Clustering US Accident Data

(ISYE 7460) – Spring 2024

Abraham Jacob Abraham

ajacob@gatech.edu (GT ID Last 3: 600)

Submitted on 4/14/2024

1. Abstract:

Through this project, I analyze publicly available US accident data using different Clustering models to identify presence of possible hidden patterns. This involves analyzing weather conditions, road environment, location, and time predictors. The cluster details would eventually help safety officials and resource officers learn more about traffic accidents patterns across the country. I will be evaluating several different models on the dataset. I plan to use intrinsic evaluation measures (or internal measures) to gauge the quality of the clustering.

2. Introduction:

Traffic fatalities ranked as the leading cause of death for children and young adults in the U.S. in 2021. Beyond the loss of life, the economic cost of accidents in 2010 totaled USD 242 billion, with nearly USD 18 billion borne by taxpayers, according to the most recent report from the National Highway Traffic Safety Administration. Through this work, we hope to help reduce human and economic losses resulting from road accidents.

I will be running the different Clustering models by performing iterative search for best set of hyperparameters, as well as using existing packages from Python that identify best parameters based on Cross Validation. Some challenges involved were the run-time performance of the models. I had to finally use a sampled dataset (~100K records) for most of the models to run within 5 minutes on an 8GB RAM machine. Another issue which I faced was with regards to identifying or capturing valuable trends / patterns in the data. The inherent data associations were found to have a higher dimension relationship which becomes hard to express on a 2d or 3d chart.

This analysis could provide impetus and the tools necessary for government and other civil agencies to understand specifics on under what circumstances would accidents occur and plan out strategies to combat them. The goal would be to
we hope to help reduce human and economic losses resulting from road accidents

3. Problem Statement with Data Sources:

The data for this project was obtained from this site: https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents. This is a countrywide car accident dataset that covers **49 states of the USA**. The accident data were collected from **February 2016 to March 2023**, using multiple APIs that provide streaming traffic incident (or event) data. These APIs broadcast traffic data captured by various entities, including the US and state departments of transportation, law enforcement agencies,

traffic cameras, and traffic sensors within the road networks. The dataset currently contains approximately **7.7 million** accident records.

Note: I used the sampled **500K** record set for my purpose, as during my iterative model building process, I did find that the run-time of the models was in hours, and I was not able to get the needed output. The goal of the project is to:

- Perform necessary EDA on the data.
    - Identify outliers and decide strategies on how to deal with them
    - Look at correlation between the variables and extract only the ones that would be beneficial for the clustering model
    - Perform other EDA and perform Data filtering as needed
- Execute different Clustering models. Measure which model was able to cluster the data 'better.'
- Look to visualize the clusters and derive any insights
- If the visual scale does not work, look at insight derivation using other methods
- Learning opportunities for future

4. Proposed Methodologies:

After performing the necessary EDA on the data, I plan to run the below models for Clustering:

1. Run Kmeans after dimensionality reduction using FAMD
2. Run Gaussian Mixture models after dimensionality reduction using PCA
3. Run Spectral Clustering after dimensionality reduction using PCA
   TSNE is also being used as a dimensionality reductions strategy but more so from a visualization standpoint.

Some quick notes on the models that would be used.

4.1 Dimensionality Reduction:

We explored PCA, TSNE and FAMD for dimensionality reduction.

PCA is a linear dimension reduction technique that seeks to maximize variance and preserves large pairwise distances. For example, PCA assumes that the data is linear, meaning that the variables have a straight-line relationship with each other. However, if the data is nonlinear such as curved or cyclical patterns, PCA may not capture the underlying structure well. This can lead to poor visualization especially when dealing with non-linear manifold structures.

TSNE (t-Distributed Stochastic Neighbor Embedding) is an unsupervised, non-linear technique primarily used for data exploration and visualizing high-dimensional data. In simpler terms, TSNE gives you an intuition of how the data is arranged in a high-dimensional space. TSNE differs from PCA by preserving only small pairwise distances or local similarities whereas PCA is concerned with preserving large pairwise distances to maximize variance. T-SNE provides better visualizations of clustered data. It is an effective and efficient graphical way to assist cluster analysis with respect to determining the number of clusters and cluster memberships. PCA visualizations are often not ideal, because data of one cluster may significantly overlap data of another cluster but TSNE shows clear separation in the data .

FAMD (Factor Analysis of Mixed Data) is a method of dimensionality reduction when the sample involves variables that are of mixed data types. This is an extremely valuable alternative approach, combining PCA for continuous variables and multiple correspondence analysis (MCA) for categorical variables. This is specifically important for this project, as the accident data has both continuous and categorical data attributes

4.2 Clustering Algorithms:

K-means clustering aims to partition data into k clusters in a way that data points in the same cluster are similar and data points in the different clusters are farther apart. Similarity of two points is determined by the distance between them.

To begin, we choose a value for k (the number of clusters) and randomly choose an initial centroid for each cluster. We then apply a two-step process:

- Assignment step — Assign each observation to its nearest center.
- Update step — Update the centroids as being the center of their respective observation.

We repeat these two steps over and over until there is no further change in the clusters. At this point, the algorithm has converged, and we may retrieve our final clustering's.

Gaussian mixture models are a probabilistic model for representing normally distributed subpopulations within an overall population. Mixture models in general do not require knowing which subpopulation a data point belongs to, allowing the model to learn the subpopulations automatically. Since subpopulation assignment is not known, this constitutes a form of unsupervised learning.

The result of each cluster is not associated with a hard-edged sphere, but with a smooth Gaussian curve. Just as in the k-means expectation-maximization approach this algorithm can sometimes miss the globally optimal solution, and thus in practice multiple

random initializations are used. Though GMM is often categorized as a clustering algorithm, fundamentally it is an algorithm for density estimation. That is to say, the result of a GMM fit to some data is technically not a clustering model, but a generative probabilistic model describing the distribution of the data.

Spectral clustering is based on the principles of graph theory and linear algebra. Instead of directly clustering the data in the input space, it constructs a similarity graph where data points are nodes and edges represent similarities between the points. The algorithm then leverages the spectral properties of the graph (i.e., the eigenvalues and eigenvectors of the graphs Laplacian matrix) to project the data into a lower dimensional space. In the transformed space, traditional clustering techniques, such as k-means, can be applied more effectively, even when the data has complex, non-linear boundaries in the original space.

Additionally, I will be performing Hyperparameter tuning on each of the models to determine the best set of parameters to use in the final model. I will be using the RandomizedSearchCV package in Python for this purpose.

4.3 Measuring quality of clustering

There are two main classes for evaluating quality of clusters. We use **external evaluations** when there is a true output label present, and we use **internal evaluations** when there are no true labels present. For our use case, we will be using **internal evaluations**. Some of the evaluation measures are:

- ***Silhouette Index****: A composite index reflecting the compactness and separation of clusters****. A larger average Silhouette index indicates a better overall quality of the clustering result*** .

- ***Calinski-Harabasz Index***: The score is defined as ratio of the sum of between cluster dispersion and of within-cluster dispersion. ***Higher values show better clustering***. The Calinski-Harabasz index is like the F Statistic used in ANOVA.

- ***Davies-Bouldin Index****: The score is defined as the average similarity measure of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between-cluster distances. Thus, clusters which are farther apart and less dispersed will result in a better score.* ***Lower values show better clustering.***

The following metrics were also used for cluster evaluation:

- **F1 Score**: The intrinsic methods discussed thus far looks at understanding the

underlying clusters of the data and how truly separable they are: all with no knowledge of true labels. Once we have such clusters, it is always useful to run a CV F1 Score on top of it along with the true labels to get a feel of how good the cluster assignment was in terms of classification. We used the LGBM Classifier Multi-Class Classifier for this purpose. An F1 score closer to 1 indicates the highest quality model and all the clustering models we created had a good F1 score indicating that a classification model can correctly predict high-quality clusters.

