COMP 551 - Group 58: Mini Project 1

Osman Warsi 260763921
Marwan Khan 260762251
Anthony Johansen 260712168

Abstract

In this project we investigated the performance of two regression models, namely k-nearest neighbours (KNN) and decision trees, on predicting COVID-19 hospitalization cases from related symptoms searched on google, We found that the k-nearest neighbour regression approach achieved better accuracy than decision trees on the data split based on region, and decision trees performed slightly better on the data split based on time. This is because the decision trees were overfitting as they had a significantly higher root mean squared error (RMSE) on the validation data as opposed to the training data. However, KNN was significantly slower to train due to its expensive real time execution. From the experimental results, it is shown that the weather variables are as relevant in predicting the new hospitalization cases when compared to the other variables found in our weekly symptom dataset. Thus, from this result, we can conclude that temperature and humidity are important features for predicting new hospitalization cases. Moreover, it is indicated that the higher the value of temperature the lower number of infection cases.

Introduction

We used three datasets, where in dataset_1 for each day, the count for the searches were mapped to each of the symptoms and organized the data by geographic region. The resulting dataset is a weekly time series for each region showing the relative frequency of searches for each symptom [1]. dataset_2 includes time series data for COVID-19 cases, deaths, tests, hospitalizations, discharges among other attributes [2]. We used weather data as our additional feature, it consists of weather and COVID-19 data in the date range of 22/1/2020 to 21/3/2020, obtained from Kaggle [3]. By using Pearson correlation matrix we were able to find that some of the symptoms in our dataset that were highly correlated to the new hospitalization cases had similar effects to the common COVID-19 symptoms according to google [4]. These symptoms include nasal polyp, allergic conjunctivitis, and hypersomnia. This suggests that regions with high frequency of these symptom searches had high hospitalization cases. From running the weather data on our KNN and decision tree models we were able to deduce that temperature enhances the performance of our models in predicting more accurate hospitalization cases, In addition to that there is a relatively strong negative correlation with temperature and new hospitalization cases in the Pearson correlation matrix. This is backed up from the experimental results shown in Table 2 of this PMC paper, they have proved that climatic conditions such as temperature and humidity contribute to the spread of the virus. Based on the results of the models, one can draw a conclusion that when the temperature is low and humidity is also low, infection rate increases. On the other hand, when both temperature and humidity are high, the infection rate of COVID-19 decreases. [5]. This suggests that regions with lower temperature have a greater spread of COVID-19 cases, hence increasing the number of new hospitalization cases.

Datasets

We started by creating a dataframe for each of the datasets using pandas and loaded data relevant to the USA. Only the hospitalized_new column was used from dataset_2, dataset_1 was cleaned by setting a threshold of 25 non-NaN values in the rows and a threshold of 255 in the columns. This was done to reduce noise and keep data that is less sparse, resulting in lower (RMSE) for both KNN and decision trees. The time resolution of all the datasets was then matched to weekly; we noticed that some of the most common symptoms of COVID-19 were not present in the weekly symptoms search data, but were present in the daily symptoms search data, including fever, common cold, cough and shortness of breath. We did not merge these symptoms into our weekly dataset as the values could not be normalized. We merged dataset_1 and dataset_2 into the same dataset, titled dataset_3 to display the weekly emerging hospitalization cases.

For Task 2.1 we started by identifying the symptoms with the most data in our dataset. Due to the datasets missing large amounts of data for certain regions and symptoms we decided to create a new dataset containing symptoms with minimal missing data. We then normalized this new data in order to be able to compare data across different regions as the data we imported was region specific. The data was normalized using the following procedure.

- Store the values of the column as floats in a variable
- Create a maximum processor object (from sklearn)
- Create an object to transform the data to fit in the processor

Once the data was normalized we were able to compare search frequencies across different regions for our 3 top symptoms: Shallow Breathing, Ventricular Fibrillation, and Aphonia. The data plots are located in the appendix. Figure 6.1 and 6.2 are a comparison between using the normalized and non-normalized data. We can immediately tell there is a large difference between the two, and the normalized plot shows how in fact the search trends are a lot more similar between the regions than the unnormalized data would lead us to believe.

