Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

Washington Area Bicyclist Association Crash Reporting Data Analysis

The dataset comes from a survey conducted by the Washington Area Bicyclist Association (WABA). It has several free form text columns along with some categorical data and nominal data. “WABA-CrashTracker.xslx” contains survey response organized by supporter key and multiple rows for each key. Each row for a response key represents a question and corresponding response. For analysis, this data needs to be normalized or flattened to represent one row for each supporter key and multiple columns for each response. The xslx files has been saved to disc as a csv file with pipe delimiter. This was read into Python and flattened using “pivot_table” and saved as a new file with pipe delimiter.

Data

Below figure [1] shows the str output of the data. The data contains mostly text data types and below listed columns are nominal and categorical.

- Nominal columns: supporter, supporter_city, supporter_state and supporter_zip.
- Categorical columns: Age, Attorney_Yes_No, Citation_Yes_No, Compensation_Yes_No, Crash_Type, Cyclist_Statement_Yes_No, Follow_Up_Yes_No, Gender, Lighting, Police_Report_Yes_No, Police_Yes_No, Vehicle_Type, WABA_Member_Yes_No.

Rest of the columns are free text input columns where user was free to input what he wants.

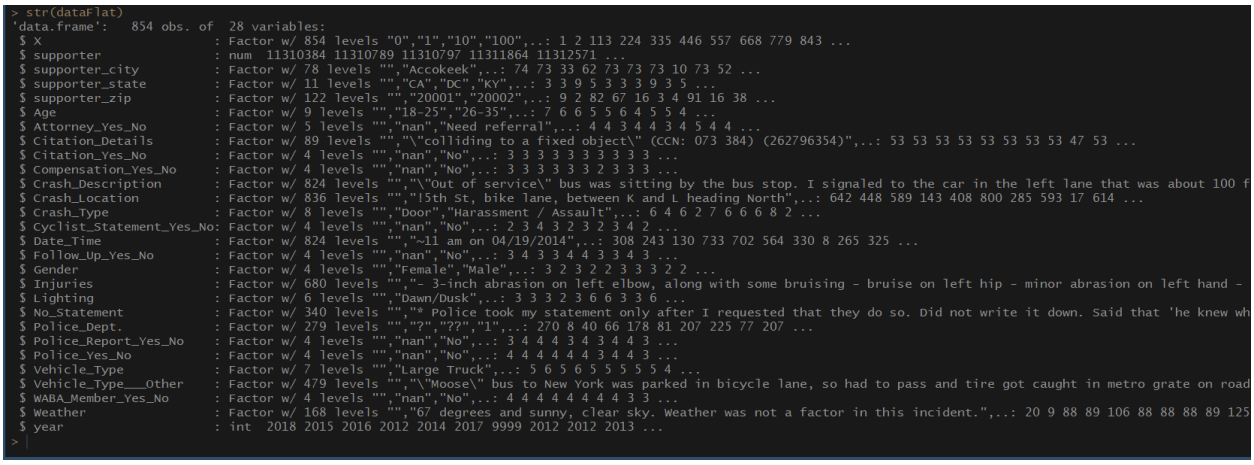

Figure [1]. Str output from R for the flattened dataset.

About WABA

`Washington Area Bicyclist Association (WABA) is a member-supported organization which advocates safe bicycling and promotes bicycling as fun, fitness and affordable way of transportation. WABA was founded on May 1, 1972 and started addressing the issues like bicycle parking, motorist and bicyclist education, theft and bike trails [1].

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

Bicycle Crash Reporting Survey – Why is it done

This survey is designed to collect data about bike accidents in Washington D.C. area in order to help WABA identify and address issues related to bicyclist's safety on the road. This could help involve local authorities, educate motorists and bicyclists about bike safety. Survey is not shared with police department and may be used and shared for advocacy purposes [2].

Dataset Issues

Privacy Concerns

Survey data was presented in xslx file. Even though the survey collected information about the supporter that was taking survey, the file provided only contains an id assigned to the supporter during the survey, supporter city, state and zip.

Data Quality

Informative data like crash location has been entered by user as free form text and is error prone when trying to parse to obtain structured data. Even though there are several categorical data types, most of them had an option to select "other" and input a free form text. Date and time of crash is also a free form text and some values do not have year and some have future dates.

