Steps From Data Gathering To Unique Clustering Algorithm

**Data Gathering**

1. Retrieve mental health data from CDC API

2. Partition dataset to only include New York based entries

3. Retrieve NYC Covid-19 data from Github repository

4. Merge CDC data and NYC data together based on “date” column

5. Check for dataset class imbalance (does not exist)

**Initial ML Approach**

1. Explore conventional machine learning algorithms (logistical regression, linear regression, support vector machine, decision tree, k-nearest neighbors, gaussian naïve bates, random forests) and confirm accuracies through 10-fold cross validation:

LogReg| Mean=0.757098 STD=0.128332

SVM| Mean=0.831768 STD=0.115647

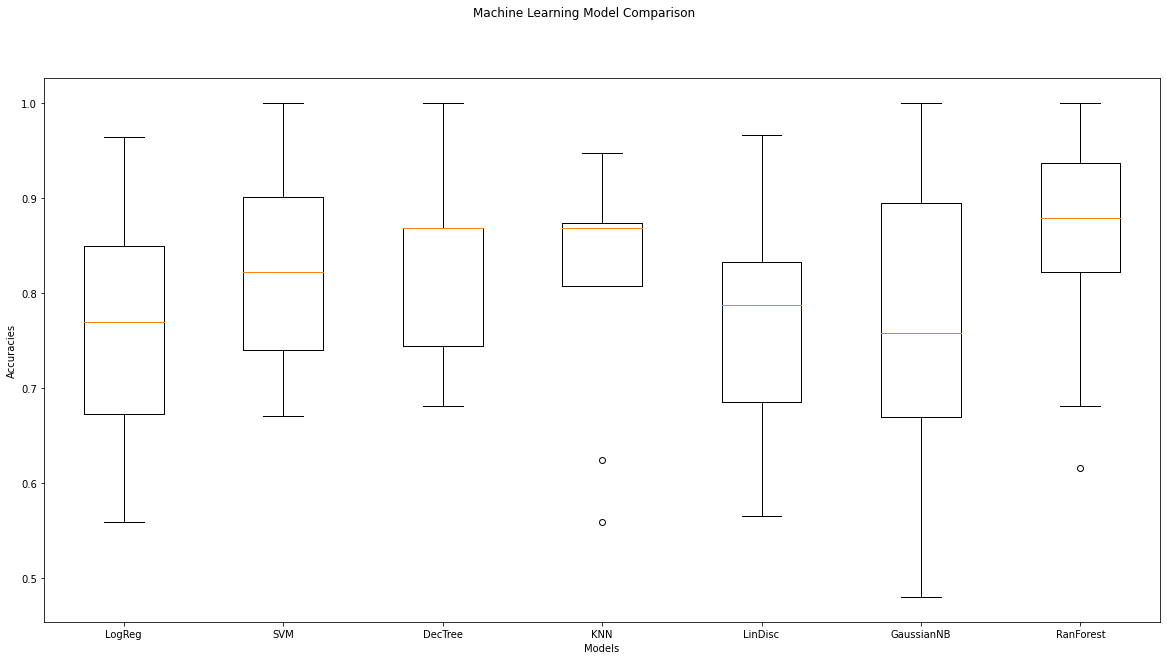
DecTree| Mean=0.829812 STD=0.095345

KNN| Mean=0.811424 STD=0.130377

LinDisc| Mean=0.763470 STD=0.119177

GaussianNB| Mean=0.765258 STD=0.175464

RanForest| Mean=0.850156 STD=0.124884



Random Forest had the highest accuracy at 85%.

2. Split dataset into training and testing sets with a 70:30 ratio.

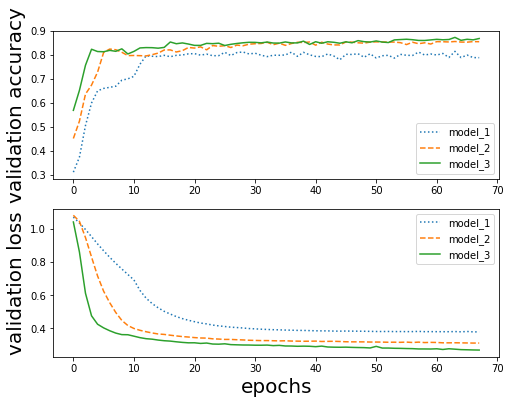
3. Created 3 different neural network models.

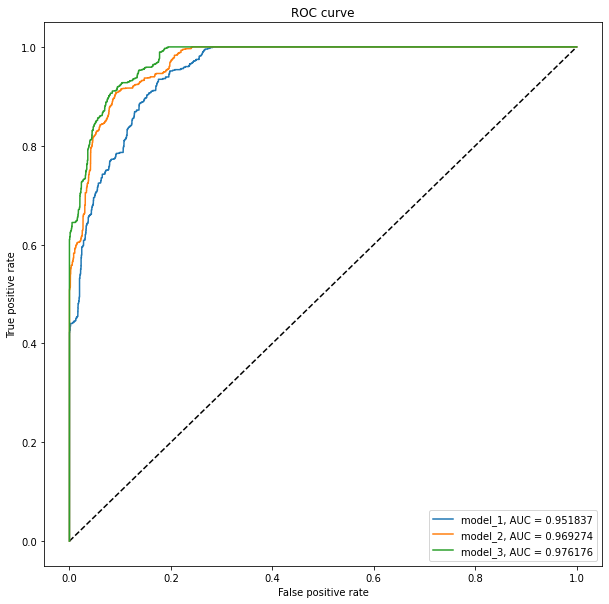
Model 1: 2 layers (input layer 8, output layer 3)

Model 2: 3 layers (input layer 8, hidden layer 8, output layer 3)

Model 3: 4 layers (input layer 8, 2 hidden layers 8, output layer 3)

|  |  |  |
| --- | --- | --- |
| Model | Loss | Accuracy |
| 1 | 0.37815380096435547 | 0.7872792482376099 |
| 2 | 0.3107653558254242 | 0.8539570569992065 |
| 3 | 0.2683994174003601 | 0.8672590851783752 |