Interestingly enough we notice a common downwards trend for the Shallow Breathing and Aphonia symptoms as the weeks progress which could be attributed to the decrease in COVID cases during this time. However, the frequency of the Ventricular Fibrillation search stays very varied over the 30 weeks. We would have expected this symptom to follow the general trend and reduce in frequency over time, however it is possible that the search frequency stayed high due to reasons beyond our scope.

The merged dataset_3 was further converted to numpy arrays for training. We one hot encoded the region and the date column values as we needed numeric values for training. The data was then imputed to resolve the NaN values. The dataset was thereafter split using two strategies, the first being the split based on region and the second being the split based on time.

Pearson correlation matrix was used for feature filtering by removing symptoms that had minimal correlation with the dependent variable hospitalize_new. From observing the correlation matrix, we obtained these values: 1 implies stronger positive correlation, -1 implies stronger negative correlation and 0 implies no correlation. Below are the symptoms with the greatest impact on emerging hospitalization cases, possessing a correlation of over ±0.1.

symptom:Allergic conjunctivitis	0.137456
symptom:Crepitus	0.397839
symptom:Depersonalization	0.413924
symptom:Epiphora	0.297734
symptom:Gingival recession	0.117662
symptom:Hyperemesis gravidarum	0.126766
symptom:Hypersomnia	0.108848
symptom:Hypomania	-0.116910
symptom:Nasal polyp	0.307346
symptom:Night terror	0.120250
symptom:Rumination	0.165863
symptom:Viral pneumonia	-0.147416

We can observe nasal polyp, allergic conjunctivitis, and hypersomnia, which demonstrate a relatively strong positive correlation with the new hospitalization cases; they provide similar effects to the actual COVID-19 symptoms. One of the most common symptoms according to google's COVID-19 alert page is tiredness, which is related to hypersomnia, another serious symptom is difficulty breathing, which is also a symptom of nasal polyp. One of the fewer occurring symptoms is conjunctivitis, represented by allergic conjunctivitis in our dataset. Using this correlation matrix we removed features with the lowest correlation (value close to 0) with hospitalized_new to reduce noise and enhance accuracy.

Weather data was included as an extra feature [3]. We experimented by merging the temperature, humidity and wind speed column into our previously merged dataset_3, it was observed that temperature had the strongest relative negative correlation to the new hospitalization cases. Hence, we merged it into dataset_3 and discarded the other weather dataset features. The strong negative correlation tells us that in regions where the average weekly temperature was low resulted in higher hospitalization cases. This indicates that increased temperature results in lower infection counts as mentioned in the PMC article [5]. In conclusion, the temperature feature improved the performance of these models by reducing the RMSE of KNN by 2 units and RMSE of decision tree by 11 units (Fig 3.3 & 3.4 in the appendix).

Results

A Visualization of the Search Trends Dataset in Lower Dimensions

When reducing our search trends dataset to a lower dimension we first wanted to see how many principal components we should keep. Through the trial of several components we came to the conclusion that in order to retain ~95% of the variance we would need at least 10 principal components, as can be observed in our plot located in the appendix under figure 7.1. Once we had the number of principal components required we performed PCA on the data by fitting for 10 components and performing a transform, the result of which was stored in a dataset. Now that the reduced dimension dataset was created we were able to visualize the search trends dataset in a lower dimension, the results of which are located in the appendix under figure 7.2.

The results of Clustering on Raw vs PCA Reduced Data

We used the K-means clustering method to group the search trends data. The Elbow method was used to determine the ideal value of the numbers of clusters k, from observing the plots in Fig 5.1 & 5.2 (in the appendix) we chose 15 as our value for the raw data and 10 for the PCA reduced data. K-means++ was used for centroid initialization as an attempt to push the centroids as far from one another as possible, covering as much of the occupied data space as they can from initialization.

It can be observed from the k-means cluster plots in Fig 5.3 & 5.4 (in the appendix) that there are fewer and more clear clusters for the PCA reduced data as opposed to the raw data.

