Team 2
Chicago Car Accidents Assessment Utilizing Clustering Analysis

1st Hodgetts, Michael
Electrical Engineering Dept
EE-695 - Machine Learning
Hoboken, NJ
mhodgetts@comcast.net

2nd Paladugu, Rithvika
Computer Engineering
EE 695 - Machine Learning
Hoboken, NJ
paldugurithvika@gmail.com

***Abstract*—Practice as a team to analyse unsupervised data by exploring various clustering methods (k-means, k-modes, kprototype, Gaussian and DBSCAN) on Chicago Accident database.**

I. Introduction

Utilizing two datasets from Chicago Department website (Crashes and People related to Crashes), we like to find areas/neighborhoods within the city that have different characteristics in terms of the attributes available. Dataset is large (750K rows) so we decided to look at only 2 years, i.e. 2021 and 2022 which reduces the row count to about 250K. There is a good amount of prep to get data suitable for cluster analysis (ETL, Hot Encoding, data cleanup) We tried 5 different cluster methods with varying results. We'll give some brief description of each method and its pros and cons for this particular data set. We will illustrate some of the main differences these assigned clusters present. Provide a summary of each cluster characteristics to illustrate the main differences between them.
Figure 1 illustrates Chicago Heatmap of Crashes by Zip code. Hade to use partial data since the original data had no zip for the crashes so had to use Nomatin to extract in python but there a limitation on how many you can search without further payment. You can see that the majority of the crashes are in the lower half of the city.

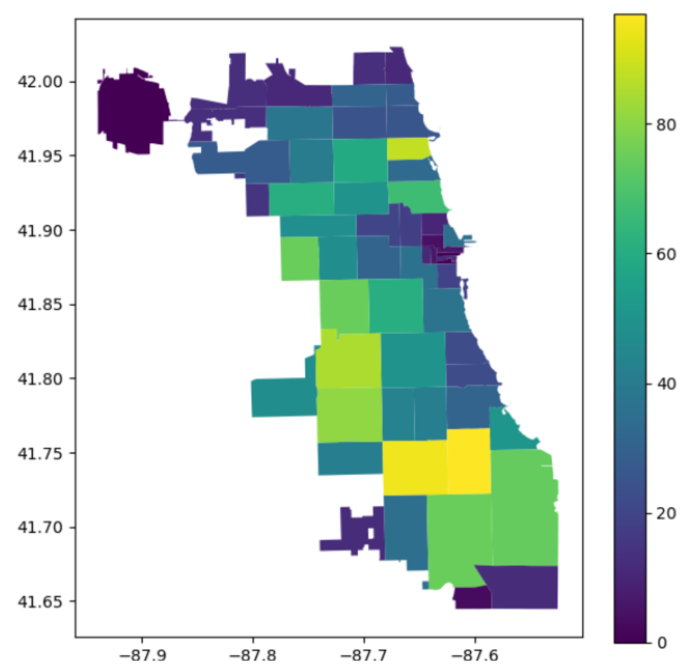

Fig. 1: Chicago Heatmap for Crashes by Zipcode

II. Related Work

There are many examples on how clustering solves problems in various business use cases. Segmentation analysis, anomoly detection and a form of classification are some of the use cases. There are also Deep Learning techniques that we could have explored but we ran out of time. The analysis here could be useful to identify hot areas of accidents due to poor signage, road conditions, and other conditions that could be improved to reduce accidents or injuries in the Chicago area. Practicing how to get meaningful insights is essential in the business world. Also, knowledge of mapping techniques or illustrating the clusters using coordinates or zip codes was useful. Our skills in this area are weak and had spent endless hours trying to learn shapely files and GEO methods to show how the car accidents distributed across the Chicago area. It seems you can have a semester class on mapping techniques alone.

III. Our Solution

Since the data is large and has many attributes (both numerical and categorical), reducing the number of attributes will be critical. Also, minimizing the number of clusters needed is beneficial to ensure each cluster has meaningful differences and similar volume. Created Heatmaps showing percent differences will be provided to help show major allocation of attributes to each cluster. k-modes was explored first since it handles categorical data and kmeans covers numerical. K-Prototypes covers both at the same time. We tried to use Gaussian Mixture but that seems to not be appropriate and gave us unusual results. Another method is DBSCAN which can handle hot encoding but so far find it difficult to gain the necessary clusters needed, very sensitive to (EPS) and size of data. kprototype so far has the best results with the mixed data. We will show Heatmaps to illustrate our results and provide summary description of the final clustering results. Significant code related to grouping data uing pspark to assess the cluster allocation was neccessary in evaluating key differences for each cluster. We also utilized Knime an ETL software to ensure the joining of data looked good, it sometimes harder to evaluate a join in python mode.