Specific questions to be answered

This report explores and analyzes the dataset and attempts to answer below specific questions.

- Is there a change in police involvement over time since early reporting?
- Are there specific scenarios where police do not take report?
- Map the incidents reporting harassment.
- Use Natural Language Processing and classification techniques to classify if injuries are severe or not. Injuries is a free form text column.
- What time most crashes are occurring.

Hardware and Software requirements

- Hardware
    - Windows PC with Intel® Core™ i7-6700HQ CPU @ 2.60 GHz, RAM 16.0 GB and 64 bit operating system.
    - AIT580-VM-Student virtual machine with 4 CPUs and RAM 11392 MB
- Software
    - Anaconda 3 – Jupyter Notebook
    - Python 3
    - R Studio with R 3.5.1
    - Google Cloud API with Geocoding API enabled.
    - PostgreSQL version 1.22.0

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

Data Exploration

Looking at the categorical knowing how many factors are in the data will help with further data analysis and answering questions about the data. Age is a categorical variable with 7 levels as shown below. Similarly, gender has two. Lighting column which indicates the lighting conditions at the time of crash has 4 levels. Deleting "nan" in both the columns is to be done in the analyses that include these categorical columns.

```
> unique(dataFlat$Age)
[1] 65+      56-65    46-55    36-45    nan      26-35            18-25    U
nder 18
Levels:  18-25 26-35 36-45 46-55 56-65 65+ nan Under 18
> unique(dataFlat$Gender)
[1] Male   Female nan
Levels:  Female Male nan
> unique(dataFlat$Lighting)
[1] Full Daylight                Dawn/Dusk                    Night - Some st
reet lighting                          Night - No street lighting   nan
Levels:  Dawn/Dusk Full Daylight nan Night - No street lighting Night - Some
street lighting
```

Looking at the counts of few categorical variables in the dataset helps with establishing a few potential relationships. Figure [2] shows the counts of crashes by lighting which will answer the question – what time most of the crashes occur? Most of the crashes reported in the this dataset occurred during Full Daylight.

	Lighting	counts
0	Dawn/Dusk	76
1	Full Daylight	629
2	Night - No street lighting	9
3	Night - Some street lighting	106
4	nan	20

Figure [2]. Number of crashes by lighting conditions

Count of crashes in the dataset by age, as shown in figure [3], shows that the dataset had more crashes recorded for the age group 26-35. This does not indicate or conclude that this age group has a greater number of crashes. Looking at the number of crashes as a percentage of the whole group will give a better idea of most affected age group. This might be a case of more people from this age group taking the survey.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