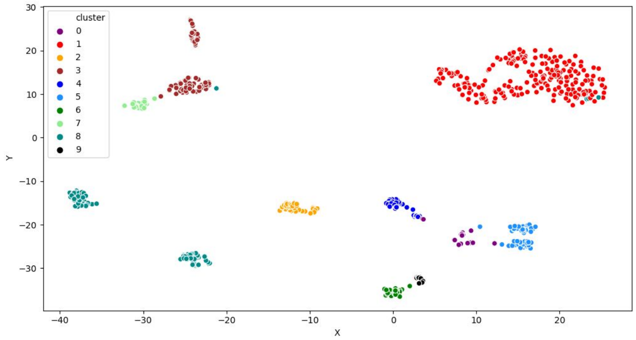


Fig 5.3: K-Means clustering of raw search trends data

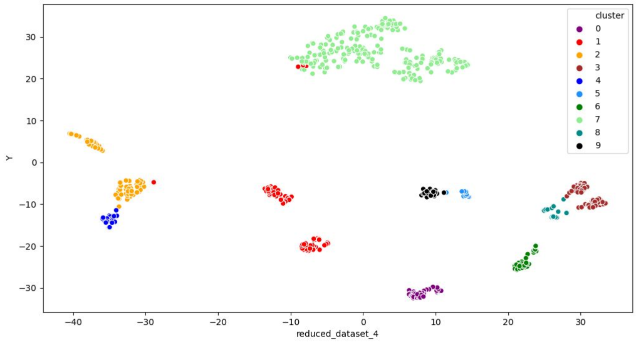


Fig 5.4: K-Means clustering of PCA reduced search trends data

K-Nearest Neighbors vs Decision Trees Performance

5-fold cross validation was used and repeated 10 times in our dataset to reach a more precise estimate of the cross validation accuracy. Fig. 4.1 shows how the cross validation accuracy changes with the value of k. It is observed that k=6 results in the best cross validation accuracy.

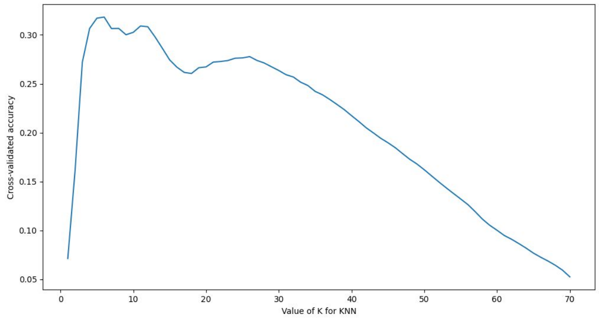
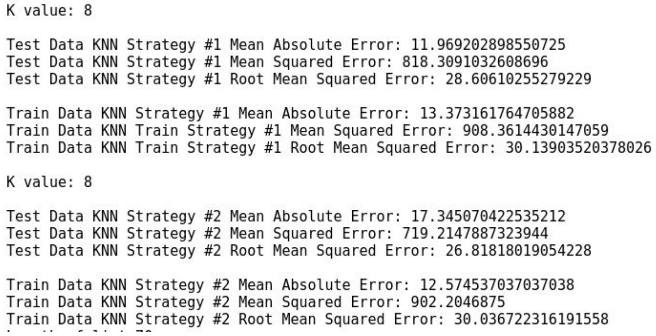


Fig 4.1: How the cross validation accuracy changes with different k values for KNN

Two KNN models were created by importing KNeighborsRegressor from sklearn library, where knn_1 was used to train the data split using the first strategy and knn_2 was used for the second strategy. The metric was set as minkowski and p as 2, equivalent to the standard Euclidean metric. Mean squared error (MSE) was plotted against k values for both splits to determine the optimal value of k where the least MSE would be obtained and the model is not overfitting, using the training and validation data. In the case of knn_2, upon consulting Fig 1.2, 8 was deducted to be the ideal value of k as it gives a low MSE on the validation and training data, and is not as likely to be overfitting. In the case of knn_1 k was chosen to be 8, Fig 5.1 shows that k=8 still gives a relatively high validation accuracy, Fig 1.1 (in the appendix) shows that it does not overfit.