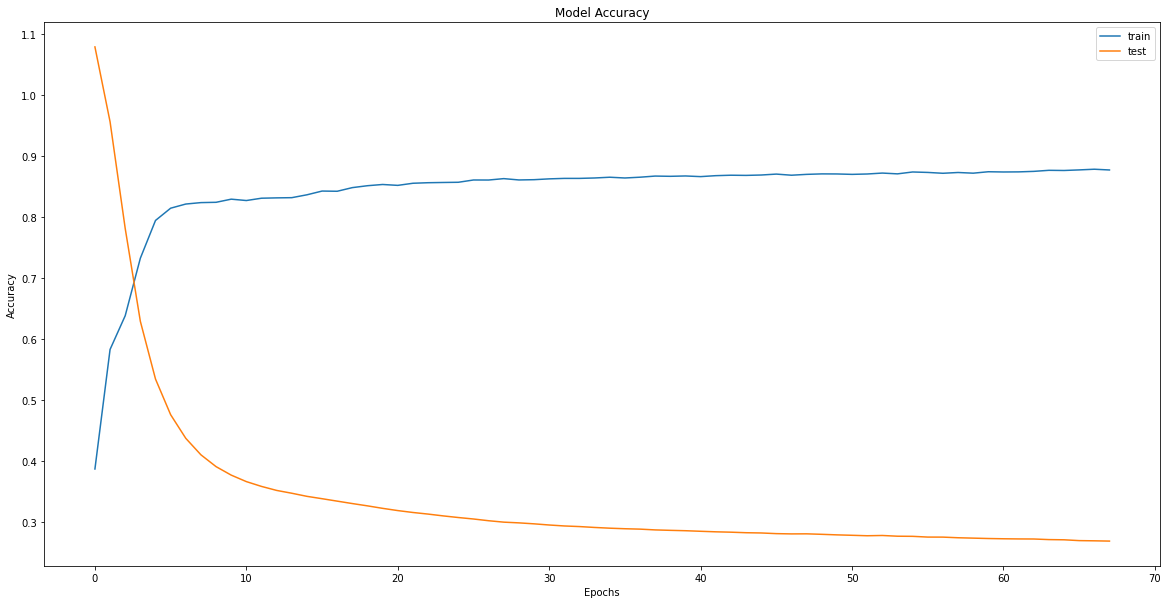


4. Implemented Kera Classifier using 10-fold cross validation,

Accuracy: 0.88 (+/- 0.02)

5. Created a self-defined ANN.

accuracy: 86.90%



precision recall f1-score support

0 0.86 0.72 0.79 6007

1 0.95 0.95 0.95 5961

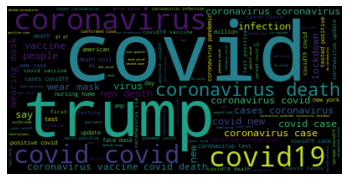
2 0.80 0.93 0.86 5924

**Added Twitter Component**

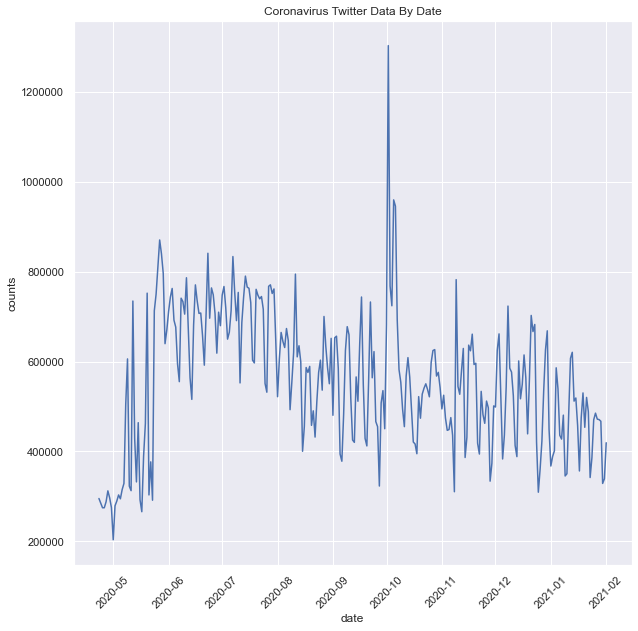
Need to reference: <https://raw.githubusercontent.com/thepanacealab/covid19_twitter/master/>

1. For each day, we generated the top 550 most tweeted words, phrases and hashtags along with their frequencies.

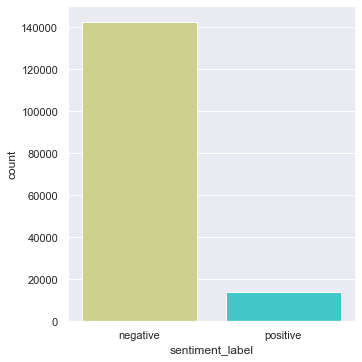
2. Generated a word cloud for the dataset.