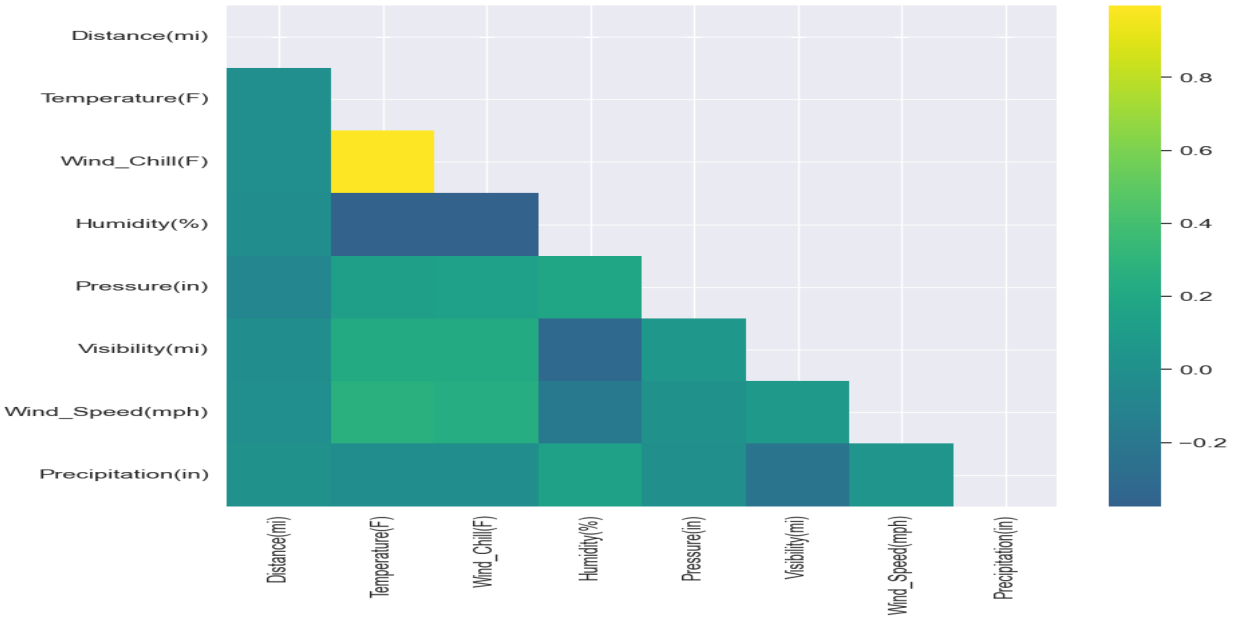
- **AIC & BIC**: Akaike's Information Criterion (AIC) and the Bayesian Information Criterion (BIC) are used for evaluating Gaussian Mixture Model (GMM). As GMM is more of a density estimation method than clustering, we need to look at the data from that aspect rather than intrinsic clustering, hence AIC/BIC provide a better measure to evaluate the results for GMM.

5. Analysis and Results – EDA :

We started off with the sample set of 500K records. I looked to further reduce the count as running the models on this count was still taking time in the range of > 4 hours on a machine with 8GB RAM. Therefore, I looked to filter on accident records from the top 3 States (**California, Florida, and Texas**) and post Jan 1, 2020. After the filtering, the count came down to **130661**. The State wise split was as below. There were no duplicates found based on all columns.

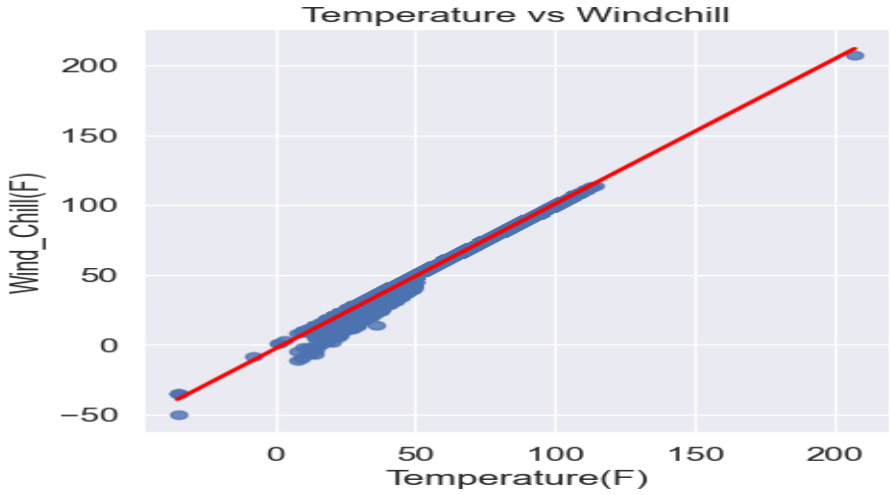
California: 70169
Florida: 42220
Texas: 18272

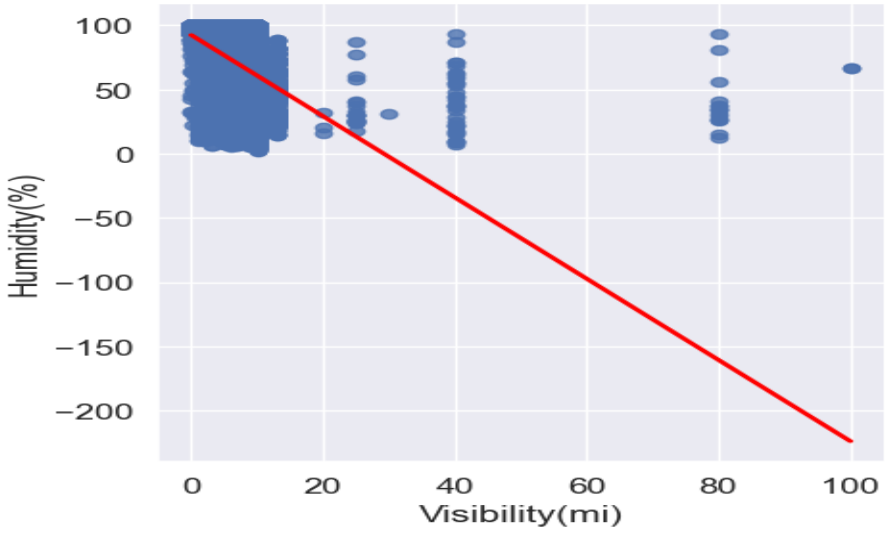
1. I started by looking at NULL attributes in County, State and Zip Code. Zip Code had about 59 rows with Nulls. Since the count was only .045%, those were removed.
2. We looked at the Continuous features (*'Temperature(F)', 'Wind_Chill(F)', 'Humidity(%)', 'Pressure(in)', 'Visibility(mi)','Wind_Speed(mph)', 'Precipitation(in)'*), and removed rows that had NULLS in all the continuous columns. This was necessary, because these columns along with the Categorical ones are important for clustering and having NULL in all columns does signify data capture issues.
3. Next, I looked at correlation analysis of the Continuous variables.

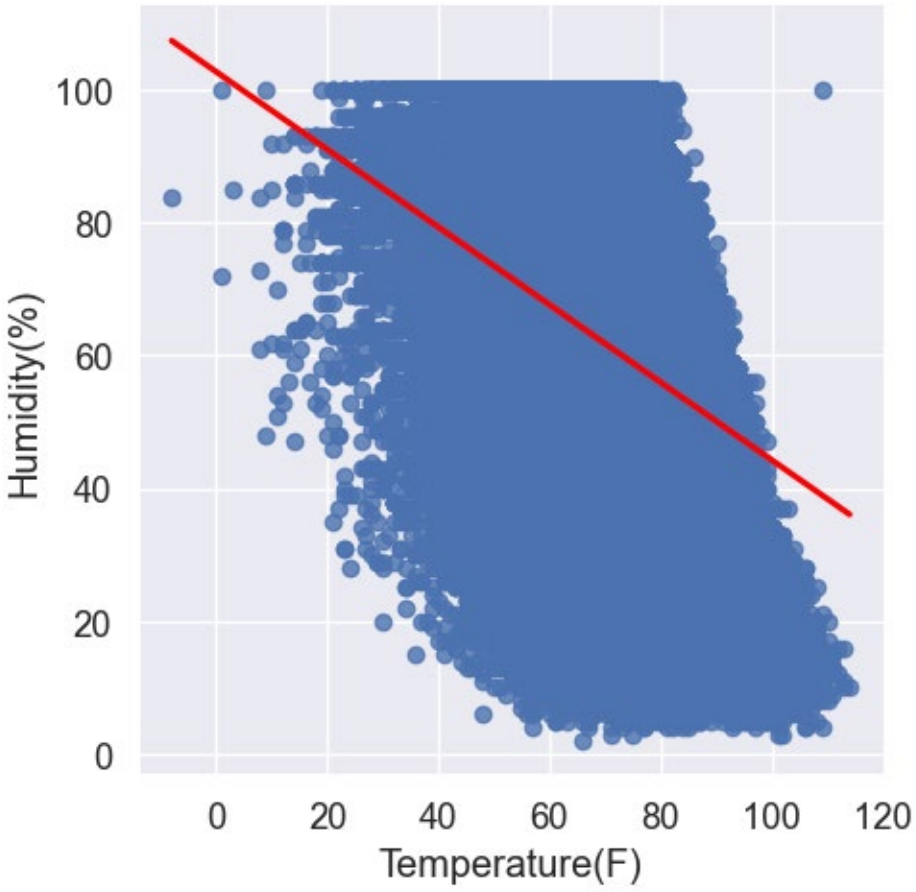


4. We can see that Temperature and Wind Chill are highly correlated. Similarly, there is negative correlation between Humidity and Visibility, Temperature and Humidity. We will look to perform VIF analysis to understand which variables can be removed.