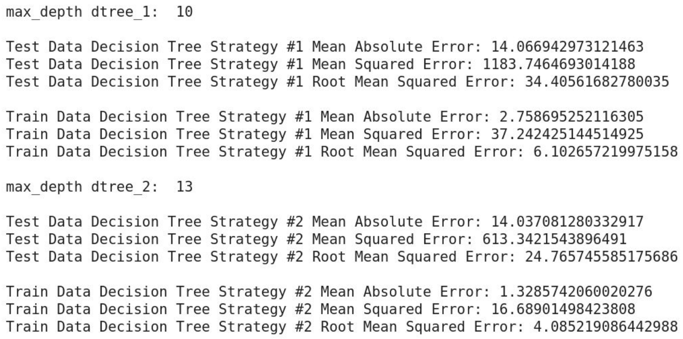
Two decision tree models were created by importing DecisionTreeRegressor from sklearn, where dtree_1 was used to train the data split using strategy 1 and dtree_2 was used for strategy 2. We created plots of mean squared error (MSE) against max_depth values for both splits to determine the optimal value of the depth of the tree where we get the least MSE, using the training and validation data. From fig 2.1 (in the appendix), 10 was determined as the optimal value of max_depth for split 1, it produces the least MSE on the validation data. From fig 2.2 (in the appendix), 13 was determined to be the optimal value of max_depth for split 1, as it also produces the least MSE on the validation data.

For the split based on region (strategy 1) it can be observed that the RMSE on the test data of KNN is much lower than that of the decision tree (Fig 3.1 & 3,2). This can be explained by observing the RMSE results on the training data of both models, where the RMSE on the training data for the decision tree is significantly lower than the RMSE on the test data, suggesting overfitting, causing RMSE on the unseen data to be higher. For split based on time (strategy 2) the RMSE of the decision tree is observed to be slightly lower than that of KNN, the possibility of the decision tree overfitting still persists.



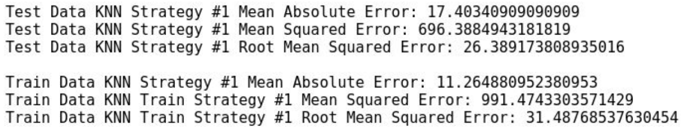
```
K value: 8

Test Data KNN Strategy #1 Mean Absolute Error: 11.969202898550725
Test Data KNN Strategy #1 Mean Squared Error: 818.3091032608696
Test Data KNN Strategy #1 Root Mean Squared Error: 28.60610255279229

Train Data KNN Strategy #1 Mean Absolute Error: 13.373161764705882
Train Data KNN Train Strategy #1 Mean Squared Error: 908.3614430147059
Train Data KNN Train Strategy #1 Root Mean Squared Error: 30.13903520378026

K value: 8

Test Data KNN Strategy #2 Mean Absolute Error: 17.345070422535212
Test Data KNN Strategy #2 Mean Squared Error: 719.2147887323944
Test Data KNN Strategy #2 Root Mean Squared Error: 26.81818019054228

Train Data KNN Strategy #2 Mean Absolute Error: 12.574537037037038
Train Data KNN Strategy #2 Mean Squared Error: 902.2046875
Train Data KNN Strategy #2 Root Mean Squared Error: 30.036722316191558
```

Fig. 3.1: RMSE of K nearest neighbors on training and validation data



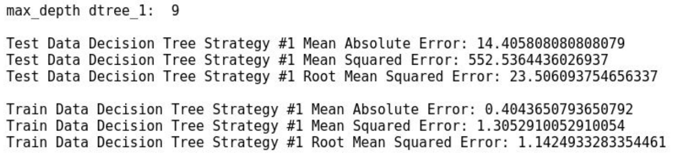
```
max_depth dtree_1:  10

Test Data Decision Tree Strategy #1 Mean Absolute Error: 14.066942973121463
Test Data Decision Tree Strategy #1 Mean Squared Error: 1183.7464693014188
Test Data Decision Tree Strategy #1 Root Mean Squared Error: 34.40561682780035

Train Data Decision Tree Strategy #1 Mean Absolute Error: 2.758695252116305
Train Data Decision Tree Strategy #1 Mean Squared Error: 37.242425144514925
Train Data Decision Tree Strategy #1 Root Mean Squared Error: 6.102657219975158

max_depth dtree_2:  13

Test Data Decision Tree Strategy #2 Mean Absolute Error: 14.037081280332917
Test Data Decision Tree Strategy #2 Mean Squared Error: 613.3421543896491
Test Data Decision Tree Strategy #2 Root Mean Squared Error: 24.765745585175686

Train Data Decision Tree Strategy #2 Mean Absolute Error: 1.3285742060020276
Train Data Decision Tree Strategy #2 Mean Squared Error: 16.68901498423808
Train Data Decision Tree Strategy #2 Root Mean Squared Error: 4.085219086442988
```