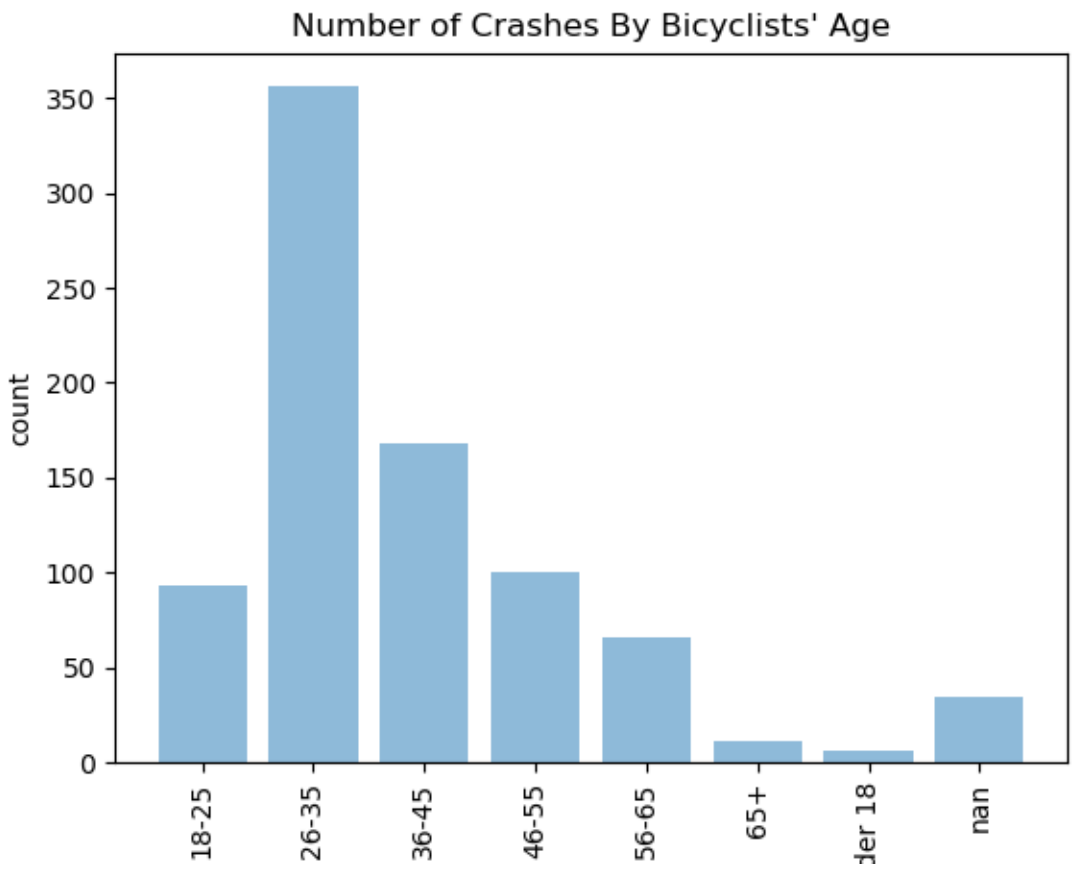

Figure [3]. Number of crashes by bicyclists' age group.

In order to answer the question, if police involvement changed since the early years of the survey, it is necessary to extract year from the free form text column Date_Time. Regular expression in Python is used to extract year. Below figure [4] shows the code snippet of the regular expression and figure [5] shows the plot of percent times police were present or took report by each year as a percent of total crashes reported each year.

Looking at just the counts seemed like police involvement increased over years but when it is looked at as a percentage of total number of crashes, as it can be seen from figure [5], there is not much change over years. 2006 with 100% involvement had one crash reported and that involved police.

There is no obvious scenario when police were present but did not take report. For further analysis, if age and gender take any role in this, Chi-Square test is performed. A series of binary variables are derived as

- If police were present but did not take report
- Bicyclist is above 35 or not

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

```
In [127]: import re

def getYear(DateText):
    x=re.search("20\d{2}",DateText)
    yearT="9999"
    if x:
        yearT =  x.group()
    else:
        #x1 = re.search(r"([1-9] |[0-9]| 2[0-9]|3[0-1])(.|-|/)([1-9] |1[0-2])(.|-|/)[0-9][0-9]", DateText)
        x1 = re.search(r"(\d{1,2})(.|-|/)(\d{1,2})(.|-|/)[0-9][0-9]", DateText)
        if x1:
            yearT =   "20" + x1.group()[-2:]
        else:
            x2 = re.search(r"([1-9] |1[0-2])(.|-|/)[0-9][0-9]", DateText )
            if x2:
                yearT = "20" + x2.group()[-2:]
            else:
                x3 = re.search(r"([1-9] |[0-9]| 2[0-9]|3[0-1])(.|-|/)[a-z]{3,9}(.|-|/)[0-9][0-9]", DateText, re.IGNORECASE)
                if x3:
                    yearT = "20" + x3.group()[-2:]
                else:
                    x4 = re.search(r"[a-z]{3,9}(.|-|/)[0-9][0-9]", DateText, re.IGNORECASE)
                    if x4:
                        yearT = "20" + x4.group()[-2:]

    return yearT
datet="October/2018 4:00pm"
getYear(datet)

Out[127]: '2018'
```