   The above is also understandable by looking at scatter plots.



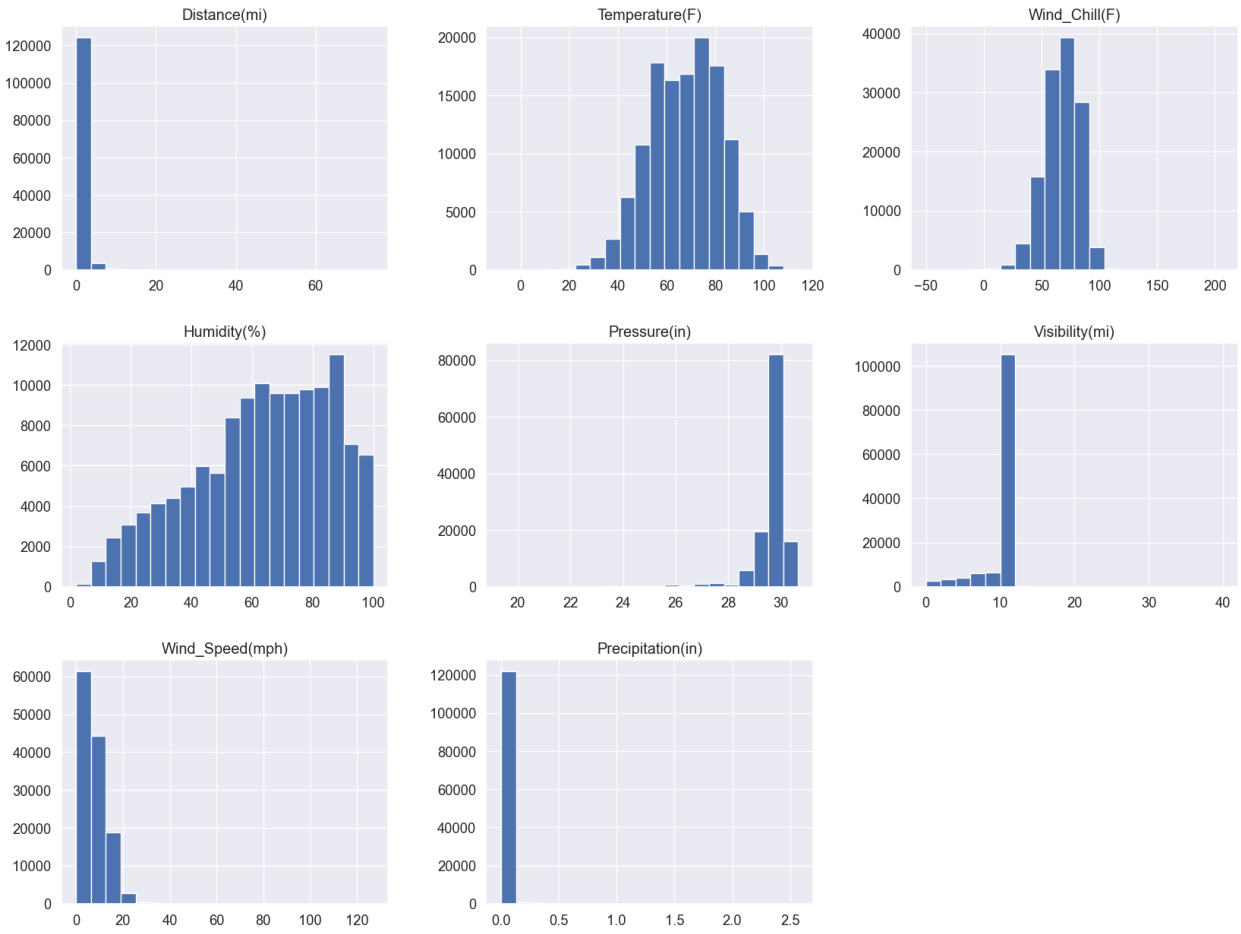




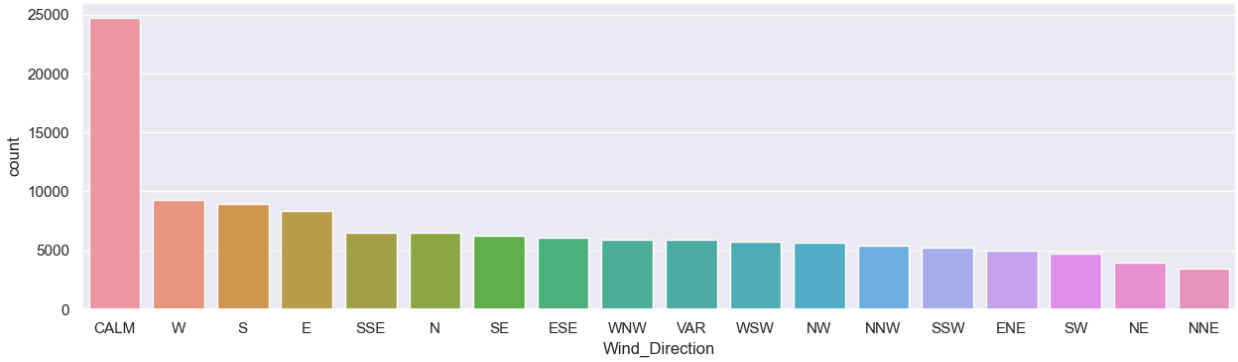
5. Based on the VIF analysis, we can see that Temperature, Visibility and Humidity have VIF scores of greater than 4. A VIF of four means that the variance (a measure of imprecision) of the estimated coefficients is four times higher because of correlation between the two independent variables

Let us further look at outliers before making decisions on which variables to keep.