Fig. 3.2: RMSE of decision trees on training and validation data



```
Test Data KNN Strategy #1 Mean Absolute Error: 17.40340909090909
Test Data KNN Strategy #1 Mean Squared Error: 696.3884943181819
Test Data KNN Strategy #1 Root Mean Squared Error: 26.389173808935016

Train Data KNN Strategy #1 Mean Absolute Error: 11.264880952380953
Train Data KNN Train Strategy #1 Mean Squared Error: 991.4743303571429
Train Data KNN Train Strategy #1 Root Mean Squared Error: 31.48768537630454
```

Fig 3.3: RMSE of KNN with temperature feature on training and validation data



```
max_depth dtree_1:  9

Test Data Decision Tree Strategy #1 Mean Absolute Error: 14.405808080808079
Test Data Decision Tree Strategy #1 Mean Squared Error: 552.5364436026937
Test Data Decision Tree Strategy #1 Root Mean Squared Error: 23.506093754656337

Train Data Decision Tree Strategy #1 Mean Absolute Error: 0.4043650793650792
Train Data Decision Tree Strategy #1 Mean Squared Error: 1.3052910052910054
Train Data Decision Tree Strategy #1 Root Mean Squared Error: 1.1424933283354461
```

Fig 3.4: RMSE of decision trees with temperature feature on training and validation data

Discussion and Conclusion

In this project we learned that regions with the highest frequency of COVID like symptoms searches had the higher hospitalization cases compared to other regions. These symptoms include nasal polyp, allergic conjunctivitis, and hypersomnia. The frequency of these symptoms increased with time, along with the hospitalization cases. We also learned that including temperature into our dataset improved the performance of our model on the validation data, and suggested a strong negative correlation with the new hospitalization cases. In our future work, we aim to look at how to improve the performance of our models by considering additional weather features such as wind speed and rainfall. We are also planning to update this study with more analyses and cases continuously by fine-tuning the prediction and visualization methodology, perhaps by using different regression models or neural networks. Experimenting on these new features will hopefully help us in the fight against coronavirus and decreasing the number of cases.

Statement of Contributions

Marwan: Task 1 (data preprocessing), Task 3 (supervised learning)
Anthony: Task 2.1(Visualization), Task 2.2(PCA)
Osman: Task 2.3 (K means clustering)

Appendix

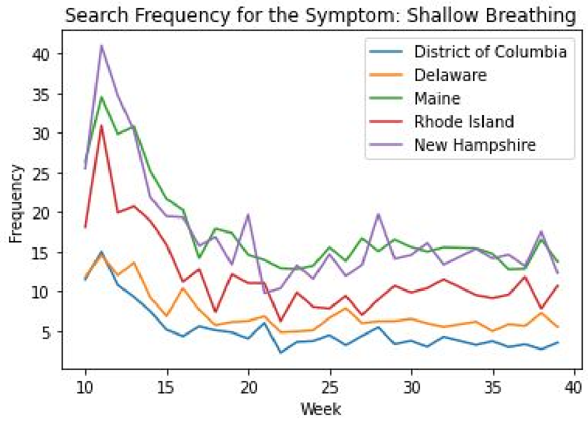


Figure 6.1: Search Frequency of Shallow Breathing(Pre-Normalization)

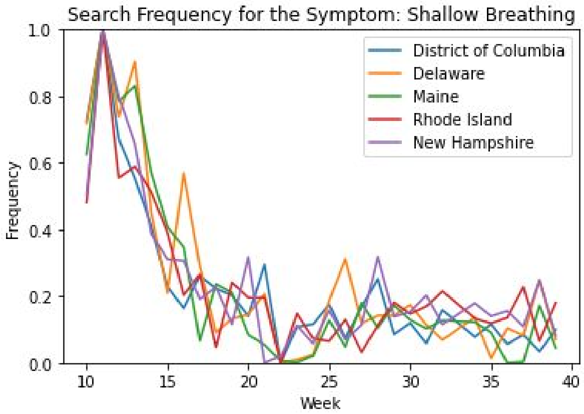


Figure 6.2: Search Frequency of Shallow Breathing (Post-Normalization)