A. Description of Dataset

The dataset can be found here: Chicago Crashe Data, Two datasets; one for crashes in Chicago and the other are the people characteristics involved in those crashes. The 'Crash ID' is the key attribute to join the 2 datasets. We reduced the size to only years 2021 and 2022 which still leaves about 250K rows and 68 columns. We used Knime (ETL software) to help do this join and ensure it was properly achieved. The output of the Knime workflow is our starting point for the python code section. Here is the workflow using KNIME ETL software:

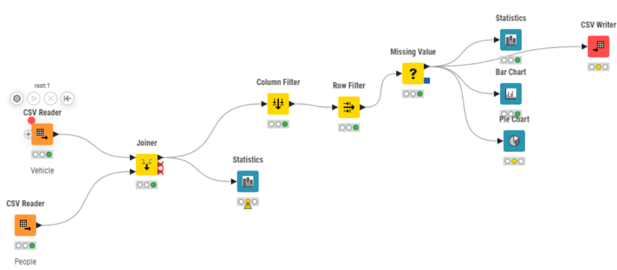

Fig. 2: Chicago Dataset Prep

Knime workflow takes in both car crashes file and people related to those crashes and join them using the crash id common attribute. We then filter redundant columns due to the join and evaluate which columns not relevant and impute mean data for missing values for those remaining.
Data had very few problems with missing data but we removed as necessary or added average values as needed. We separated the dataset into two parts, categorical and numerical. Performed Hot Encoding on the categorical for useon some of the algorithms and scaling on the numberical (0,1), then combined them back together as one file. New shape of the file is now 248K by 111 columns with hot encoding. We created another dataset for k-prototypes which has 31 columns based on what attributes we deemed valuable. Pie charts were useful in this particular dataset since most attributes are categorical. Many Attributes had too many categories that added more complexity so therefore we re-classify those less than 1 percent into one group called remainder to help reduce the number of hot columns needed.

Below is the list of columns we are working with in the dataset. Some have been removed since they will not contribute to the model performance.

```
df.columns

Index(['CRASH_RECORD_ID', 'RD_NO', 'CRASH_DATE', 'POSTED_SPEED_LIMIT',
       'TRAFFIC_CONTROL_DEVICE', 'DEVICE_CONDITION', 'WEATHER_CONDITION',
       'LIGHTING_CONDITION', 'FIRST_CRASH_TYPE', 'TRAFFICWAY_TYPE',
       'ALIGNMENT', 'ROADWAY_SURFACE_COND', 'ROAD_DEFECT', 'REPORT_TYPE',
       'CRASH_TYPE', 'INTERSECTION_RELATED_I', 'NOT_RIGHT_OF_WAY_I',
       'HIT_AND_RUN_I', 'DAMAGE', 'DATE_POLICE_NOTIFIED',
       'PRIM_CONTRIBUTORY_CAUSE', 'SEC_CONTRIBUTORY_CAUSE', 'STREET_NO',
       'STREET_DIRECTION', 'STREET_NAME', 'BEAT_OF_OCCURRENCE', 'NUM_UNITS',
       'MOST_SEVERE_INJURY', 'INJURIES_TOTAL', 'INJURIES_FATAL',
       'INJURIES_INCAPACITATING', 'INJURIES_NON_INCAPACITATING',
       'INJURIES_REPORTED_NOT_EVIDENT', 'INJURIES_NO_INDICATION',
       'INJURIES_UNKNOWN', 'CRASH_HOUR', 'CRASH_DAY_OF_WEEK', 'CRASH_MONTH',
       'LATITUDE', 'LONGITUDE', 'LOCATION', 'PERSON_ID', 'PERSON_TYPE',
       'CRASH_RECORD_ID (right)', 'RD_NO (right)', 'VEHICLE_ID',
       'CRASH_DATE (right)', 'CITY', 'STATE', 'ZIPCODE', 'SEX', 'AGE',
       'DRIVERS_LICENSE_STATE', 'DRIVERS_LICENSE_CLASS', 'SAFETY_EQUIPMENT',
       'AIRBAG_DEPLOYED', 'EJECTION', 'INJURY_CLASSIFICATION', 'HOSPITAL',
       'EMS_AGENCY', 'DRIVER_ACTION', 'DRIVER_VISION', 'PHYSICAL_CONDITION',
       'PEDPEDAL_ACTION', 'PEDPEDAL_VISIBILITY', 'PEDPEDAL_LOCATION',
       'BAC_RESULT', 'BAC_RESULT VALUE'],
      dtype='object')
```

Fig. 3: Chicago Dataset

EDA

Here we show a sampling of some of the attributes classes for the categorical data and numerical. There are many columns so we will just show a sample of some. You can see more in the code readout. More than half of the Top

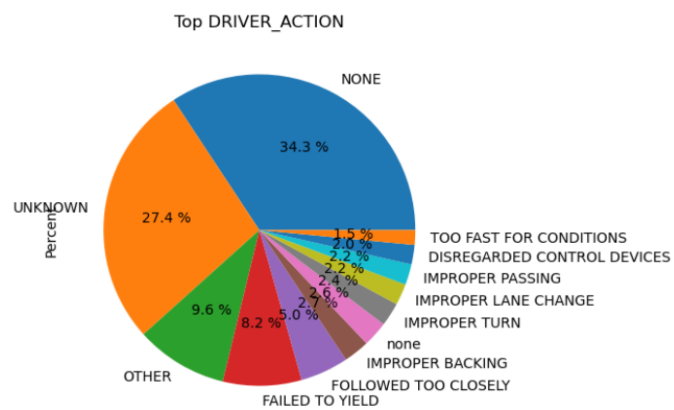

Fig. 4: Categorical Pie Charts (TOP Driver Reasons)

