA heat map was not fully implemented. The Decision Tree model is more sensitive to such bias, hence, we need to further explore the noise-reduction technique.

4 Discussion and Conclusion

4.1 Conclusion

From the above comparisons, we learned that only the with the appropriate combination of the learning model and the pre-processing of the data, including feature pruning of the search trend dataset would yield better results.

As we have shown above, splitting by time is better than by region and the KNN performs slightly better than the decision tree is due to the more flexibility in the hyper-parameters.

4.2 Future Improvements

1. Feature pruning.

   From our original dataset, we get around 122 symptom search trend data as features. In order to improve the model performance, we could use the results we get from the PCA and data clustering to prune the features. From our result, we selected eight features among the 122 symptoms. Using these as the input for model instead of having all 122 features as input could possibly improve the model performance to certain degree.

2. Integration with other data sets.

   Since the region based data has lower precision than time-based, other than the search trend and hospitalization datasets, we could integrate other relevant dataset in order to improve the performance of our model. For example, we could import the data for overall population that's prone to virus across different regions, which can be used to normalize the hospitalization data and to reduce the effect of different regions has on our model.

3. Time delay in new cases.

   Another thing we could possibly do in order to improve the model performance is to consider the time delay in new cases. For example, there could possibly be different period of time delay across different symptoms. We could perhaps import the data related to this aspect in order to make our model predict more accurately.

4. Normalizing the hospitalization data

   One of the main issues we have in our dataset is that our search trend dataset was normalized by unknown normalization factors. In this case, we couldn't easily compare the data across different regions and our models are prone to be affected by it. Aside from setting mean value as the normalization baseline, another possible solution would be to normalize the hospitalization data based on certain baseline as well. One approach we could take is to re-scale the hospitalization data based on the regional mean, which can help reduce the effect of regional difference has on the model performance. One thing worth noting is that this approach requires us to make an additional assumption that the normalization factor are similar for search trend and hospitalization data sets.

5 Statement of Contributions

All three of student partners worked together to understand the problem and write the code. Each individual contributed equally in this mini project.