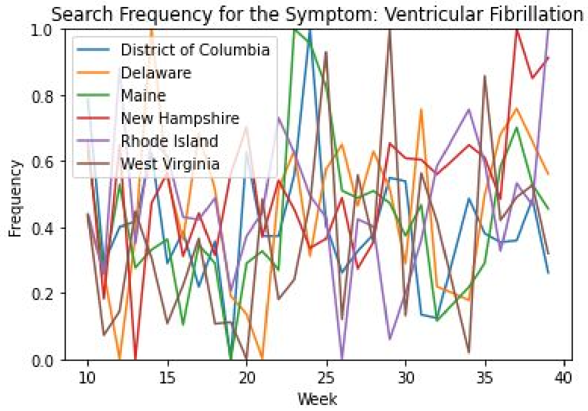


Figure 6.3: Search Frequency of Ventricular Fibrillation

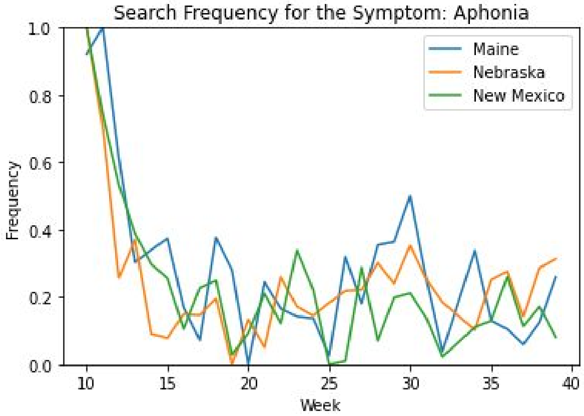


Figure 6.4: Search Frequency of Aphonia

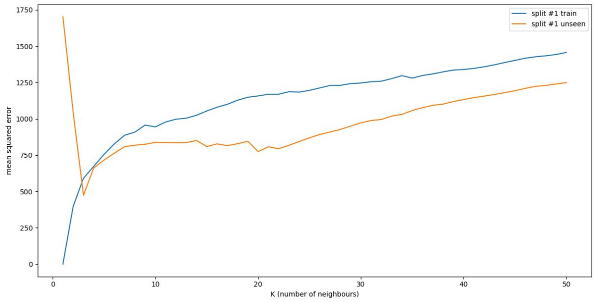


Fig 1.1: Plot of MSE against K neighbors with split based on region (strategy #1)