Driver clasifications are not known which hinders our evaluation
Majority of crashes where in clear weather
Almost 50 percent of crashes had some control place like a stop sign or traffic light
Here are some of the numeric attributes we examined
Couple of callouts for the numerical data is that crashes overindex in the winter months and there is some difference among which of the week seems to matter as well. Age shows a signficant amount around 39 years old, this seems

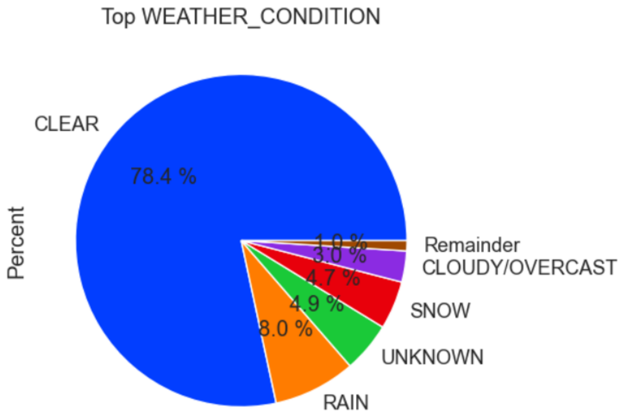

Fig. 5: Categorical Pie Charts (Sampling View)

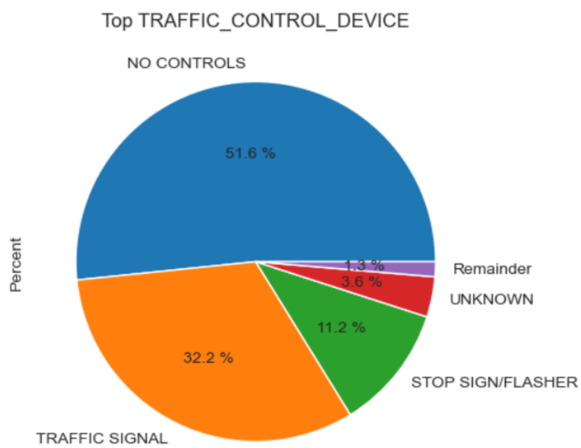

Fig. 6: Traffic Controlpie

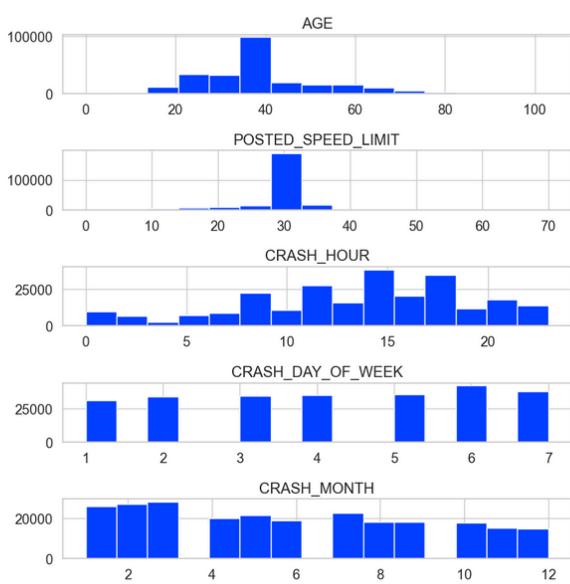

Fig. 7: Numeric Histograms (Sampling View)

like a possible data issue but in general the lower the age more likely to have car crash. Also, the Crash hour chart shows a peak around dinner time which seems like an expected result.

B. Machine Learning Algorithms

Our data is raw and has no classification or specific purpose so it lends itself to utilize unsupervised data techniques. We explored K-modes, kmeans, GMM, kprototype and DBSCAN and will investigate other possible methods to find insights. kprototype is our best hope so far to get good results since it handles both categorical and numeric attributes. Kmodes can only handle category and kmeans does numerical. We noticed Kprototype takes a very long time to process the model. Elbow curve took over 24 hours. Since we have a large dataset, training and other tasks take a good length of time for all model types. We created elbow curves for most of these technique. We'll discuss more in our solution section for each algorithm mentioned
We did some Chi Testing on the attributes to find any that were not relevate but all picked were significant. We utilized

	variable	chi2_test_stat	p_value	dof
34	INJURIES_UNKNOWN	0.000000e+00	1.000000e+00	0
41	PERSON_ID	1.243115e+06	4.985663e-01	1243110
67	BAC_RESULT VALUE	5.929951e+02	2.216402e-39	205
45	VEHICLE_ID	1.219355e+06	1.280826e-45	1197360
36	CRASH_DAY_OF_WEEK	8.620735e+02	5.757343e-162	30
...	...	...	...	...
30	INJURIES_INCAPACITATING	1.071638e+04	0.000000e+00	30
31	INJURIES_NON_INCAPACITATING	5.991988e+04	0.000000e+00	55
32	INJURIES_REPORTED_NOT_EVIDENT	2.847958e+04	0.000000e+00	40
33	INJURIES_NO_INDICATION	4.908476e+04	0.000000e+00	135
68	Cluster	1.243115e+06	0.000000e+00	25