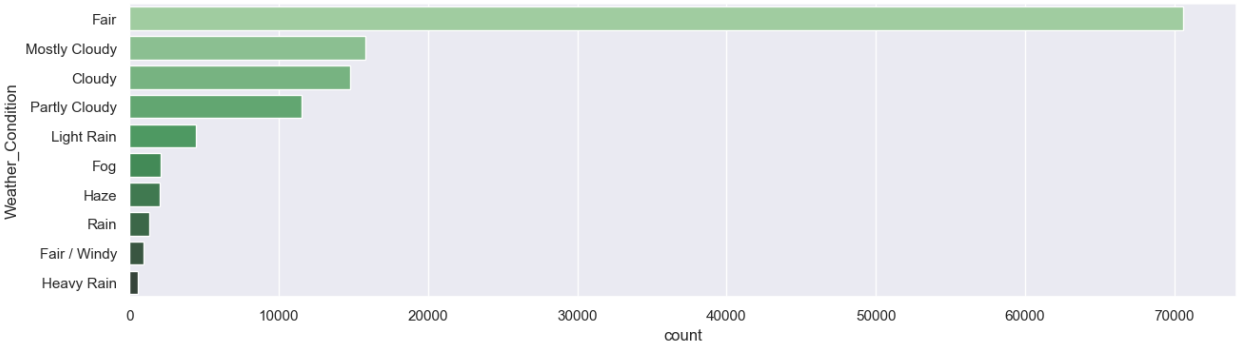
6. Additionally NULLS in Humidity, Wind Speed, Pressure and Temperature individually were fixed using interpolation, while NULLS in Wind Direction and Sunrise Sunset were set with the mode value of the specific features (as they were Categorical).
7. The method I used to check for outliers was via the Interquartile range (IQR). For Minor Outliers (IQR x 1.5) was used as threshold. For Major Outliers (IQR x 3) was used. Based on this, there were 6 rows with Outliers for Temperature, 528 rows with outliers for Windspeed and 4423 for Outliers on Pressure.
    a. For Temperature, instead of removing the records, I decided to update the outlier values with the median value (68 degrees)
    b. For Windspeed though, I did some more analysis. The strongest wind ever recorded in the United States (not including tornadoes or hurricanes) was recorded at the summit of Mount Washington, New Hampshire, one of the windiest places on earth. On April 12th, 1934, a wind gust was recorded at 231 mph! Therefore, instead of IQR, I only looked at outliers above the 231 range. It came down to just 1 record and I updated that with the median (7 mph)
    c. For Pressure, The highest pressure ever recorded in the lower 48 states occurred in December 1983 in Miles City, Montana, where it reached 1,064 mb, or 31.42 inHg, during a severe cold wave. Using that range instead of IQR, it gave 1 record which was updated with the median value (29.88)
    d. Precipitation, Humidity did not have any outliers
    e. For Visibility, we found around 23K records impacted, which is huge. I just decided to keep those values. And the outliers were replaced with the median value (10 miles).
    f. The continuous feature distribution after outlier updates is shown below:



8. Finally, concluding with the VIF analysis, excluding Visibility brought down the VIF scores for all Continuous variables to below 4. So, we will be removing it.
9. Next is the Categorical variable analysis.
    a. Severity is heavily skewed to value 2. There is no point in considering this feature for clustering.
    b. Wind Direction: I had to do an update to the mapping values per https://windy.app/blog/what-is-wind-direction.html . Results are as below. Since there is a fair split, we can consider this feature.



c. Below is the split on Weather condition. Since almost 70% of the data is skewed on Fair weather, I decided to not consider this feature for clustering



d. Variables like (Turning Loop, Side, Civil Twilight, Nautical Twilight, Astronomical Twilight, No Exit, Bump, Traffic Calming) are heavily skewed to a single Category and hence would not be considering them

10. Additionally, I added Weekday, Year, Month and Hour to the model (based on the Accident Start Time field) to see if that would aid in clustering.

The final set of variables considered for building the Clustering models are as below. And we are left with ~116K records to cluster on

Continuous variables:

- Temperature
- Windspeed
- Air pressure
- Distance(mi)
- Humidity

Categorical variables:

- Stop
- Traffic_Signal
- Give_Way
- Wind_Direction
- Junction
- Crossing
- Sunrise_Sunset
- *Weekday – extracted from the Start Time attribute*
- *Hour - extracted from the Start Time attribute*
- *Year - extracted from the Start Time attribute*
- *Month - extracted from the Start Time attribute*

6. *Analysis and Results – Model building :*

6.1 Kmeans with FAMD

Before commencing with FAMD and Kmeans, I did a Hopkins test on the Continuous variables to check if there was clustering ability present in the data. A score close to 1 signifies that there is clustering possibility present. I got a score of **.946**, which does signify that we can proceed forward. As explained earlier, Kmeans only works with numerical data. Therefore, we need to perform some conversion and dimensionality reduction to be able to perform good clustering. FAMD is one such strategy. It is like PCA, except that it works with mixed data types.

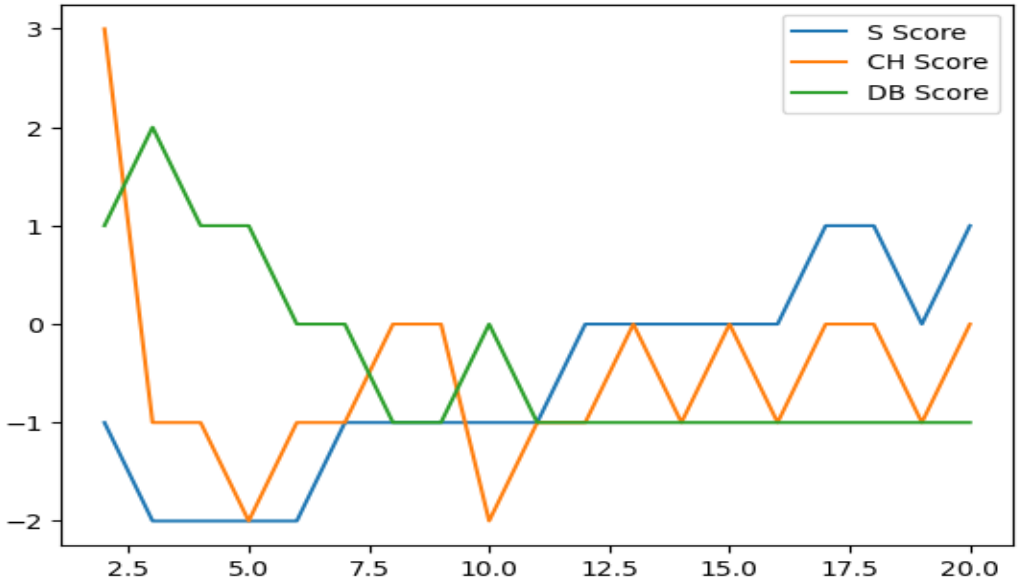
The dataset has 5 Continuous variables and 11 Categorical variables. If we count the continuous plus the distinct categories in the Categorical variables, we count to about 80 different columns if we had to do One Hot Encoding ! I finally fixed a FAMD component count of 65 (this explains 98.9% of the variance) and transformed the data

I tried to get the best number of clusters from Kmeans, by two methods. One was brute force but looking at cluster values from a range and calculating the Davies Bouldin, Calinski Harabasz, Silhouette and CV F1 Score. And then looking at the cluster value with higher CH and Silhouette Scores and CV F1 Score and lowest DB Score. *Note: For CV F1, it is doing a multi-class classification with the LGBM model.*

After testing, I found that cluster count of 17 gave the best result with scores as below. We could have gone with 20, but there was negligible difference for the measures when compared to Cluster # 17.

Silhouette	DB	CH	CV F1 score
.085	2.9	2108.61	99%

The accompanying graph below shows each Cluster for which the model was evaluated and its corresponding measurements cores. The X Axis shows the cluster value, while the y axis is the normalized score for Silhouette, CH, and DB values.



Next, I tried a **RandomizedSearchCV** on Kmeans to pick the best set of hyperparameters with scoring set on Silhouette. This though, gave me a cluster count of 20. I decided to go with Cluster count of 17 because I felt the larger number of clusters, it might over-generalize the data.

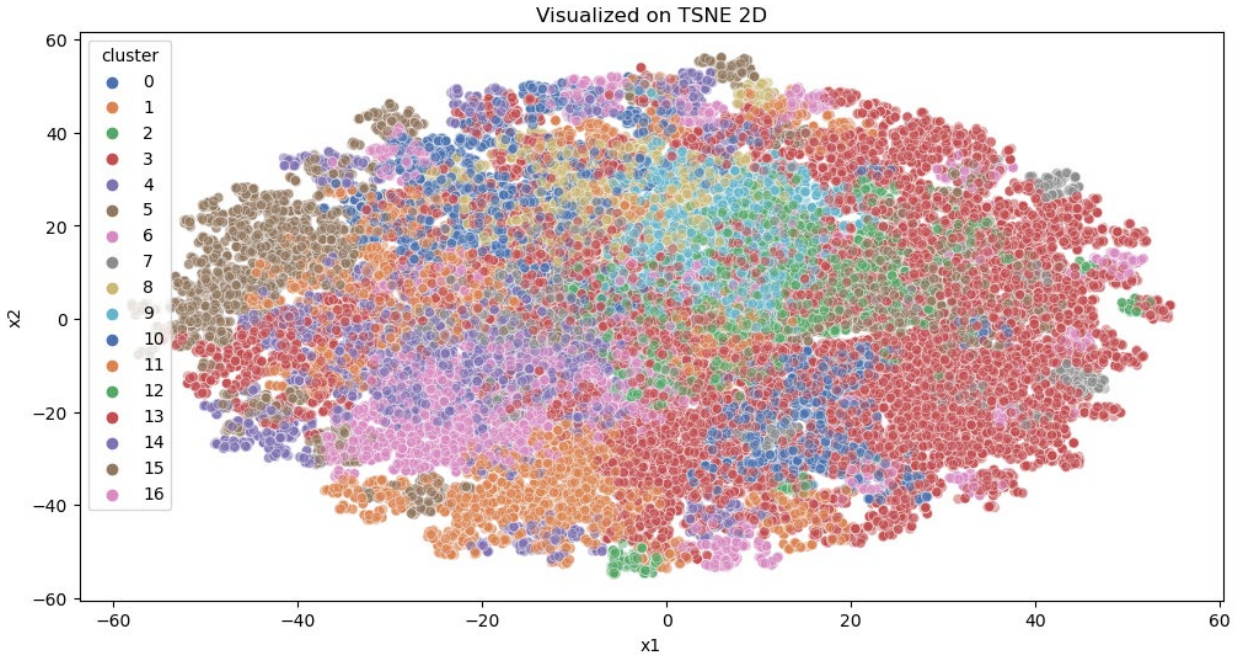
The Cluster count distribution is given below:

Cluster	Count	Cluster	Count
0	4330	9	6015
1	4100	10	6044

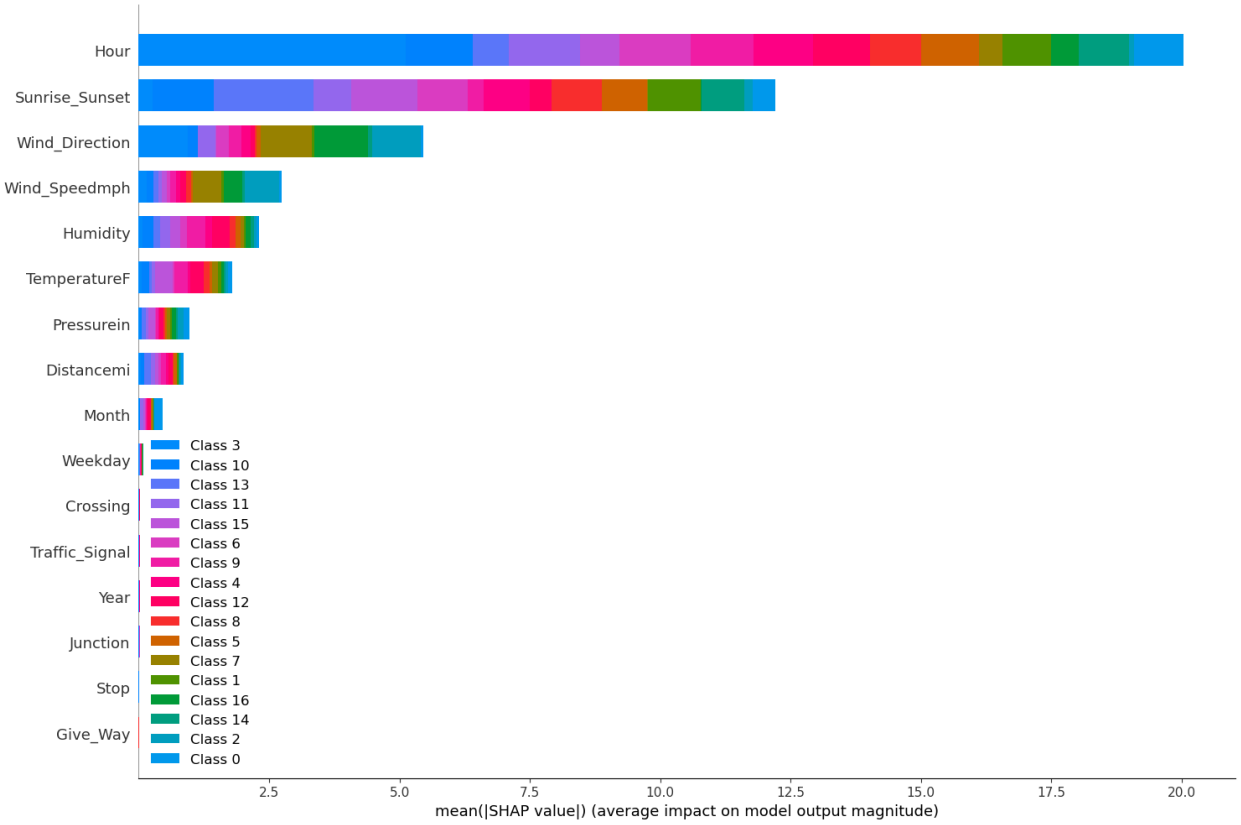
2	3947	11	7257
3	**37894**	12	5275
4	5414	13	3007
5	5173	14	4437
6	6487	15	3625
7	4055	16	4110
8	4368		

Some visualizations based on the final Kmeans model run of 17 are given below:

1. **TSNE Visualization**: I applied TSNE dim reduction on top of the FAMD'ed set to see how the visualization would look. Although there is some overlap, it does produce some good cluster formations. We can especially see that Cluster 3 is spread across having the largest presence (~38K).



2. Based on the LGBM Multi-Class Classification done on the dataset with Cluster association as the response variable, I tried to look at which set of features most contributed to the cluster assignment. The results are below, and we can see that **Hour, Sunrise_Sunset and Wind_Direction** were the major features. This is done via the SHAP Feature importance analysis