Figure [4]. Python regex to extract year from the free form column Date_Time

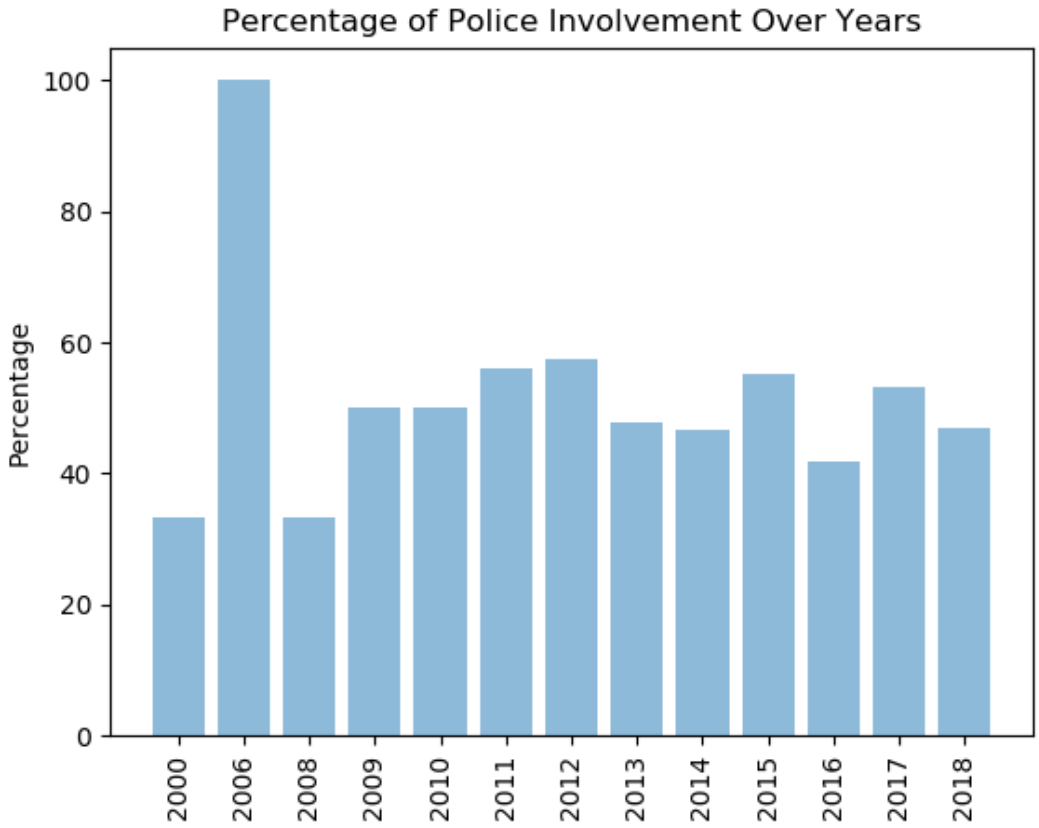

Figure [5]. Police involvement as percentage of crashes reported each year.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

Chi-Square Test

With the binary variables generated, Chi-Square test is conducted. Null hypotheses for each test are

- There is no relation between being above 35 and police not taking report.
- There is no relation between being male or female and police not taking report.

As it can be seen from the results below in figure [6], null hypothesis is accepted in both the cases and it can be concluded that being male/female and being above 35 years old is not correlated to police taking report or not.

```
> table(chisqDataAge$ageAbove35, chisqDataAge$policeRportTaken )

     0   1
  0  29 205
  1  28 164
> chisq.test(chisqDataAge$ageAbove35, chisqDataAge$policeRportTaken, correct = FALSE)

        Pearson's Chi-squared test

data:  chisqDataAge$ageAbove35 and chisqDataAge$policeRportTaken
X-squared = 0.4365, df = 1, p-value = 0.5088

> table(chisqDataGen$Gender, chisqDataGen$policeRportTaken )

           0   1
           0   0
  Female  19 144
  Male    38 228
  nan      0   0
> chisq.test(chisqDataGen$Gender, chisqDataGen$policeRportTaken, correct = FALSE)

        Pearson's Chi-squared test

data:  chisqDataGen$Gender and chisqDataGen$policeRportTaken
X-squared = 0.60643, df = 1, p-value = 0.4361
```

Figure [6]. Chi-Square tests and results.