Fig. 8: Chi Square Testing View

pspark to use groupby by clusters by percent of occurence to see what patterns emerge. We then exported this view to an excel so we can do further analysis to create a heatmap. We'll look at both the numberical and categorical and for our best solution we describe what each cluster overindexes on compared to the others

C. Implementation Details and Comparison

In this section we'll discuss the results of each method and show more in detail the best solution results. The table below shows a summary of the different clustering algorithms.
Table above illustrates how each cluster method can be applied. DBSCAN didn't seem suited for this dataset and it was very difficult to work with in terms of tuning parameters like EPS and Min samples. Doesn't seem to scale well with all this data. Kmodes and KPrototype we fairly close but KP had less error with both numberical and categorical combined to the only categorical of the kmodes.

Team 2 Model Summary

Method	Application	Model Training Speed	Cluster Volumes	Error	Results/Notes
kmodes	Categorical	7.5 minutes	1) 32.1% 2) 30.4% 3) 23.4% 4) 14%	1375000	kmodesand kPrototypes were similar in categorization but kmode had higher error
kmeans	Numerical	1.8 seconds	1) 25% 2) 25% 3) 25% 4) 25%	1 x10^15	equal distribution but can only be used for numerical
GMM	Both	5 seconds	1) 58% 2) 28.2% 3) 11.8% 4) 1.7%	AIC=-1.33x10^7	One cluster dominated in the distribution, results were not as good as KP
DBSCAN	Numerical	5.5 seconds	1) 70.3% 2) 17.5% 3) 6.3% 4) 6%		One cluster dominated in the distrubution and results didn't look reasonable
Kprototype	Both	51 minutes	1) 28.4% 2) 27.4% 3) 23.5% 4) 20.6%	128000	Best of All, Reasonable results but took much longer to train

Fig. 9: Model Summary

Kmodes

Kmodes is designed for categorical data only and it a can take the data without hot encoding. K-modes was successful and did a good job of clustering but you have to take out the numerical data and run that on kmeans. Therefore it becomes a 2 step process. The cluster allocation is similar to K-prototype and its error was also one of the lowest in term of SSE.

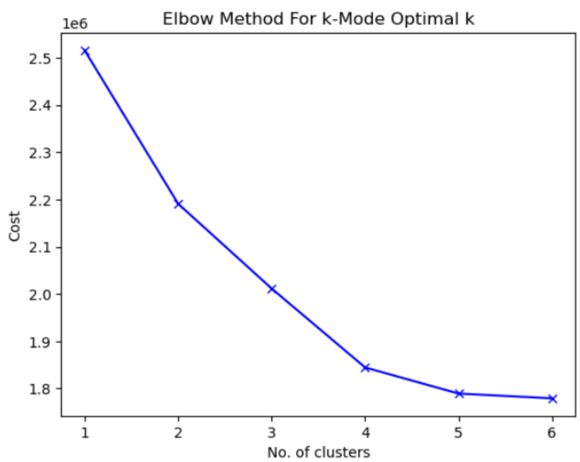

Fig. 10: k-mode

The elbow chart indicates that 3-4 clusters should be adequate
Dark Green represent more allocation vs dark red. Percent volume of each cluster is reasonable between 20-30 percent.

Kmeans

Kmeans above in the chart reflects the numerical data only, you can try to run the categorical data with hot encoding but it accuracy or ability to cluster diminishes.
Here is the numerical cluster view, We used mean and count to assess the numerical data
The numerical results were disapointing in that the shared volume across the clusters is similar for larger allocations but there are some minor differences between for the smaller insignificant occurences. Below we show an example for Posted Speed limit allocation for kmeans, you see very little distinction among the clusters.

GMM

Sum of perc_of_count_total	Column La			
Row Labels	0	1	2	3
ALIGNMENT	31.9268	13.95	30.3	23.27
Remainder	0.258	0.118	0.2292	0.4012
STRAIGHT AND LEVEL	31.2832	13.5764	29.6732	22.4868
STRAIGHT ON GRADE	0.3856	0.2544	0.4008	0.3824
CRASH_TYPE	31.9268	13.95	30.3	23.27
INJURY AND / OR TOW DUE TO CRASH	7.984	11.6444	2.982	9.1376
NO INJURY / DRIVE AWAY	23.9428	2.3044	27.3212	14.1328
DAMAGE	31.9268	13.95	30.3	23.27
$500 OR LESS	2.732	1.0352	3.1548	2.1832
$501 - $1,500	5.214	1.3432	14.7724	2.4812
OVER $1,500	23.9808	11.5704	12.376	18.606
DEVICE_CONDITION	31.9268	13.95	30.3	23.27
FUNCTIONING PROPERLY	24.2948	11.1012	3.0936	0.5492
NO CONTROLS	4.3516	1.6516	24.9776	21.536
Remainder	0.5676	0.3644	0.3508	0.206
UNKNOWN	2.7128	0.8316	1.8812	0.9792
FIRST_CRASH_TYPE	31.9268	13.95	30.3	23.27
ANGLE	3.24	6.1028	2.932	1.14
FIXED OBJECT	0.4016	0.4124	0.454	1.8216
HEAD ON	0.314	0.1876	0.2484	0.3488
PARKED MOTOR VEHICLE	0.5156	0.2144	2.8892	10.8832
PEDALCYCLIST	0.3612	0.4272	0.2928	0.34
PEDESTRIAN	0.4384	1.016	0.2604	0.6916
REAR END	9.3888	2.0904	11.1424	2.0952
REAR TO FRONT	0.5044	0.0536	0.8868	0.254
Remainder	0.338	0.1616	1.0448	0.6572
SIDESWIPE OPPOSITE DIRECTION	0.4224	0.096	0.7184	0.4076
SIDESWIPE SAME DIRECTION	5.7504	0.7216	6.9592	2.9696
TURNING	10.252	2.4652	2.4748	1.6616

