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
palduguri thvika@gmail.com

3rd Nathan, Barry
dept. name of organization (of Aff.)
EE-695 - Machine Learning
Hoboken, NJ
nate.b4rry@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 (2021 and 2022) which reduces to about 250K. There is a good amount of prep to get data suitable for cluster analysis (ETL, Hot Encoding, data cleanup) We will try at least 3 different methods and compare the results. Since the data his mixed we have limited methods to do cluster analysis. We'll choose the better model and create an optimum number of clusters to analyze. 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. You'll see in this report we have already started this plan.

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. 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.

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. Heatmaps showing differences will be provided to help show major allocation of attributes to each cluster. Another issue is we have a lot of category data as well as continous. 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 noise (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.

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 of 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. 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 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 represented less than 1 percent of the size so we create algorithm to gather them up into a 'Remainder' for each category 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.

EDA

Here we show a sampling of some of the attributes classes for the categorical data and numerical. There are

```
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. 1. Chicago Dataset

many columns so we will just show a sample of some. You can see more in the code readout.

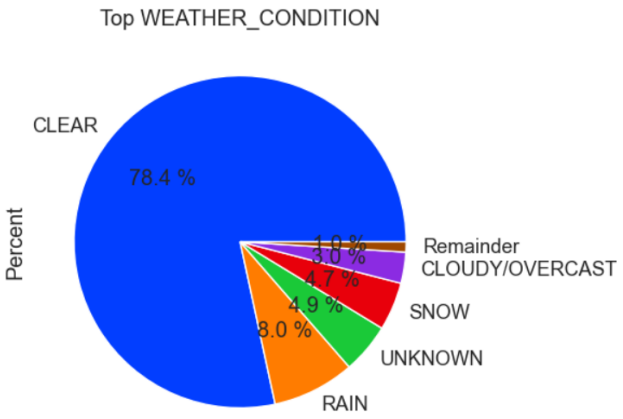
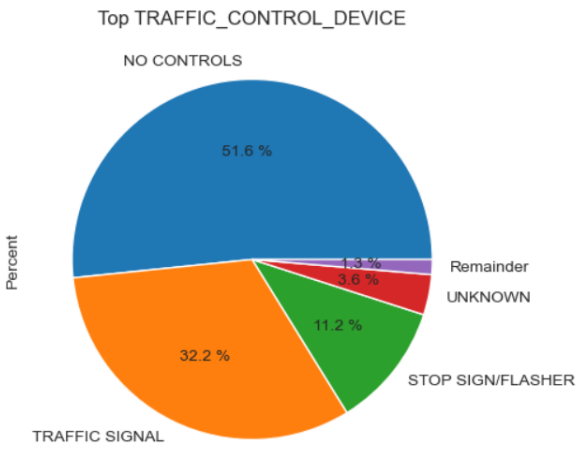


Fig. 2. Categorical Pie Charts (Sampling View)



Here are some of the numeric attributes to examine note:

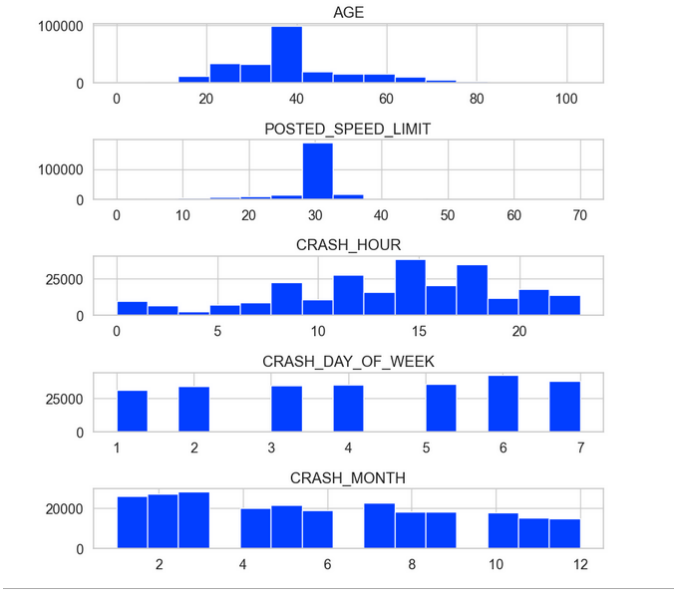


Fig. 4. Numeric Histograms (Sampling View)

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, 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. Noticed Kprototype takes a very long time to process the model. Elbow curve took over 24 hours. Since we have a large dataset, training take a good length of time for all model types. We created elbow curves for both K-modes and Kprototypes in the figures below: linebreak

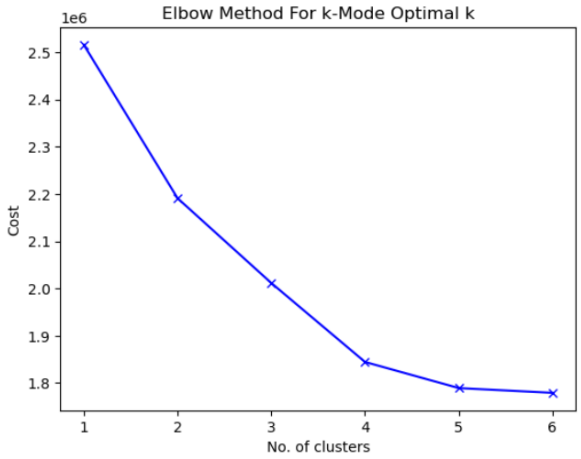


Fig. 5. k-means

From figures 2 and 3 you can see 4 clusters would be reasonable. We currently picked 4 clusters as a start. The

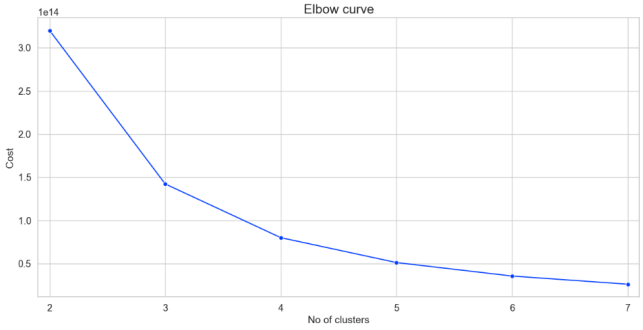


Fig. 6. k-prototype

difficulty is assessing the cluster characteristics with so many attributes. We did some Chi Testing on the attributes to find any that were not relevate but all picked were significant. We

	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. 7. Chi Square Testing View