3. I also took a stab at taking the min, max and mean / median values of the features in each Cluster

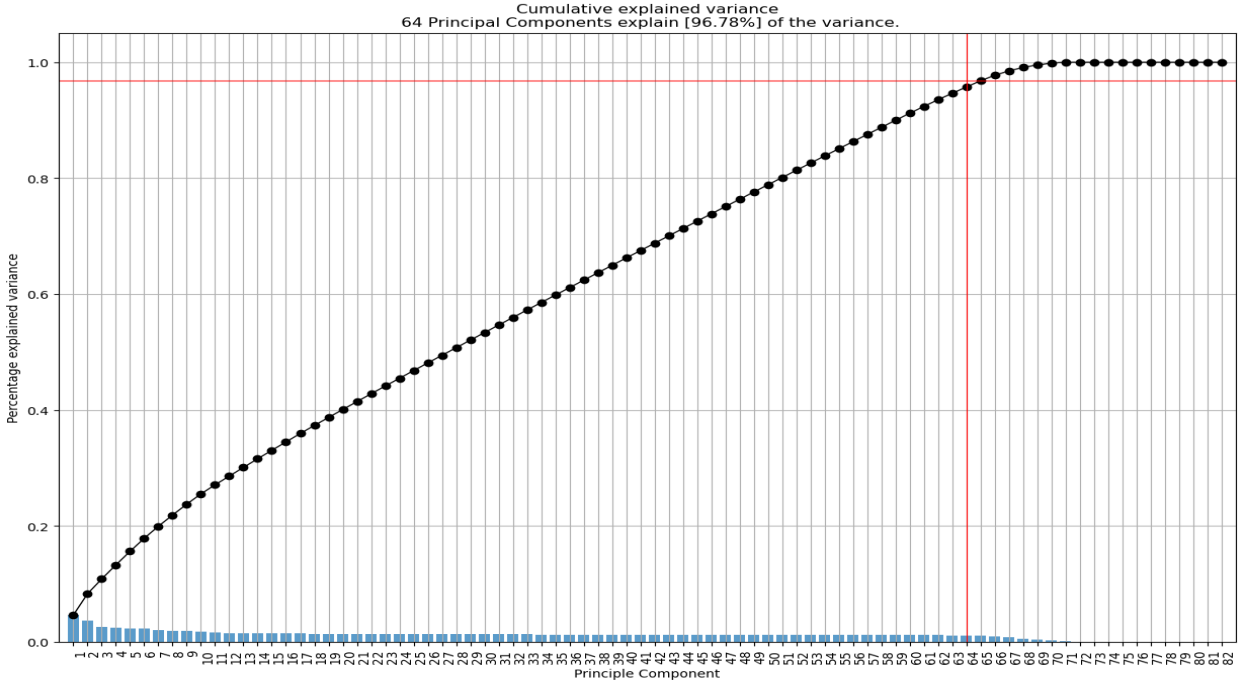
	Temperature(F)			Month			Wind_Speed(mph)			Pressure(in)			Humidity(%)			Distance(mi)			Weekday			Year			Hour		
	mean	min	max	median	min	max	mean	min	max	mean	min	max	mean	min	max	median	min	max	median	min	max	median	min	max	median	min	max
cluster																											
0	66.331409	12.0	108.0	6.0	1	12	6.993457	0.0	41.0	29.632893	22.87	30.44	63.384950	4.0	100.0	0.0870	0.0	1.830	3.0	0	6	2021.0	2020	2023	19.0	19	19
1	69.207317	3.0	100.0	6.0	1	12	7.181545	0.0	70.0	29.671301	23.35	30.59	60.832317	5.0	100.0	0.0760	0.0	1.824	3.0	0	6	2021.0	2020	2023	10.0	10	10
2	71.538257	21.0	109.0	6.0	1	12	9.071193	3.0	127.0	29.581497	22.86	30.38	59.189891	3.0	100.0	0.1040	0.0	1.831	3.0	0	6	2021.0	2020	2023	14.0	0	23
3	66.429096	-8.0	113.0	6.0	1	12	6.807437	0.0	77.0	29.638496	22.89	30.52	63.542263	2.0	100.0	0.1160	0.0	1.831	3.0	0	6	2021.0	2020	2023	17.0	0	23
4	74.803903	19.0	113.0	7.0	1	12	9.126339	0.0	37.0	29.692623	23.10	30.42	50.980298	5.0	100.0	0.1335	0.0	1.828	3.0	0	6	2021.0	2020	2023	13.0	13	13
5	73.290869	8.0	107.0	6.0	1	12	8.660738	0.0	40.0	29.681389	22.67	30.59	53.760842	4.0	100.0	0.1300	0.0	1.830	3.0	0	6	2021.0	2020	2023	12.0	12	12
6	75.190792	12.0	114.0	6.0	1	12	9.423642	0.0	33.0	29.687705	22.81	30.48	49.946174	4.0	100.0	0.1440	0.0	1.831	3.0	0	6	2021.0	2020	2023	14.0	14	14
7	63.066543	11.0	111.0	6.0	1	12	8.904316	1.0	40.0	29.744910	23.96	30.60	61.338307	3.0	100.0	0.1000	0.0	1.823	3.0	0	6	2021.0	2020	2023	14.0	0	23
8	65.966194	15.0	106.0	6.0	1	12	6.259043	0.0	35.0	29.662035	22.60	30.66	66.805556	4.0	100.0	0.0650	0.0	1.831	3.0	0	6	2021.0	2020	2023	9.0	9	9
9	60.856276	9.0	94.5	6.0	1	12	4.315877	0.0	32.0	29.713697	23.24	30.44	78.739651	5.0	100.0	0.0330	0.0	1.830	2.0	0	6	2021.0	2020	2023	7.0	7	7
10	63.596349	8.0	93.0	6.0	1	12	5.000662	0.0	29.0	29.741669	22.79	30.60	74.338269	6.0	100.0	0.0205	0.0	1.831	2.0	0	6	2021.0	2020	2023	8.0	8	8
11	74.518511	22.0	111.0	6.0	1	12	9.455537	0.0	36.0	29.663923	23.07	30.43	52.743627	3.0	100.0	0.1280	0.0	1.830	3.0	0	6	2021.0	2020	2023	16.0	16	16
12	59.040664	8.0	104.0	6.0	1	12	4.655893	0.0	39.0	29.676687	22.79	30.51	80.159242	8.0	100.0	0.0430	0.0	1.830	2.0	0	6	2021.0	2020	2023	6.0	6	6
13	61.713003	11.0	97.0	7.0	1	12	5.312271	0.0	36.0	29.625005	23.13	30.52	70.840040	6.0	100.0	0.1150	0.0	1.830	3.0	0	6	2021.0	2020	2023	22.0	22	22
14	71.698332	17.0	111.0	6.0	1	12	7.740290	0.0	38.0	29.665137	22.90	30.59	56.424160	5.0	100.0	0.0870	0.0	1.830	3.0	0	6	2021.0	2020	2023	11.0	11	11
15	57.848000	12.0	91.0	5.0	1	12	4.507034	0.0	32.0	29.628927	22.89	30.49	78.868690	10.0	100.0	0.0960	0.0	1.828	3.0	0	6	2021.0	2020	2023	5.0	5	5
16	71.317397	18.0	112.0	6.0	1	12	9.695377	3.0	40.0	29.613491	22.82	30.36	66.320560	6.0	100.0	0.0990	0.0	1.830	3.0	0	6	2021.0	2020	2023	15.0	0	23

The SHAP analysis mentioned that Hour influences cluster formations. We could see that certain clusters (like 0, 1,4,5,6 etc.) only have accidents pertaining to a specific hour as compared to Clusters like 2,3 which is more spread out.

6.2 GMM with PCA

The next model evaluated was GMM. We must do dimensionality reduction, but this time, I opted for One-Hot Encoding to convert the Cat variables to numerical and then performed PCA on top of it.

With PCA, we looked to use 64 components which explains 97% of variance.

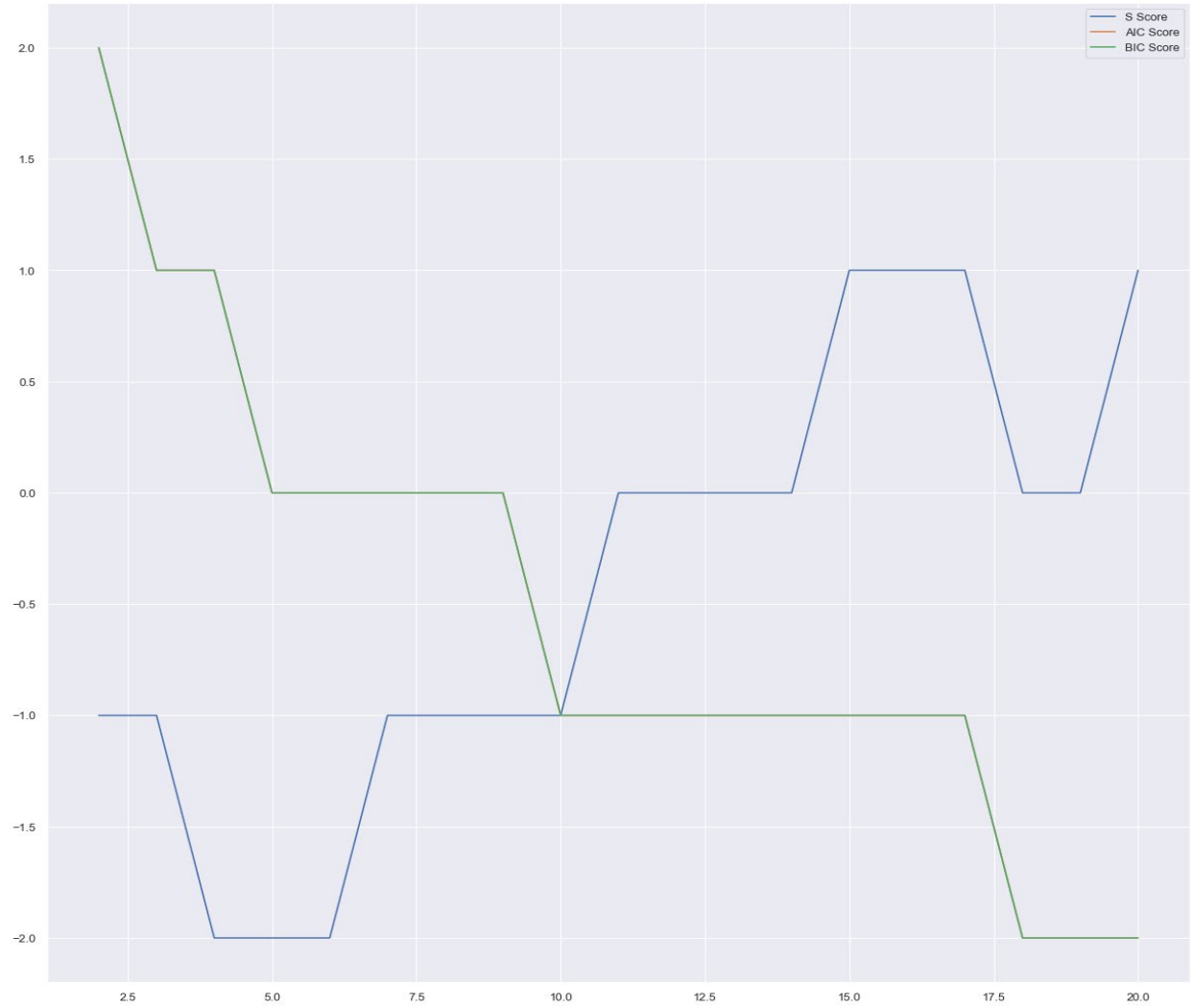


I started with Brute Force on GMM to identify best cluster number based on Silhouette, AIC and BIC Scores. The result was with Cluster value of 20. The other scores are :