Fig. 11: Kmode Cluster Allocation (Partial View)

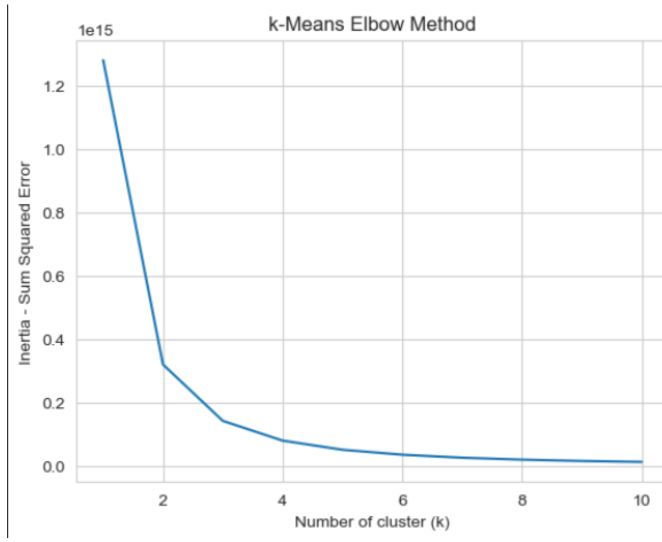

Fig. 12: kmeans

We used both numeric scaling and hot encoding here but we think that GMM is suited better with only numerical data. Here the cluster allocation was skewed towards one cluster that has the majority of volume. This was not our ideal solution. Here below is partial view of the cluster allocation. The first cluster is dominate in size with over 69 percent of the population.

The AIC and BIC are similar for all cluster values checked

DBSCAN

DBSCAN was very sensitive to the amount of data and columns. Only numerical data was used here and finding the EPS and min sample values was a challenge, Had to iterate through 50 different EPS values to find a sweet spot. By changing EPS by only 0.01 increments at that sweet spot changed clusters size and number of cluster significantly. The best EPS value for this particular dataset was 28.
We ran category data as well and found it very sensitive to the size of data and number of columns, could not run with

POSTED_SPEED_LIMIT	62388	62110	62120	62005	248623
0	110	112	114	109	445
1			2		2
2	4			2	6
3	19	20	15	18	72
4				1	1
5	193	162	163	132	650
7		5			5
9	1		2	2	5
10	1438	1295	1394	1242	5369
11			1	3	4
12				1	1
14			1		1
15	2065	1710	1812	1699	7286
20	2425	2408	2354	2395	9582
22			1		1
24	6	2	5	11	24
25	3816	3594	3555	3806	14771
29		2			2
30	46747	47290	47105	47040	188182
32	2			2	4
33				2	2
34	6		4	2	12
35	4441	4072	4209	4341	17063
38			2		2
39	8	5	7	3	23
40	672	773	777	631	2853
45	354	542	530	479	1905
50	14	51	12	35	112
55	63	57	53	49	222

Fig. 13: kmean example for Posted Speed Limit

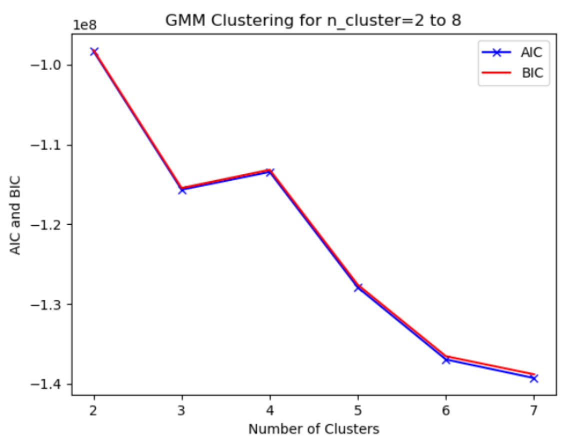

Fig. 14: GMM Elbow

the required column size. Had to reduce the size of the file to get the results below (Show partial view):

The results were not optimal and we saw again high allocation to one cluster. The -1 or noise cluster had most of the volume, I spent a consider amount of time to get this result, very sensitive to parameter settings. We don't think DBSCAN was useful in this application.