uUtilized pspark to use groupby by clusters by percent of occurence to see what patterns emerge. We may have to build this view outside of Python to get a good illustrative view ith excel. At the current time, we're assessing how to decribe these these clusters and find recommendations to provide back to the city of Chicago. This figure below issustrates the comparison we are doing with 4 clusters verses the categorical columns. You can see clearly differences in of how the clustering divided up the percent allocation. Percent allocation is based total population so we can compare to each different attribute. Will do the same for the numerical but it will look a little different since we will use averages. 2

IV. Comparison

Gaussian clustered results didn't look correct since the majority of the volume was placed into one cluster, we think this is because it can't handle or interpret hot encoding or dummy variable too well. Kmodes and Kprototype gave similar results. report. Also DBSCAN was very senstive to EPS and Samples and could not achieve 4 clusters but only 3 with a majority of volume in one cluster alone. We will continue to examine if we can resolve this. We will add to this section in our final.

Sum of perc_of_count_total	Column Labels				
Row Labels	**0**	**1**	**2**	**3**	**Grand Total**
ALIGNMENT	**17.1056**	**26.1112**	**25.3148**	**30.9176**	**99.4492**
Remainder	0.212	0.1844	0.2132	0.3968	1.0064
STRAIGHT AND LEVEL	16.6004	25.6088	24.6852	30.1252	97.0196
STRAIGHT ON GRADE	0.2932	0.318	0.4164	0.3956	1.4232
CRASH_TYPE	**17.1056**	**26.1112**	**25.3148**	**30.9176**	**99.4492**
INJURY AND / OR TOW DUE TO CRASH	5.0084	4.3912	16.3424	6.006	31.748
NO INJURY / DRIVE AWAY	12.0972	21.72	8.9724	24.9116	67.7012
DAMAGE	**17.1056**	**26.1112**	**25.3148**	**30.9176**	**99.4492**
$500 OR LESS	1.86	2.4584	1.8864	2.9004	9.1052
$501 - $1,500	4.25	7.0856	3.7328	8.7424	23.8108
OVER $1,500	10.9956	16.5672	19.6956	19.2748	66.5332
DEVICE_CONDITION	**17.1056**	**26.1112**	**25.3148**	**30.9176**	**99.4492**
FUNCTIONING PROPERLY	5.496	5.0384	21.6872	6.8172	39.0388
NO CONTROLS	10.3228	18.8628	1.8648	21.4664	52.5168
Remainder	0.2916	0.3168	0.4916	0.3888	1.4888
UNKNOWN	0.9952	1.8932	1.2712	2.2452	6.4048
FIRST_CRASH_TYPE	**17.1056**	**26.1112**	**25.3148**	**30.9176**	**99.4492**
ANGLE	2.054	2.2632	6.3104	2.7872	13.4148
FIXED OBJECT	0.6588	0.666	0.5636	1.2012	3.0896
HEAD ON	0.1988	0.266	0.316	0.318	1.0988
PARKED MOTOR VEHICLE	1.5084	10.1572	0.3744	2.4624	14.5024
PEDALCYCLIST	0.2192	0.1828	0.6348	0.3844	1.4212
PEDESTRIAN	0.4892	0.4324	1.1116	0.3732	2.4064
REAR END	5.0976	3.3412	4.2936	11.9844	24.7168
REAR TO FRONT	0.3344	0.4976	0.1596	0.7072	1.6988
Remainder	0.538	0.5256	0.244	0.894	2.2016
SIDESWIPE OPPOSITE DIRECTION	0.344	0.5164	0.2248	0.5592	1.6444

Fig. 8. Example of Cluster Heatmap for Categorical Attributes

V. Future Directions

We still need to try other methods and see if the allocations are significantly different. We'll describe each cluster characteristics in more detail. We like to show so mapping features of how the clusters map over Chicago but find this difficult so far to accomplish. There another method call Squeezer which can be used with mixed data but has little documentaion. Deep learning techniques utilaing autoencoders can be examined. We'll add to this section for our final report

VI. Conclusions

We'll add our conclusions at the final report

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)$https : //antonsruberts.github.io/kproto - audience/$