3. Performed a simple time series analysis on the Twitter data.



4. Used NLTK Sentiment Analyzer to determine polarity of dataset. Distribution shown below:



5. Merge twitter dataset with the original data from CDC and NYC based on the date column.

**Results from adding Twitter data**

LogReg| Mean=0.804866 STD=0.122419

SVM| Mean=0.765962 STD=0.130235

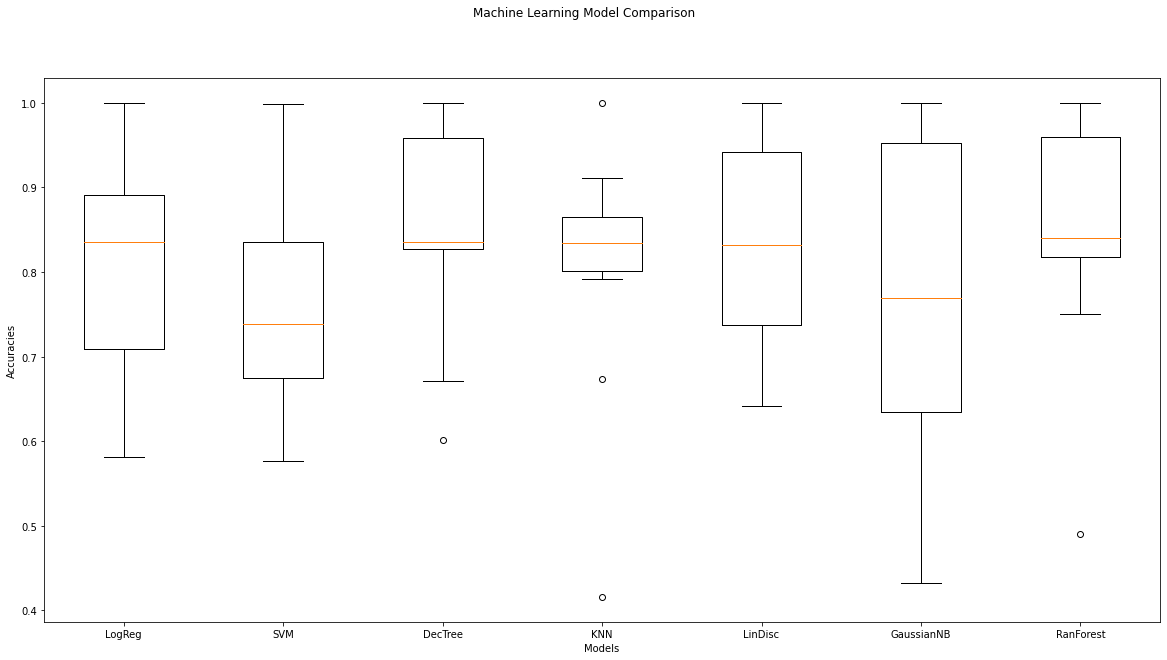
DecTree| Mean=0.843900 STD=0.127495

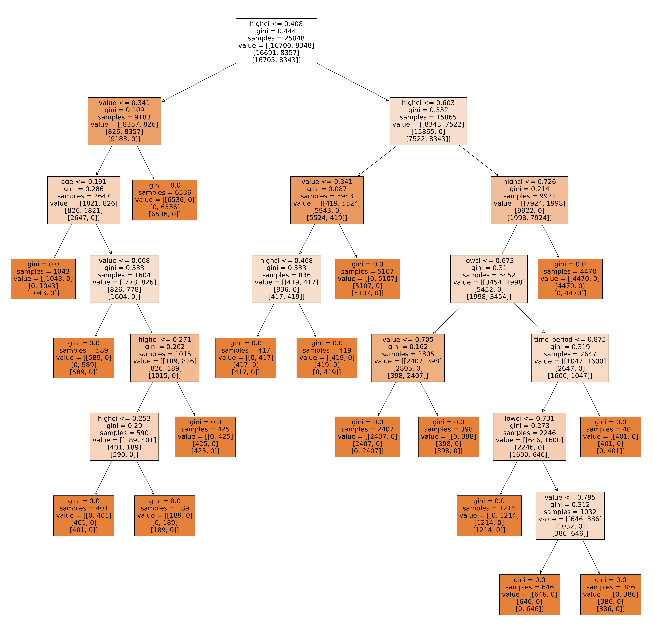
KNN| Mean=0.800781 STD=0.150575

LinDisc| Mean=0.835157 STD=0.114623

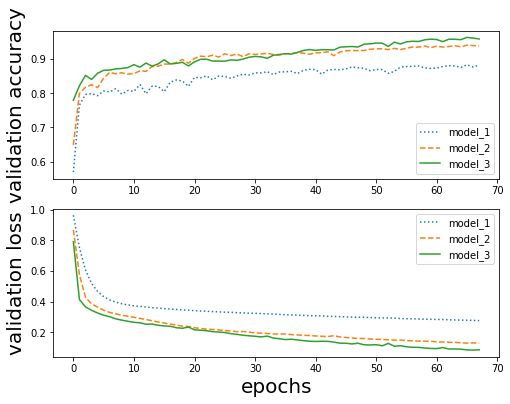
GaussianNB| Mean=0.765278 STD=0.187195

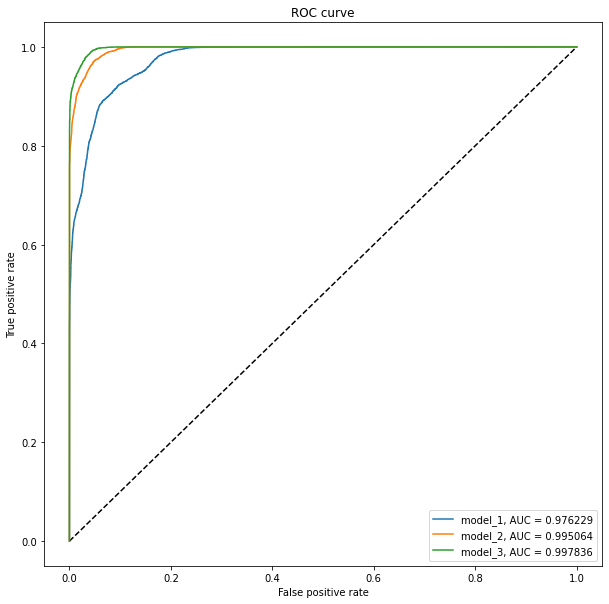
RanForest| Mean=0.840914 STD=0.143776





|  |  |  |
| --- | --- | --- |
| Model | Loss | Accuracy |
| 1 | 0.27748996019363403 | 0.8817064166069031 |
| 2 | 0.13010241091251373 | 0.937779426574707 |
| 3 | 0.08703181892633438 | 0.9580849409103394 |