Mapping Harassment Locations

Location column in the dataset is a free form text where bicyclists entered locations of intersections, landmarks etc. In order to parse this information, API provided by Google Maps Platform has been used. I have signed up for free trial at https://cloud.google.com/maps-platform/ after which it provides Geocoding API key. This key is used in the Python code to call the API and parsed the returned JSON to get latitude and longitude. Bokeh library in Python provides GMapOptions and gmap to point the locations on a google map [3]. Figure [7] shows the code snippet which has API key masked for security. Harassment locations are shown in the figure [8] with zoomed in on the dense area.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

In [224]:
```
import  requests
api_key=


def retrAdd(addresst):

    geocode_url = "https://maps.googleapis.com/maps/api/geocode/json?address={}".format(addresst)
    if api_key is not None:
        geocode_url = geocode_url + "&key={}".format(api_key)

    # Ping google for the reuslts:
        results = requests.get(geocode_url)
    # Convert JSON to dictionary
        results = results.json()
        if len(results['results']) == 0:
            return ""

        lat = float(results['results'][0]['geometry']['location']['lat'])
        lng = float(results['results'][0]['geometry']['location']['lng'])
        return [lat, lng]
```

In [225]:
```
addressr = "11th street NW between U St and Vermont Ave"
retrAdd(addressr)
```

Out[225]: [38.91684, -77.02512899999999]

In [239]:
```
from bokeh.io import output_file, show
from bokeh.models import ColumnDataSource, GMapOptions
from bokeh.plotting import gmap
output_file("gmap.html")
map_options = GMapOptions(lat=38.8450, lng=-77.1232, map_type="roadmap", zoom=12)
p = gmap(api_key, map_options, title="WABA")

source = ColumnDataSource(
    data=dict(lat=lat_l,
              lon=lng_l)
)

p.circle(x="lon", y="lat", size=15, fill_color="red", fill_alpha=0.8, source=source)
show(p)
```

Figure [7]. Geocode API and Bokeh for parsing location and placing on a map.