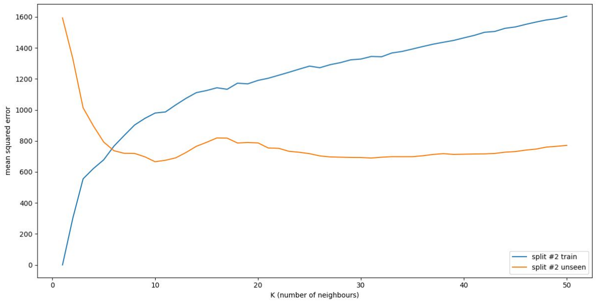
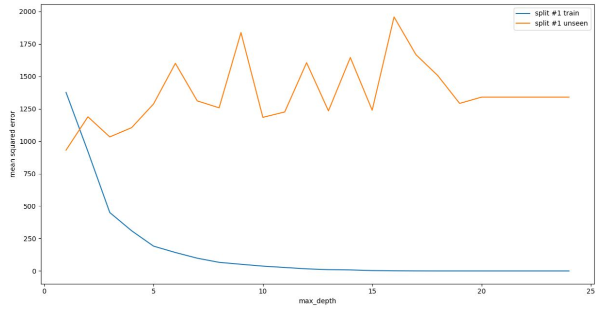
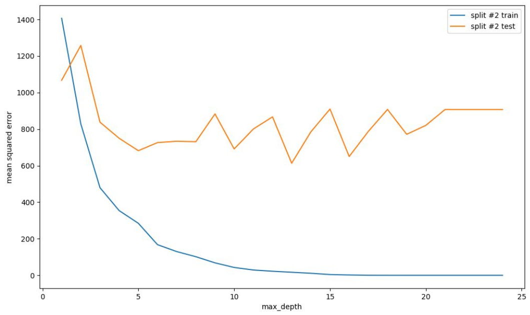
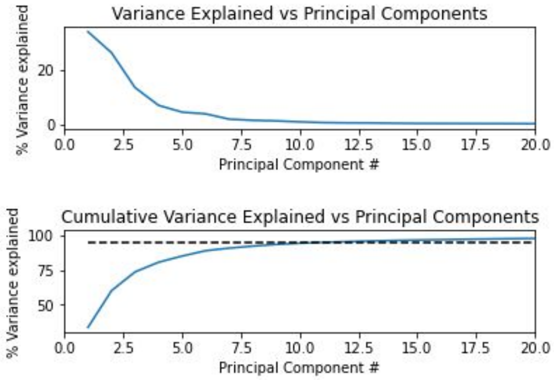
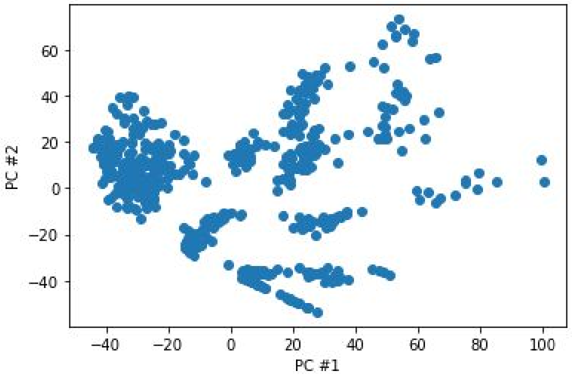
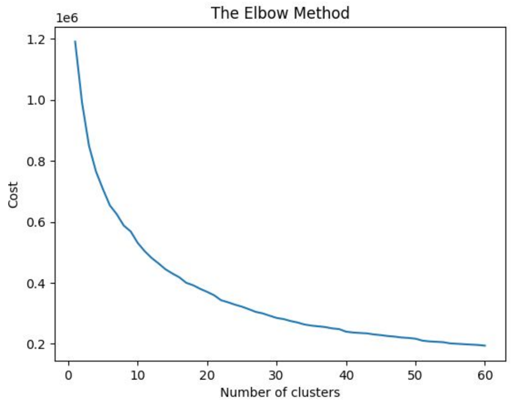
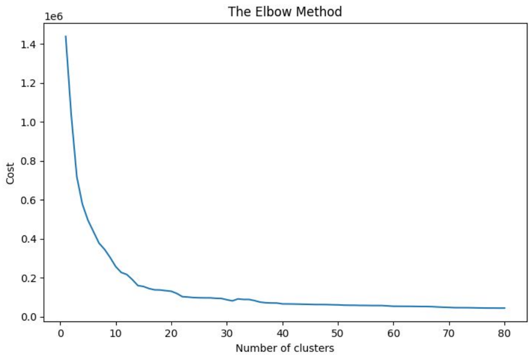


Fig 1.2: Plot of MSE against K neighbors with split based on time(strategy #2)

 Fig 2.1: Plot of MSE against max_depth values with split based on region (strategy #1)	 Fig 2.2: Plot of MSE against max_depth values with split based on time (strategy #2)
 Figure 7.1: Visualization of the Explained Variance vs Principal Components	 Fig 7.2 : Plot of Principal Component 1 vs Principal Component 2
 Fig 5.1: Using Elbow method to find ideal value of k clusters for the raw data	 Fig 5.2: Using Elbow method to find ideal value of k clusters for the PCA reduced data

References:

[1] Google LLC "Google COVID-19 Search Trends symptoms dataset". http://goo.gle/covid19symptomdataset , Accessed: 08-10-2020

[2] Creative Commons Attribution 4.0 International. https://github.com/google-research/open-covid-19-data/tree/master/data/exports/cc_by, Accessed: 08-10-2020

[3] Kaggle covid19 global weather data. Kaggle. 2020 [Google Scholar] https://www.kaggle.com/winterpierre91/covid19-global-weather-data, Accessed: 12-10-2020

[4] Google COVID-19 Alert. https://www.google.com/search?q=covid+19+symptoms&oq=covid+19sy&aqs=chrome.1.69i57j0i10i457j0i10l6.5510j0j7&sourceid=chrome&ie=UTF-8, Accessed: 10-10-2020

[5] Zohair Malki,a El-Sayed Atlam. Association between weather data and COVID-19 pandemic predicting mortality rate: Machine learning approaches. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7367008/#bib0010, Accessed: 11-10-2020