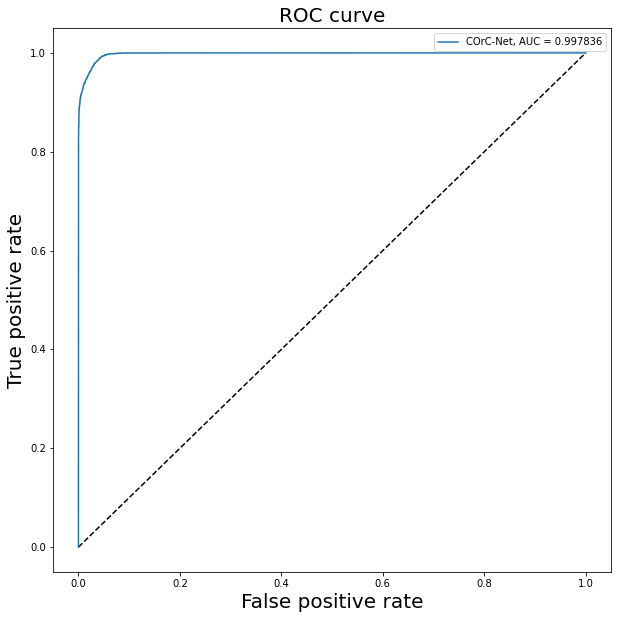


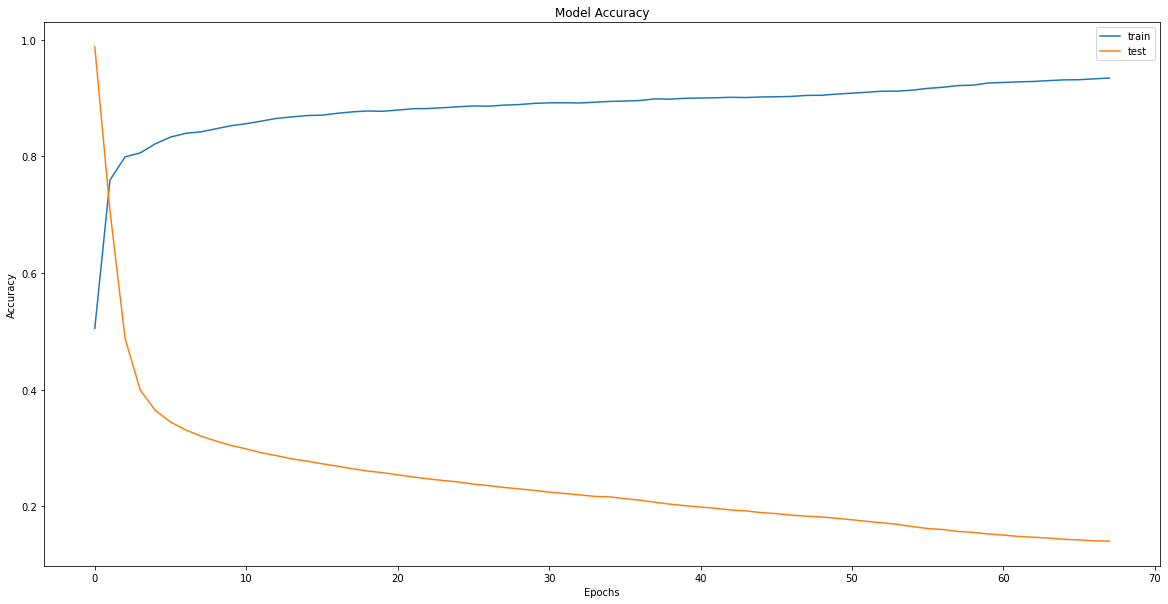
Keras Classifier 10-fold cross validation:

Accuracy : 0.96 (+/- 0.02)

Self-defined ANN:

accuracy: 92.87%





precision recall f1-score support

0 0.95 0.92 0.94 3580

1 0.99 0.98 0.99 3571

2 0.93 0.98 0.95 3585

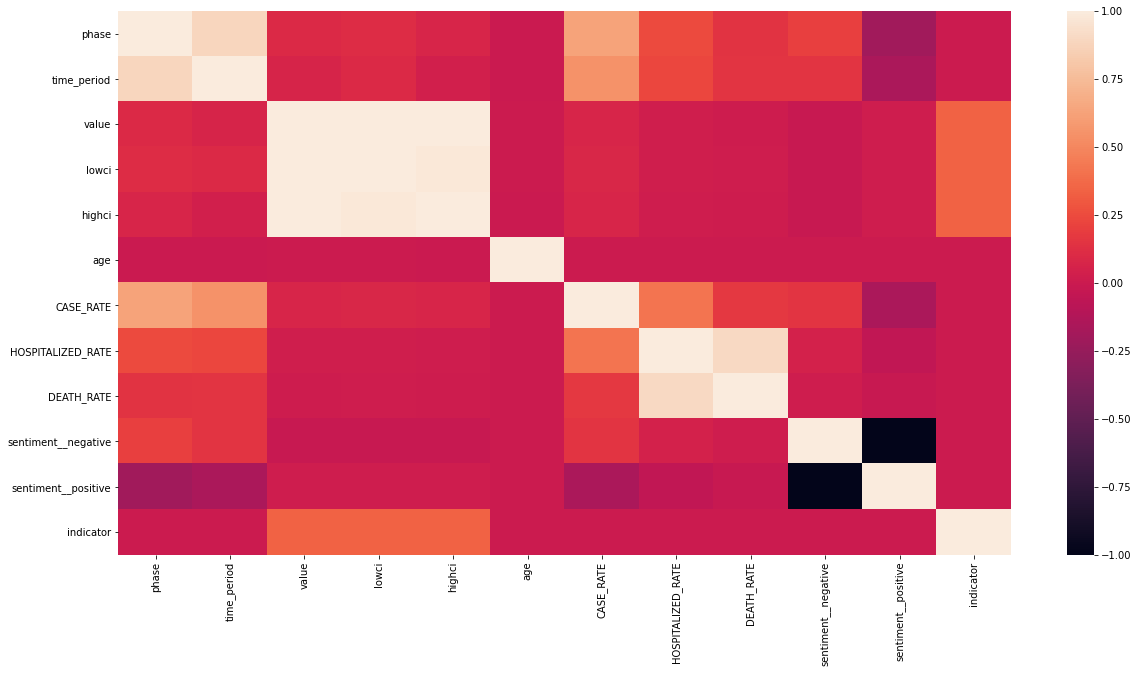
**Statistical Significance Testing**

**Chi-Squared Results:**

|  |  |  |  |
| --- | --- | --- | --- |
| Chi Value | Degrees of Freedom | Critical Value | p-value |
| 47175933.879299 | 393613 | 395073.544980 | 0.0293676150 |