K-prototype

K-Protoype was our best soluton in that we can use all the data and train at the same time. This took significantly longer to train as you can see compared to others in the Comparison table. The volume allocation was similar to kmodes. So we can use kmeans for numerical and kmodes for categorical and combine them but KP does it together. We'll use KP to explain the main difference we see in the 4

Sum of perc_of_count_total	Column			
Row Labels	0	1	2	3
ALIGNMENT	**69.9264**	**6.234**	**5.924**	**17.3648**
Remainder	0.706	0.0756	0.0628	0.162
STRAIGHT AND LEVEL	68.2392	6.086	5.7756	16.9188
STRAIGHT ON GRADE	0.9812	0.0724	0.0856	0.284
CRASH_TYPE	**69.9264**	**6.234**	**5.924**	**17.3648**
INJURY AND / OR TOW DUE TO CRASH	22.4772	2.044	1.9196	5.3072
NO INJURY / DRIVE AWAY	47.4492	4.19	4.0044	12.0576
DAMAGE	**69.9264**	**6.234**	**5.924**	**17.3648**
$500 OR LESS	6.5468	0.5324	0.5108	1.5152
$501 - $1,500	16.9176	1.4428	1.398	4.0524
OVER $1,500	46.462	4.2588	4.0152	11.7972
DEVICE_CONDITION	**69.9264**	**6.234**	**5.924**	**17.3648**
FUNCTIONING PROPERLY	27.3164	2.5424	2.4064	6.7736
NO CONTROLS	37.2568	3.1744	3.0032	9.0824
Remainder	1.0532	0.0904	0.0928	0.2524
UNKNOWN	4.3	0.4268	0.4216	1.2564
FIRST_CRASH_TYPE	**69.9264**	**6.234**	**5.924**	**17.3648**
ANGLE	9.3232	0.922	0.8264	2.3432
FIXED OBJECT	2.1652	0.2068	0.196	0.5216
HEAD ON	0.778	0.0688	0.0624	0.1896
PARKED MOTOR VEHICLE	10.5436	0.8124	0.7348	2.4116
PEDALCYCLIST	1.128	0.0748	0.0528	0.1656
PEDESTRIAN	1.6504	0.1652	0.1904	0.4004
REAR END	17.298	1.5396	1.4956	4.3836
REAR TO FRONT	1.182	0.108	0.104	0.3048
Remainder	1.5448	0.146	0.12	0.3908
SIDESWIPE OPPOSITE DIRECTION	1.1624	0.108	0.0856	0.2884
SIDESWIPE SAME DIRECTION	11.4648	1.0072	1.0308	2.898
TURNING	11.686	1.0752	1.0252	3.0672

Fig. 15: GMM Cluster Results

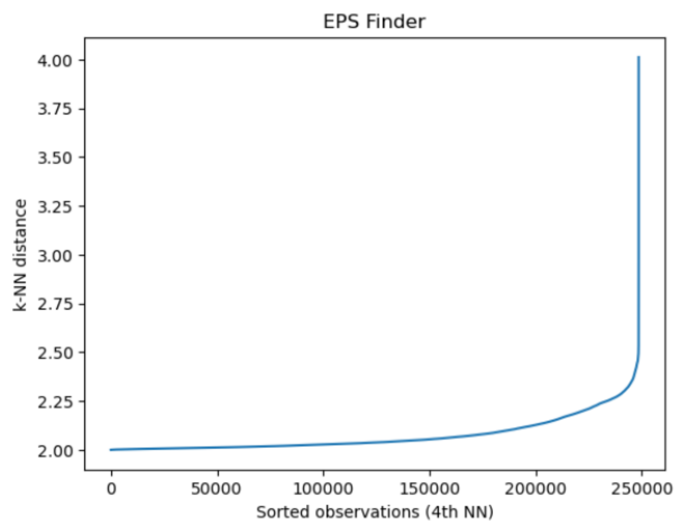

Fig. 16: DBSCAN EPS Finder

clusters we trained on. Also, we did not here of this type of clustering method before and we'll definetly use in the future.

linebreak

Cluster-Characteristics

Below is a partial view of the KP Heatmap for the 4 clusters

Key Cluster Findings

Cluster 1

-In general 1 has the lowest occurrence of crash attributes
-lowest occurrence of any crash damage level
-Volume is lowest
-unlikely for Hit and run cases
-first 1-3 day of week occurance