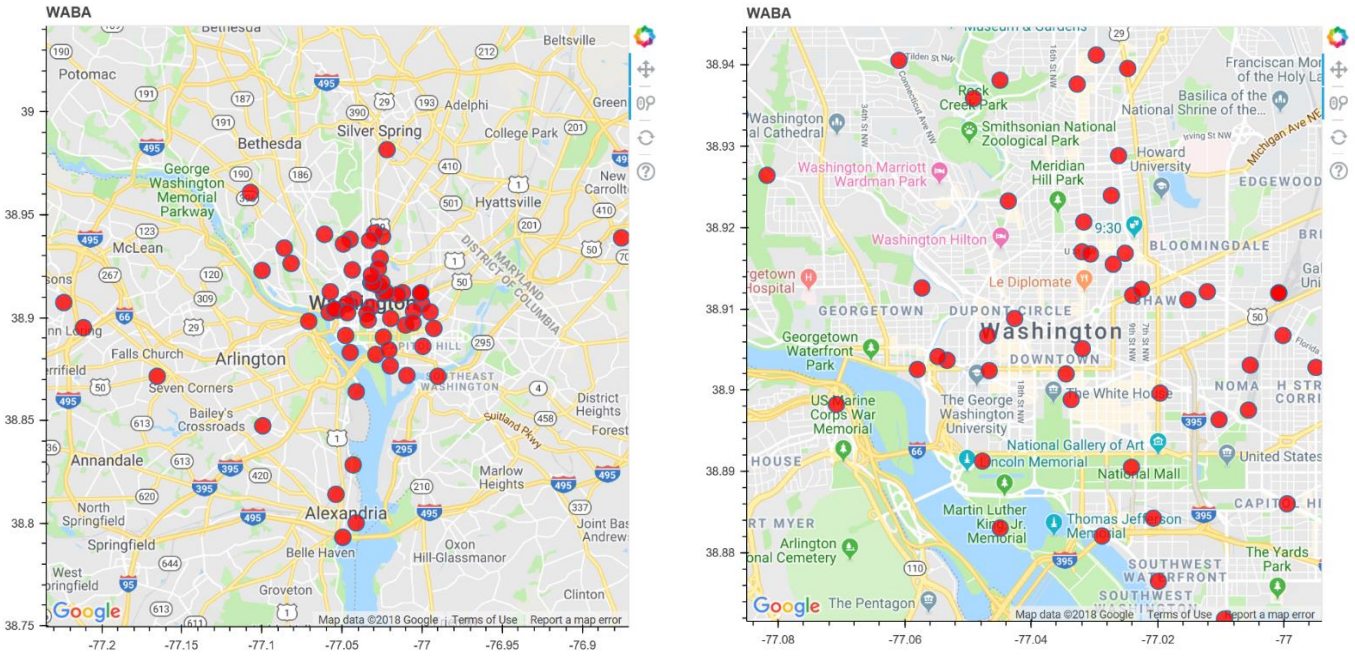

Figure [8]. Harassment Locations on map, zoomed in on the dense are in the left is on the right.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

SQL on PostgreSQL

Flattened data has been imported into PostgreSQL database which was provided in the virtual machine AIT580-VM-Student. Table name is waba_flatten_survey and structure and series of queries with results are show in below figures from [9] and [10].