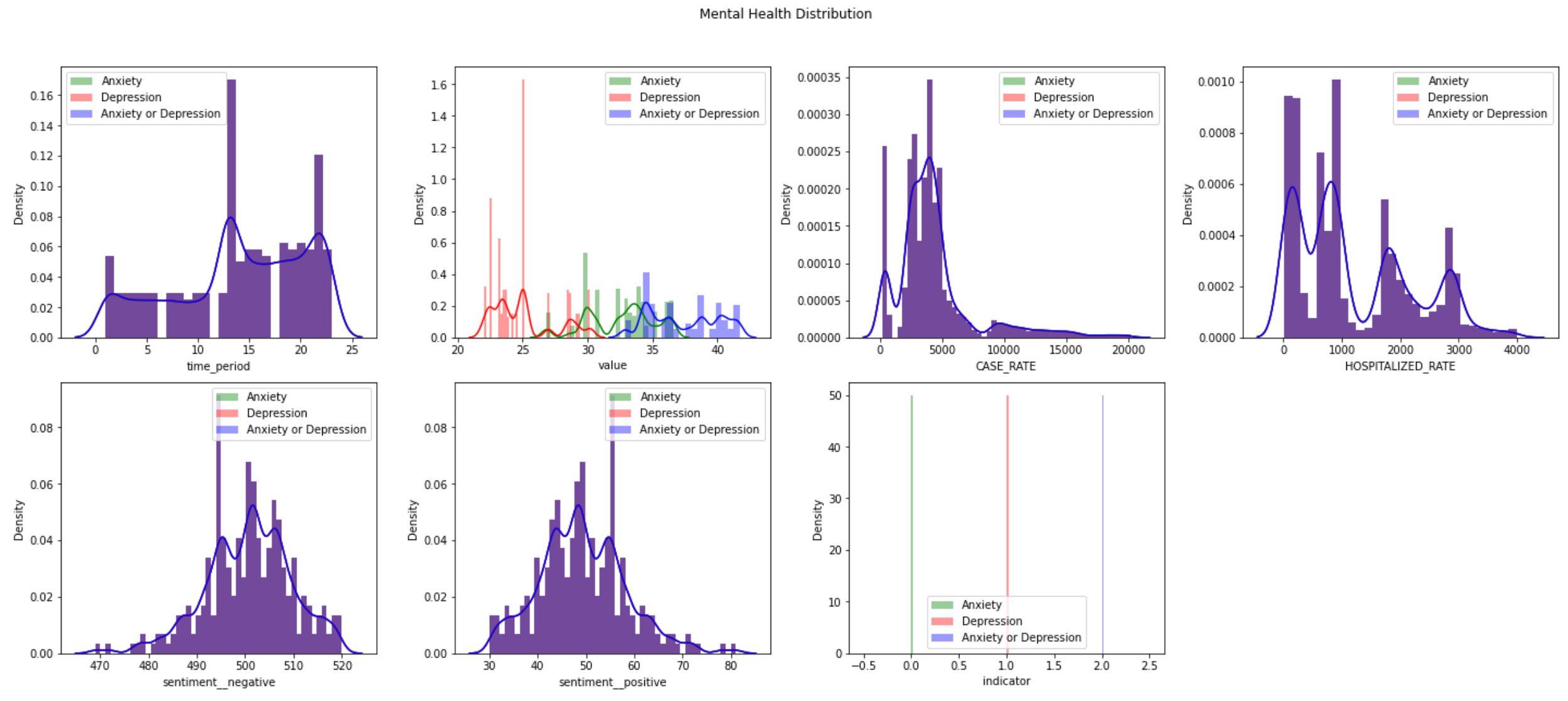
Dataset can be considered statistically significant since its p-value is less than 0.05

**Pearson Correlation Test:**

****

Based on the Pearson Correlation: the significant columns are: **time\_period, value, case\_rate, hospitalized rate, sentiment\_negative, sentiment\_positive, indicator**

Mental Health Distribution Based On Indicator Field:



Shapiro-Wilkes Test

Statistics=0.390, p=0.000

Sample looks Gaussian

One-Way ANOVA Test

F-Statistic=53726.238, p=0.000

Reject Null Hypothesis and Accept Alternate Hypothesis

**Supervised Learning On Statistically Significant Features**

LogReg| Mean=0.749707 STD=0.141091

SVM| Mean=0.804002 STD=0.160931

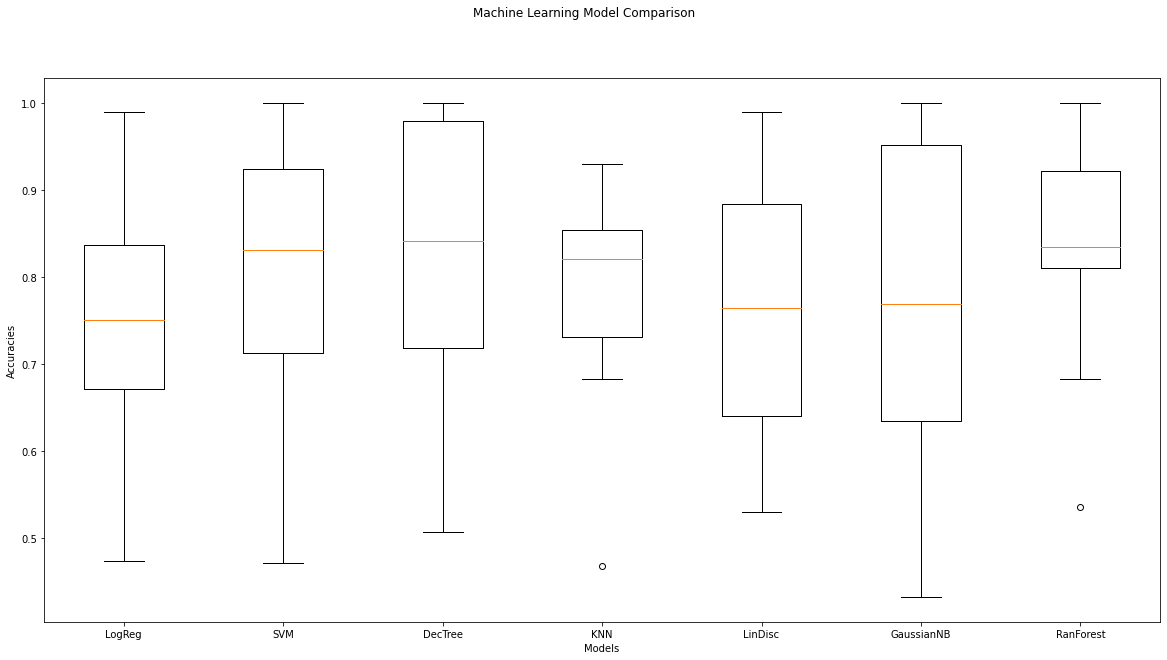
DecTree| Mean=0.828635 STD=0.156796

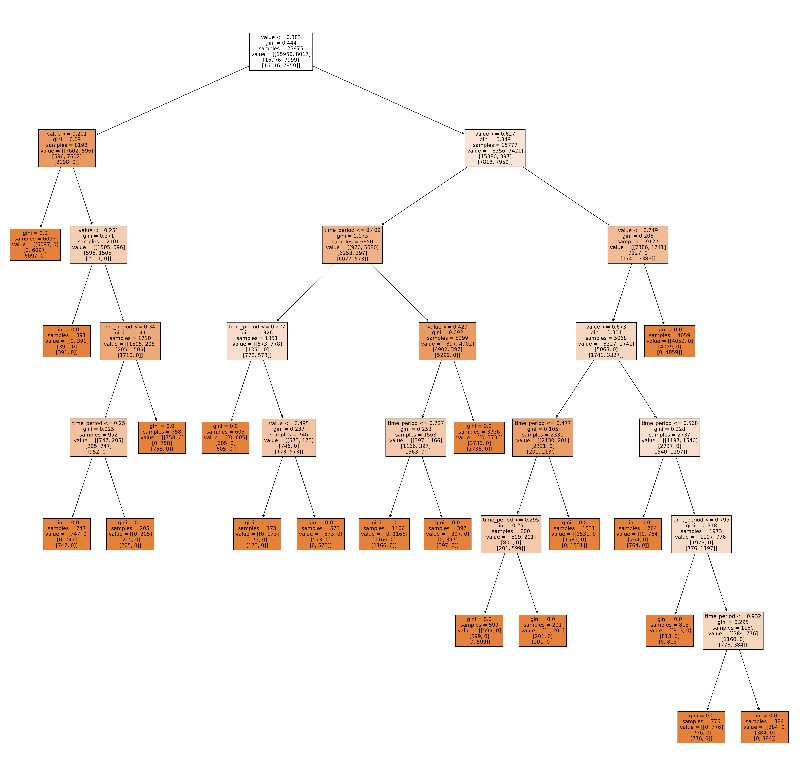
KNN| Mean=0.779827 STD=0.127464

LinDisc| Mean=0.760999 STD=0.142302

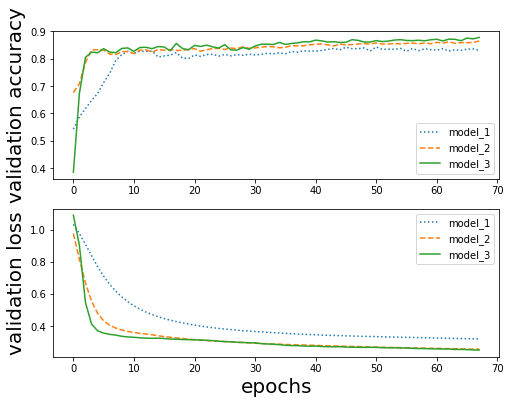
GaussianNB| Mean=0.765278 STD=0.187195

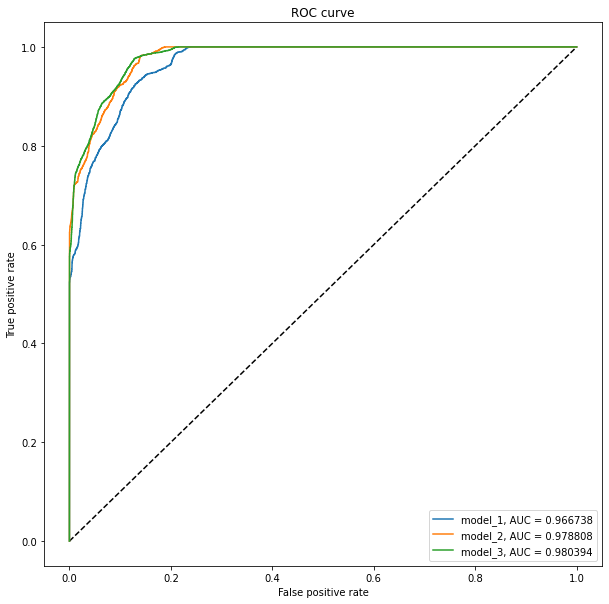
RanForest| Mean=0.831382 STD=0.130121





|  |  |  |
| --- | --- | --- |
| Model | Loss | Accuracy |
| 1 | 0.3207794040938681 | 0.8311457363028198 |
| 2 | 0.2585969111898002 | 0.8653569311542044 |
| 3 | 0.25180282072841653 | 0.8783978321619104 |



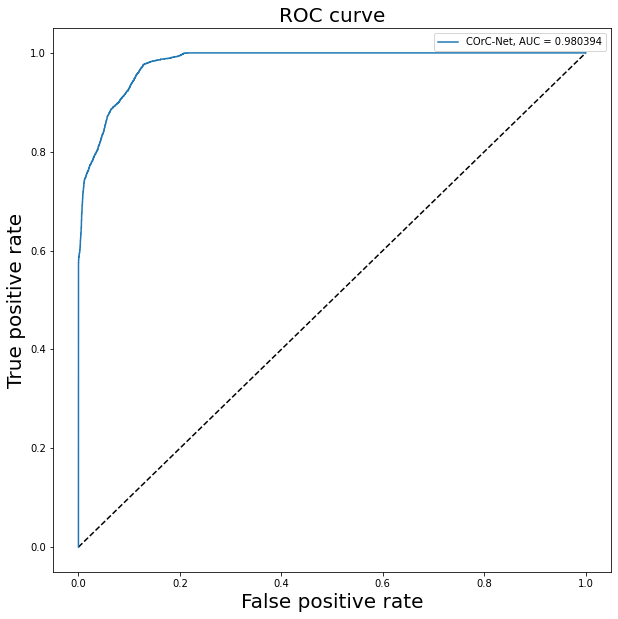


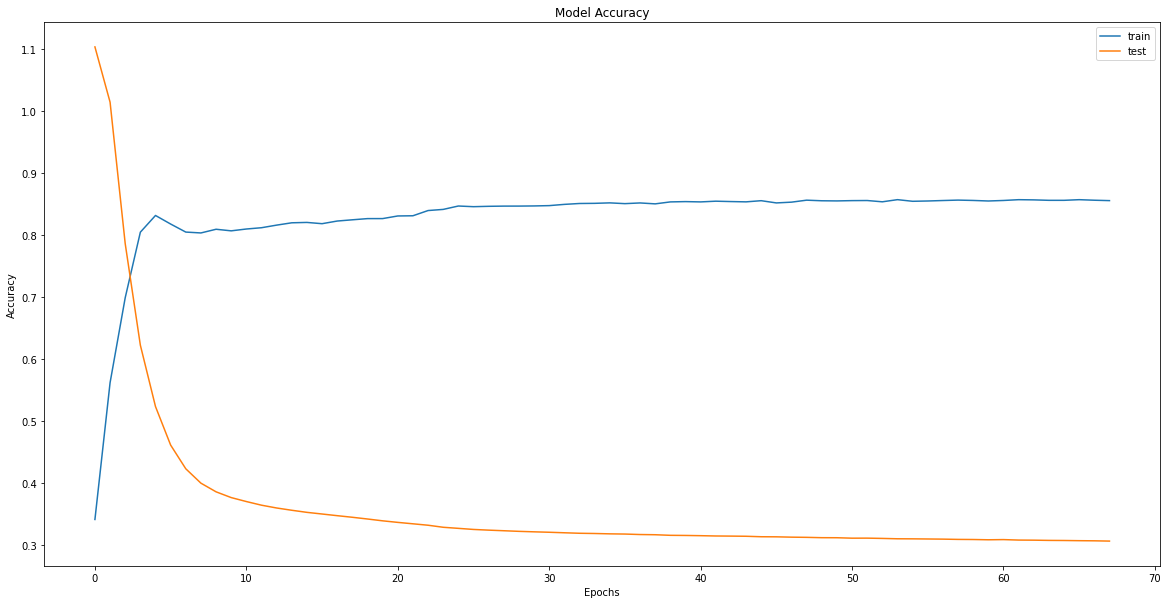
Keras Classifier 10-fold cross validation:

Accuracy : 0.88 (+/- 0.02)

Self-Defined ANN:

accuracy: 88.37%





precision recall f1-score support

0 0.83 0.72 0.77 3580

1 0.93 0.90 0.91 3571

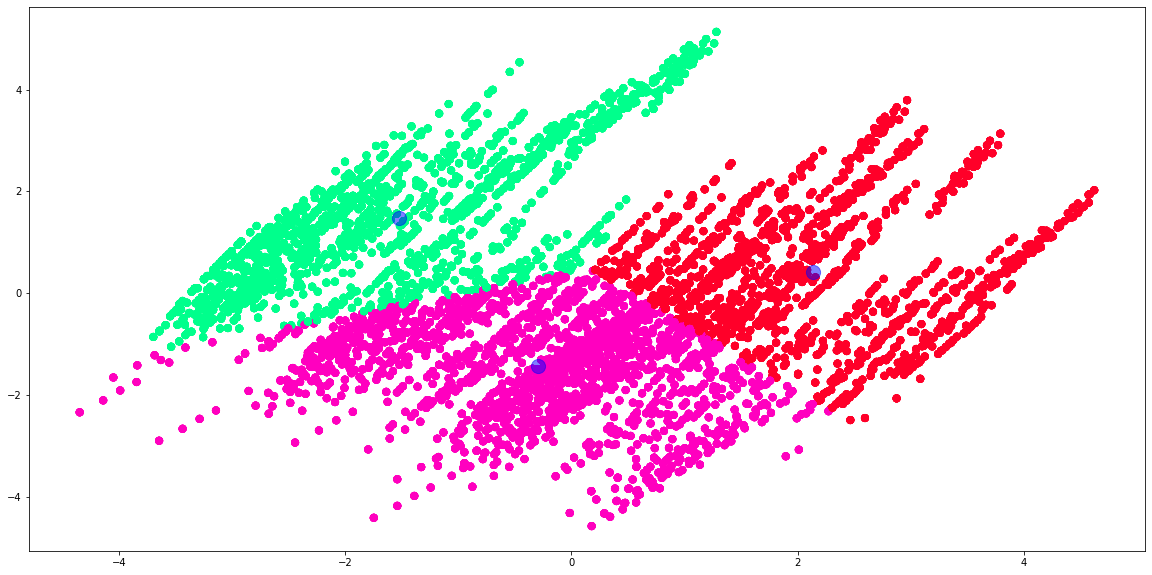
2 0.82 0.95 0.88 3585

**Clustering Approach (Before my own algorithm)**

Used PCA to reduce dimensionality to 2 features: PC1 and PC2.

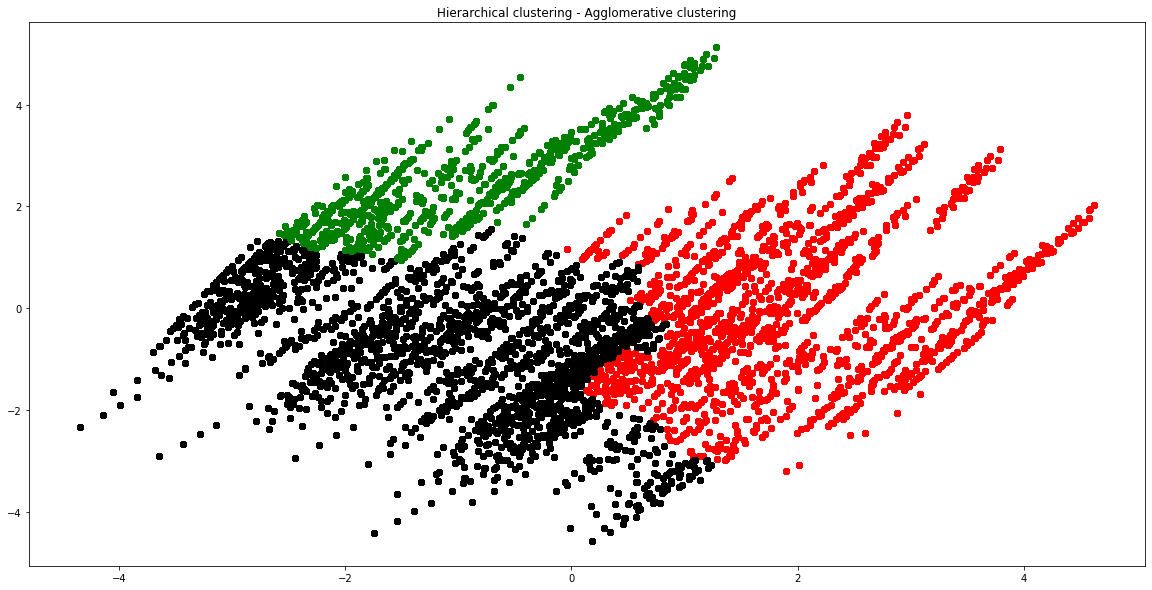
**Note:** This is clustering with twitter