Sum of perc_of_count_total	Column Labels			
Row Labels	-1	0	1	2 (
ALIGNMENT	21.3076	18.4684	0.142	0.082
Remainder	0.3608	0.0396		
STRAIGHT AND LEVEL	20.4712	18.3384	0.142	0.082
STRAIGHT ON GRADE	0.4756	0.0904		
CRASH_TYPE	21.3076	18.4684	0.142	0.082
INJURY AND / OR TOW DUE TO CRASH	7.092	5.6008	0.002	8E-04
NO INJURY / DRIVE AWAY	14.2156	12.8676	0.141	0.081
DAMAGE	21.3076	18.4684	0.142	0.082
$500 OR LESS	2.7968	0.9624	0.002	
$501 - $1,500	5.6228	4.114	0.024	0.004
OVER $1,500	12.888	13.392	0.116	0.077
DEVICE_CONDITION	21.3076	18.4684	0.142	0.082
FUNCTIONING PROPERLY	9.1664	6.3196	4E-04	
NO CONTROLS	9.4284	12.0088	0.141	
Remainder	0.5308	0.0696		
UNKNOWN	2.182	0.0704	8E-04	0.082
FIRST_CRASH_TYPE	21.3076	18.4684	0.142	0.082
ANGLE	2.972	2.266		0.006
FIXED OBJECT	0.9584	0.3368		4E-04
HEAD ON	0.34	0.1392		
PARKED MOTOR VEHICLE	2.618	3.5136	0.138	0.004
PEDALCYCLIST	0.4468	0.1092		
PEDESTRIAN	0.7644	0.1216		
REAR END	4.5964	5.2964		0.043
REAR TO FRONT	0.5656	0.1424	8E-04	0.002
Remainder	0.6656	0.2484		4E-04
SIDESWIPE OPPOSITE DIRECTION	0.4608	0.2452	0.001	8E-04
SIDESWIPE SAME DIRECTION	3.1868	3.2452	0.003	0.022

Fig. 17: DBSCAN Cluster Result

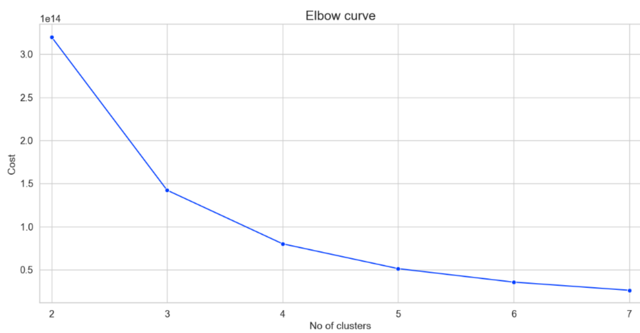

Fig. 18: k-prototype

Sum of perc_of_count_total	Column Labels								
Row Labels	0	1	2	3	Grand Total	Note			
ALIGNMENT	17.1056	26.1112	25.3148	30.9176	99.4492	min	max	major change	
Remainder	0.212	0.1844	0.2132	0.3968	1.0064	2	4	0.388354	
STRAIGHT AND LEVEL	16.6004	25.6088	24.6852	30.1252	97.0196	1	4	0.232093	
STRAIGHT ON GRADE	0.2932	0.318	0.4164	0.3956	1.4232	1	3	0.167097	
CRASH_TYPE	17.1056	26.1112	25.3148	30.9176	99.4492	1	4	0.230585	
INJURY AND / OR TOW DUE TO CRASH	5.0084	4.3912	16.3424	6.006	31.748	2	3	0.710969	xx
NO INJURY / DRIVE AWAY	12.0972	21.72	8.9724	24.9116	67.7012	3	4	0.449096	
DAMAGE	17.1056	26.1112	25.3148	30.9176	99.4492	1	4	0.230585	
$500 OR LESS	1.86	2.4584	1.8864	2.9004	9.1052	1	4	0.21936	
$501 - $1,500	4.25	7.0856	3.7328	8.7424	23.8108	3	4	0.398638	
OVER $1,500	10.9956	16.5672	19.6956	19.2748	66.5332	1	3	0.24084	
DEVICE_CONDITION	17.1056	26.1112	25.3148	30.9176	99.4492	1	4	0.230585	
FUNCTIONING PROPERLY	5.496	5.0384	21.6872	6.8172	39.0388	2	3	0.818401	xx
NO CONTROLS	10.3228	18.8628	1.8648	21.4664	52.5168	3	4	0.677187	xx
Remainder	0.2916	0.3168	0.4916	0.3888	1.4888	1	3	0.240796	
UNKNOWN	0.9952	1.8932	1.2712	2.2452	6.4048	1	4	0.356245	
FIRST_CRASH_TYPE	17.1056	26.1112	25.3148	30.9176	99.4492	1	4	0.230585	
ANGLE	2.054	2.2632	6.3104	2.7872	13.4148	1	3	0.594899	xx
FIXED OBJECT	0.6588	0.666	0.5636	1.2012	3.0896	3	4	0.375001	
HEAD ON	0.1988	0.266	0.316	0.318	1.0988	1	4	0.203957	
PARKED MOTOR VEHICLE	1.5084	10.1572	0.3744	2.4624	14.5024	3	2	1.223868	xxx
PEDALCYCLIST	0.2192	0.1828	0.6348	0.3844	1.4212	2	3	0.579652	
PEDESTRIAN	0.4892	0.4324	1.1116	0.3732	2.4064	4	3	0.570616	
REAR END	5.0976	3.3412	4.2936	11.9844	24.7168	2	4	0.637001	
REAR TO FRONT	0.3344	0.4976	0.1596	0.7072	1.6988	3	4	0.549776	
Remainder	0.538	0.5256	0.244	0.894	2.2016	3	4	0.483789	
SIDESWIPE OPPOSITE DIRECTION	0.344	0.5164	0.2248	0.5592	1.6444	3	4	0.377447	
SIDESWIPE SAME DIRECTION	3.1616	4.9012	2.3192	6.0188	16.4008	3	4	0.407513	
TURNING	2.5016	2.3616	8.7628	3.2276	16.8536	2	3	0.725445	xx
HIT_AND_RUN_I	17.1056	26.1112	25.3148	30.9176	99.4492	1	4	0.230585	
N	0.1872	0.4448	0.3756	0.3824	1.39	1	2	0.320322	
none	14.7576	6.5256	19.2136	26.7872	67.284	2	4	0.503631	
Y	2.1608	19.1408	5.7256	3.748	30.7752	1	2	1.009827	xx
INTERSECTION_RELATED_I	17.1056	26.1112	25.3148	30.9176	99.4492	1	4	0.230585	
N	0.2192	0.2308	0.4376	0.3072	1.1948	1	3	0.336432	
none	13.5644	23.7784	6.3476	28.1152	71.8056	3	4	0.548782	
Y	3.322	2.102	18.5296	2.4952	26.4488	2	3	1.204015	xx