Silhouette: 0.079440
AIC: -1.124725e+07
BIC: -1.083275e+07

Silhouette	**AIC**	**BIC**	**CV F1 score**
.079	*-1.124725e+07*	*-1.083275e+07*	73%

The accompanying graph below shows each Cluster for which the model was evaluated and its corresponding measurements cores. The X Axis shows the cluster value, while the y axis is the normalized score for Silhouette, AIC, and BIC



And then by using **RandomizedSeachCV method**, it also produced optimal Cluster # of 20 for GMM.

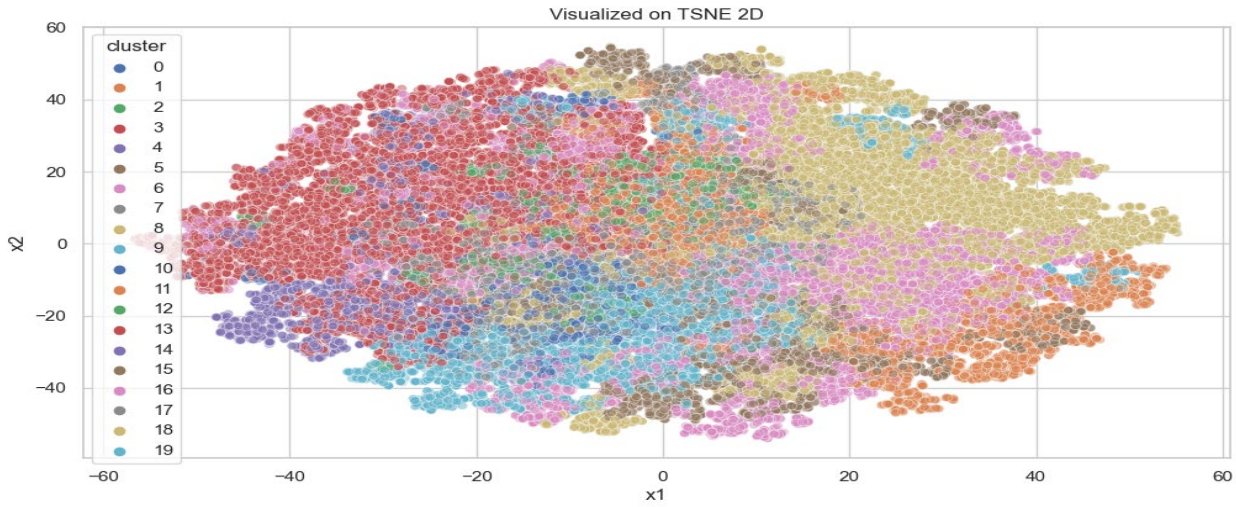
The final Cluster count distribution is given below:

Cluster	Count	Cluster	Count
0	3084	10	3556
1	6247	11	4393
2	3817	12	3732
3	20989	13	2227
4	4681	14	2284
5	6723	15	263
6	14924	16	4
7	4224	17	5668
8	13143	18	6398
9	6883	19	2903

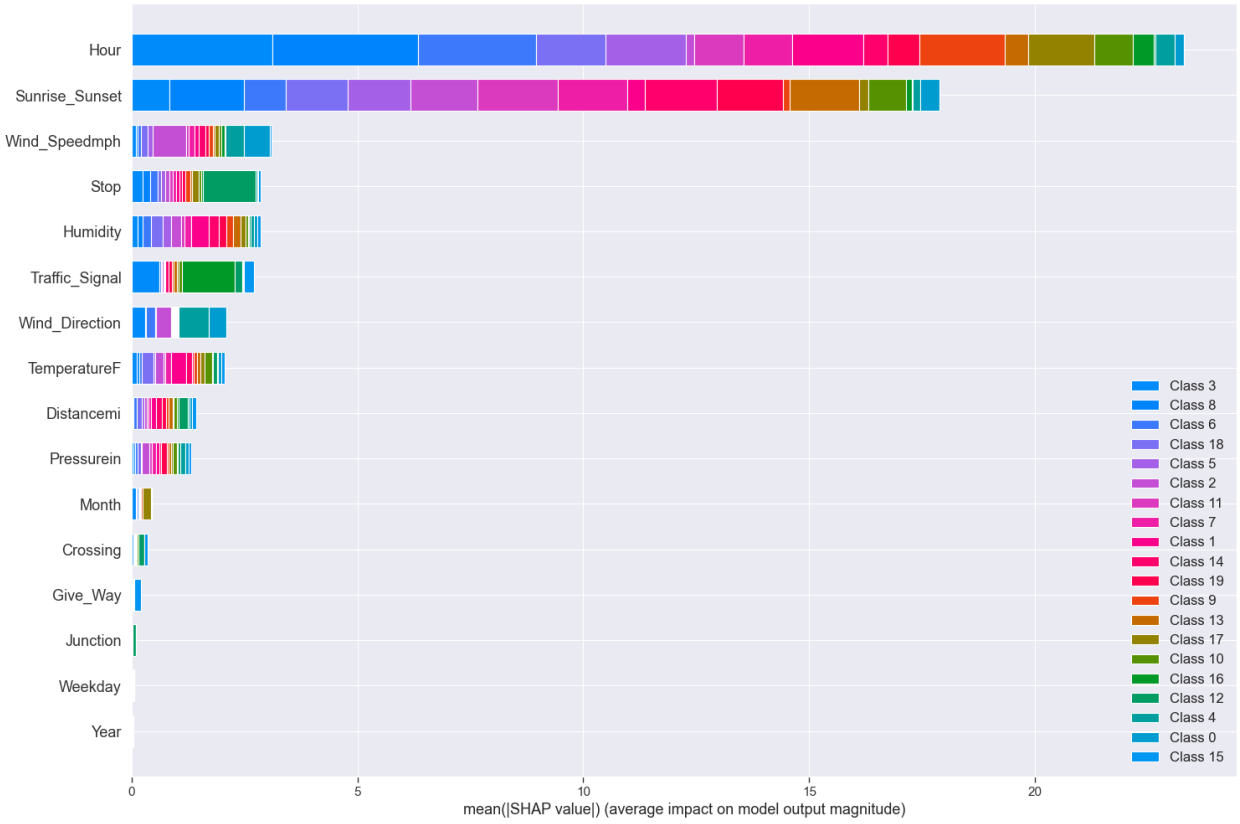
We can see that for one Cluster (#16), there are just 4 members, and instead of one Cluster in Kmeans that had > 35K assignments, in GMM, the membership is spread out.

Some visualization like what was done with Kmeans before:

1. **TSNE Visualization**: Compared to the one produced on top of Kmeans, this seems a little more uniform in terms of spread and not so much overlap (though there still is). Unlike again with Kmeans (where Cluster 3 had the major share), most of the have the same number of records.



2. **SHAP Important Features:**



Once again, **Hour, Sunrise_Sunset (**not so much **Wind_Speed**) are the top determiners of Cluster membership.