**K-Means Results:**

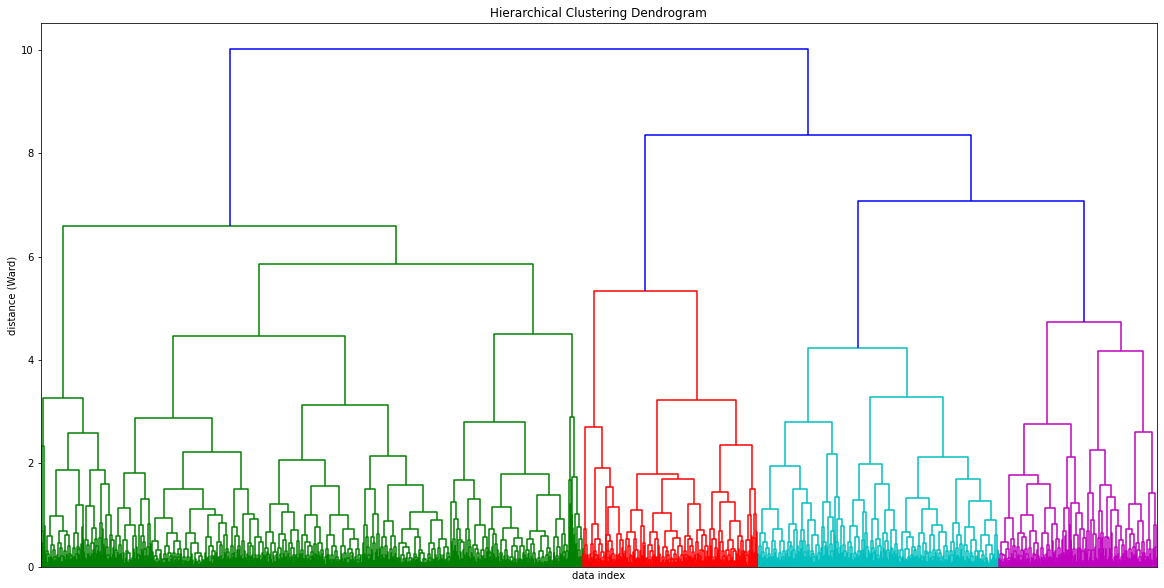
****

Accuracy: 0.6015727699530516

**Hierarchical Clustering**



Accuracy: 0.30614241001564946



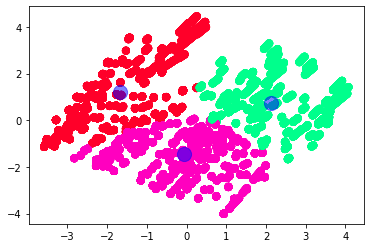
DBSCAN

Accuracy: 0.3333333333333333

|  |  |  |  |
| --- | --- | --- | --- |
| Metrics | Silhouette Score  Best value is: 1  Worst value: -1  0 indicates overlap | Calinski Harabasz  High value desirable: means the clusters are dense and well separated | Davies Bouldin  Best value is: 0  No max score  Lower the value, the better the cluster |
| K-Means | 0.4987 | 70658.2767 | 0.6646 |
| Agglomerative | 0.3287 | 21523.0176 | 0.9447 |
| DBSCAN | 0.3172 | 139481.4518 | 0.8524 |
| Gaussian Mixture | 0.4253 | 101780.5095 | 0.7767 |

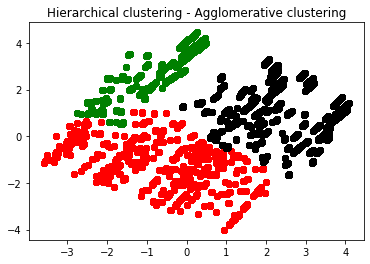
**Clustering Without Twitter Data**

**K-Means:**

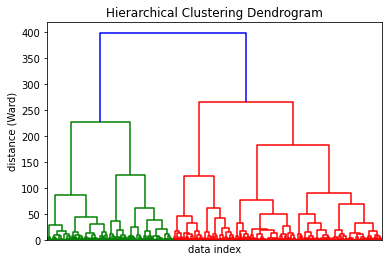


Accuracy: 0.24960876369327073

**Agglomerative Clustering:**

****

Accuracy: 0.2104851330203443



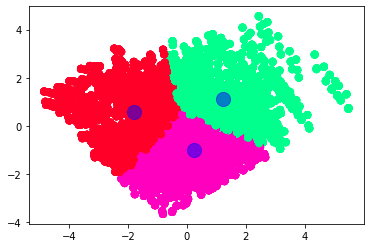
DBSCAN

Accuracy: 0.4338810641627543

|  |  |  |  |
| --- | --- | --- | --- |
| Metrics | Silhouette Score | Calinski Harabasz | Davies Bouldin |
| K-Means | 0.2023 | 8916.7504 | 1.6260 |
| Agglomerative | 0.2307 | 8757.3304 | 1.5863 |
| DBSCAN | 0.8456 | 6439.5440 | 0.7177 |
| Gaussian Mixture | 0.2024 | 9451.9827 | 1.8453 |

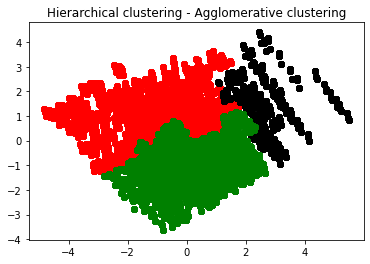
**Clustering With Statistically Significant Features**

K-Means:

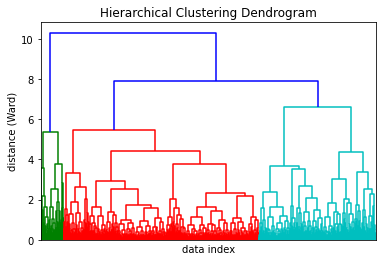


Accuracy: 0.3333333333333333

Agglomerative:



Accuracy: 0.32492175273865415



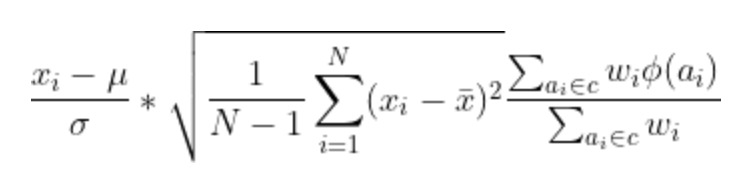
DBSCAN

Accuracy: 0.3333333333333333

|  |  |  |  |
| --- | --- | --- | --- |
| Metrics | Silhouette Score | Calinski Harabasz | Davies Bouldin |
| K-Means | 0.5157 | 75471.4352 | 0.6379 |
| Agglomerative | 0.5122 | 75051.6474 | 0.6389 |
| DBSCAN | 0.5914 | 5089798.6565 | 0.3991 |
| Gaussian Mixture | 0.4477 | 107086.3500 | 0.8708 |

**My Clustering Algorithm**

**C(f1, f2, f3, f4, f5, f6) = Z-score value \* Standard Deviation (Weighted K-Means)**



**C = min/max threshold from 0 – 1**

Set thresholds based on the value column since it had the highest weight

Accuracy: 0.8615023474178404

|  |  |  |  |
| --- | --- | --- | --- |
| Metrics | Silhouette Score | Calinski Harabasz | Davies Bouldin |
| My Clustering Algo. | 0.7032 | 184390.7946 | 0.4429 |