Fig. 19: Example of Cluster Heatmap for Categorical Attributes

Cluster 2

-In general 2 has the second lowest occurance of crash attributes
-Likely to see Parked Vehicle Crash
-likely Hit and Run
-overindex on Sex X
-overindex on Sex F
-first 1-3 day of week occurrence

Cluster 3

-Overindex on Crash Injury
-Overindex turning related crash type
-Overindex on Interection related crash
-Overindex incapaciting injury
-Overindex disregarding traffic signals
-Overindex device condition functioning properly
-Overindex disregarding stop sign
-Overindex likely to report on scene
-likely to occur in the first 6 months of years

Cluster 4

-Volume is highest
-Overindex Parked Motor Vehicle
-Overindex on no injury
-Overindex no control for device condition (non traffic light)
-Overindex to be rear ended
-Overindex for follow to closely crash type
-Overindex for months 7-12

Fig. 20: Comparison of 4 Clusters

cluster 1 and 3 cluster 2 and 4

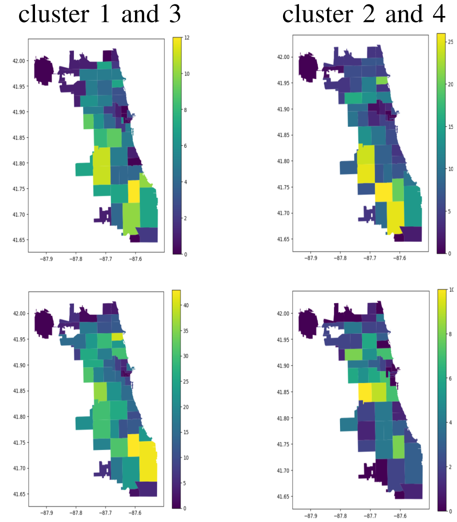

Figure 20 illustrates Cluster distribution by zip. There are differences among the 4 views. Cluster 1 seems more focused in the south central, cluster 2 focused in south west, Cluster 3 in south east and north and Cluster 4 is more spread out

IV. Future Directions

There are Deep Learning techniques that we noticed when we researching other methods but we needed more time to actually pursue it. One deep learning application was Unsupervised Deep Embedding for Clustering (DEC). There a need to better evaluate cluster information in terms of what is the cluster characteristics. A automated evaluation would be useful, we used percent allocation as a meausure but what other methods could be standardized. There another method for mixed data called Squeezer which can be used with mixed data but has little documentaion. Deep learning techniques utilizing autoencoders can be examined. Also, classification of the unsupervised data based on the 4 clusters. We can then do a multi-nominal logistic regression to better understand how the attributes contribute the most to a particular cluster. I find that interesting and would like to follow up on this.

V. Conclusions

Cluster analysis is powerful and easily understandable algorithm to utilize in unsupervised applications. The finding of k-Prototype Algorithm enables to evaluate both numeric and categorical data in one training session. Learning other methods like DBSCAN, GMM and kmodes was also beneficial to know for future applications. In terms of the Chicago data, there are some major difference between the clusters and Chicago dept of Transportation may be interested in such analysis to help prevent serious accidents or fix troublesome intersections or poor signing issues. Team learned a lot about the pros and cons of these methods

VI. References

1: https://scottmduda.medium.com/categorical-clustering-of-pittsburgh-car-accidents-using-k-modes-7c842cc15d87
2: https://data.cityofchicago.org/Transportation/Traffic-Crashes-Crashes/85ca-t3if
3: https://www.freecodecamp.org/news/8-clustering-algorithms-in-machine-learning-that-all-data-scientists-should-know/
4: https://scikit-learn.org/stable/auto$_{e}xamples/cluster/plot_{d}bscan.html$
$5 : https : //antonsruberts.github.io/kproto - audience/$
$6 : https : //geopandas.org/en/stable/docs/user_{g}uide/mapping.html$