3. Cluster Compositions:

	Temperature(F)			Month			Wind_Speed(mph)			Pressure(in)			Humidity(%)			Distance(mi)			Weekday			Year			Hour		
	mean	min	max	median	min	max	mean	min	max	mean	min	max	mean	min	max	median	min	max	median	min	max	median	min	max	median	min	max
cluster																											
0	61.277567	22.0	89.0	8.0	1	12	6.531052	3.0	25.0	29.799664	24.80	30.47	72.228771	6.0	100.0	0.0890	0.0	1.828	3.0	0	6	2021.0	2020	2023	6.0	1	23
1	60.863336	9.0	94.5	6.0	1	12	4.732298	0.0	32.0	29.721961	23.24	30.48	78.868212	5.0	100.0	0.0310	0.0	1.830	2.0	0	6	2021.0	2020	2023	7.0	7	7
2	75.033721	31.0	108.0	6.0	1	12	9.761628	3.0	36.0	29.665186	23.63	30.25	52.807558	7.0	100.0	0.1085	0.0	1.818	3.0	0	6	2021.0	2020	2023	15.0	15	16
3	59.837414	-8.0	112.0	6.0	1	12	4.950671	0.0	41.0	29.620272	22.79	30.52	73.803268	4.0	100.0	0.1140	0.0	1.831	3.0	0	6	2021.0	2020	2023	6.0	1	23
4	66.195417	14.0	100.0	6.0	1	12	7.829408	3.0	31.0	29.538555	23.40	30.34	77.193826	13.0	100.0	0.0790	0.0	1.827	3.0	0	6	2021.0	2020	2023	6.0	1	23
5	75.089786	12.0	114.0	6.0	1	12	9.545892	0.0	33.0	29.687476	22.81	30.48	50.319157	3.0	100.0	0.1420	0.0	1.831	3.0	0	6	2021.0	2020	2023	14.0	14	14
6	74.758718	14.0	112.0	6.0	1	12	9.599777	0.0	127.0	29.660879	23.07	30.44	51.512441	3.0	100.0	0.1330	0.0	1.830	3.0	0	6	2021.0	2020	2023	16.0	15	16
7	66.075601	15.0	106.0	6.0	1	12	6.280106	0.0	35.0	29.664386	23.33	30.66	66.835636	4.0	100.0	0.0690	0.0	1.831	3.0	0	6	2021.0	2020	2023	9.0	9	9
8	72.726621	3.0	113.0	6.0	1	12	8.527462	0.0	70.0	29.679169	22.67	30.59	54.759893	4.0	100.0	0.1210	0.0	1.831	3.0	0	6	2021.0	2020	2023	12.0	10	13
9	72.324483	19.0	113.0	6.0	1	12	8.955623	0.0	77.0	29.647862	22.82	30.44	55.814502	4.0	100.0	0.1290	0.0	1.827	3.0	0	6	2021.0	2020	2023	17.0	17	17
10	64.260827	10.0	102.0	6.0	1	12	6.260639	0.0	29.0	29.589504	23.21	30.43	67.250141	5.0	100.0	0.0955	0.0	1.830	3.0	0	6	2021.0	2020	2023	20.0	20	20
11	71.692486	17.0	111.0	6.0	1	12	7.743536	0.0	38.0	29.664629	22.86	30.59	56.591670	5.0	100.0	0.0880	0.0	1.830	3.0	0	6	2021.0	2020	2023	11.0	11	11
12	66.240950	12.0	108.0	6.0	1	12	7.290429	0.0	40.0	29.633978	22.60	30.47	63.517793	4.0	100.0	0.0610	0.0	1.810	3.0	0	6	2021.0	2020	2023	14.0	0	23
13	56.901670	16.0	91.0	7.0	1	12	4.527134	0.0	58.0	29.621507	23.35	30.48	74.969388	6.0	100.0	0.1135	0.0	1.829	4.0	0	6	2021.0	2020	2023	2.0	2	2
14	58.213660	1.0	109.0	8.0	1	12	4.709720	0.0	29.0	29.610852	23.29	30.39	74.645870	7.0	100.0	0.1045	0.0	1.800	3.0	0	6	2021.0	2020	2023	0.0	0	0
15	69.427757	24.0	103.0	7.0	1	12	7.790875	0.0	26.0	29.767643	26.10	30.35	66.353612	14.0	100.0	0.0000	0.0	1.793	2.0	0	6	2021.0	2020	2023	12.0	0	23
16	62.954300	8.0	97.0	7.0	1	12	5.419331	0.0	26.0	29.777373	24.03	30.44	75.023413	5.0	100.0	0.0100	0.0	1.809	3.0	0	6	2021.0	2020	2023	6.0	1	23
17	69.159626	13.0	113.0	6.0	1	12	7.837927	0.0	36.0	29.653764	23.13	30.49	59.818817	2.0	100.0	0.1070	0.0	1.828	3.0	0	6	2021.0	2020	2023	18.0	18	18
18	63.542138	8.0	93.0	6.0	1	12	5.442265	0.0	31.0	29.747690	22.79	30.60	74.350815	6.0	100.0	0.0210	0.0	1.831	2.0	0	6	2021.0	2020	2023	8.0	8	8
19	61.677575	11.0	97.0	7.0	1	12	5.285222	0.0	36.0	29.623930	23.13	30.52	70.859972	6.0	100.0	0.1200	0.0	1.830	3.0	0	6	2021.0	2020	2023	22.0	22	22

Again, we can see that for several Clusters (like 1,5,7,9), the Hour is just having value, which is showing that the accidents happening at those hours are tagged to those specific clusters

Next, I wanted to perform either of Agglomerative Clustering, or Spectral Clustering on the dataset, but it was taking a lot of time to execute the model on the 100K records. I waited for almost > 10 hours for each of them even with the default hyperparameter values. The process was not getting any quicker. I tried to sample with < 30K records, and was able to run Spectral Clustering, but that would not give a definite result to compare the different models, since the others were run on the larger dataset. Again, at this point, I did not feel I had to rerun all models against the smaller dataset, but rather research future strategies that deal with larger and higher dimension datasets.

I also did read about RAC Plus Plus Clustering, which might be a future learning to scope against. This is quicker version of the Agglomerative model.

7. *Conclusions, Future Scope and Lessons Learned:*

Adding data from above in a tabular format

Model	***Clusters***	***Silhouette***	***DB***	***CH***	***AIC***	***BIC***
KMeans	17	.085	2.9	2108.61	NA	NA
GMM	20	.079	2.75	2122.4	1.13e+07	1.1e+07

If we look at Silhouette as the measure, we could see that Kmeans did outperform. But looking at CH Score (higher is better) and DB (lower is better), we might get to say that GMM Clustering was better.

Since we do not have ground truth labels, it becomes exceedingly difficult to ascertain which model is better and what does even a 'better model' means.

Some learnings that I did have with this project which could be future scope items :

1. Working especially on clustering with high dimension data and huge record size, would need better computer power and processing power. And need research on better models that would work on this type of datasets (mixed data types, high dimension, lot of records)
2. Evaluate out several more models including non-linear ones like DBSCAN, Spectral Clustering, since many a times, the data would not exhibit a perfect linear relationship
3. Integrate with BI tools like Tableau using the additional attributes that we have like Latitude and Longitude to validate how the clustering looks on the map. There might be useful insights that could have been obtained, rather than from numerical or statistical analysis.
4. Deep dive more into the features to see if we could analyze the clusters in the data based on attributes like 'Accidents post Covid' or 'Accidents on evenings post 6 PM". This could be used as additional intrinsic measures to compare different models.

Lessons Learned:

I had also taken ISYE 6740 earlier. The main difference between that and this course is the practical experience and value add. This Data Mining course helped me understand how to approach data mining from multiple scenarios, perform EDA from all angles (not just feature selection) and then explore different models. Exploring just does not mean evaluating the model once. It involves hyperparameter tuning, feature selection , data sampling etc. And testing out multiple models. This

course helped me navigate though all those aspects and come out with a great amount of knowledge

Couple of areas of improvement:

- Include a Quiz 5 that has topics from NN and Unsupervised Learning. It gives an impetus to all to read those lessons.
- Also, some intro into NLP and LLM since that is gaining traction these days. Especially around preprocessing.

Thanks to the Team for giving us a great semester !

8. Appendix:

- Data Sources:
  - Link 1
  - Link 2
- Kmeans
  - From Scratch
- GMM
  - Basics
- Clustering Quality
  - Internal vs External measurements
- LGBM
  - Link 1
- SHAP Analysis
  - Link 1
- Mixed Data Analysis – FAMD
  - Link 1
  - Link 2