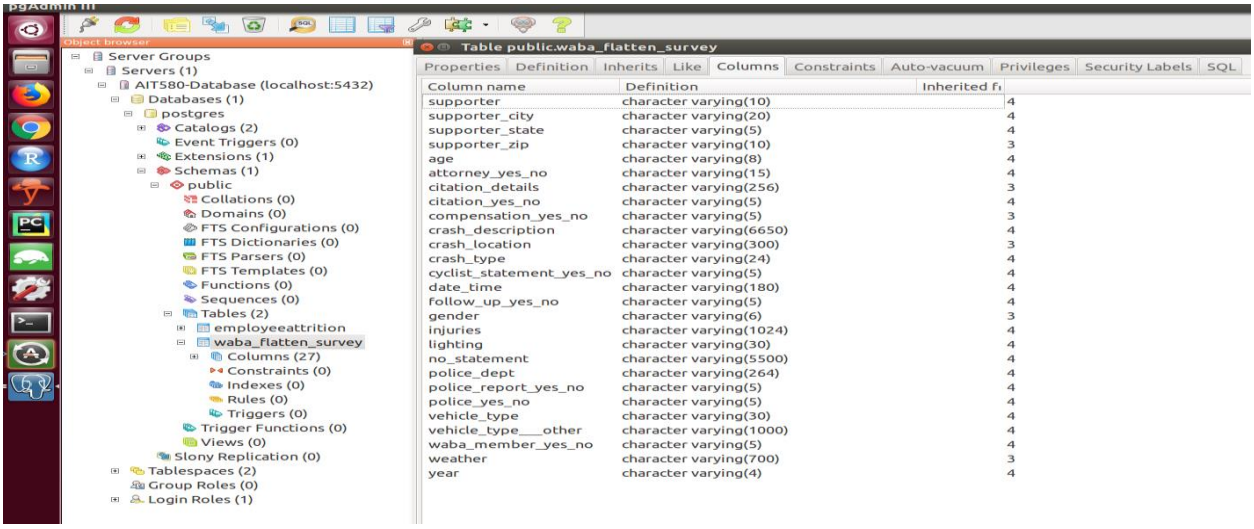

Figure [9]. PostgreSQL table structure

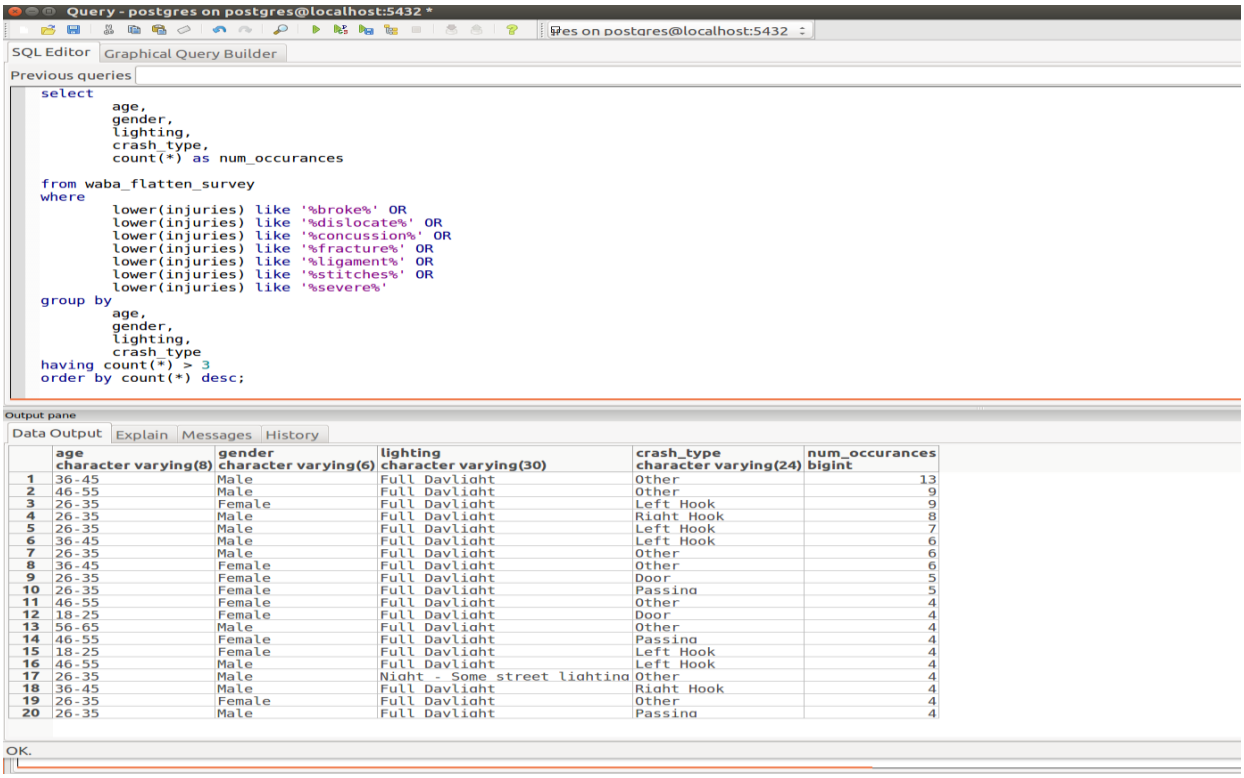

Figure [10]. Identify severe injury groups by age, gender, lighting and crash type.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

Natural Language Processing and Classification of Injuries

As the column injury is a free form text and for any analysis based on injuries involved in the crash, it is helpful to classify if injuries are severe or not. In order to train the classifier, I have created a csv file which has injury description and severity indicator high/low with 110 randomly selected survey responses to the question Injury. This file has been used to train and test the nltk "NavieBayesClassifier". Below figures [11] and [12] shows the code along with features selected by the model, results and classifier accuracy.

In [11]:
```
import nltk
nltk.download('punkt')
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
nltk.download('stopwords')
stopWords = set(stopwords.words('english'))

def format_sentence(sent):
    return({word: True for word in nltk.word_tokenize(sent) if word not in stopWords  })

print(format_sentence("The cat is very cute"))
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Lokesh\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Lokesh\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
{'The': True, 'cat': True, 'cute': True}
```

In [12]:
```
with open('smpleInjClassif.csv', 'rb') as f:
    result = chardet.detect(f.read())
rawSampData = pd.read_csv("smpleInjClassif.csv", sep = "|",  encoding = result['encoding'])
rawSampData.describe()
```

Out[12]:

	sevInd	injDesc
count	110	110
unique	2	106
top	minor	None
freq	72	5

Figure [11]. NLTK remove stop words and read in train/test data set

The features chosen seems to be ok except the word "The" but the level of confidence over minor doesn't seem to be good. Testing the classifier with the string "Broke my leg multiple stitches" classified it correctly as a major injury. The accuracy of the model is 0.48 which is not good. Further tuning with proper train/test dataset and passing selective features might improve performance of the classifier.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

```
In [13]: majorT = rawSampData[rawSampData.sevInd == "major"].injDesc.tolist()
         minorT = rawSampData[rawSampData.sevInd == "minor"].injDesc.tolist()
         major = []
         minor = []
         for i in majorT:
                 major.append([format_sentence(i), 'major'])
         for i in minorT:
                 minor.append([format_sentence(i), 'minor'])

In [14]: training = major[:int((.8)*len(major))] + minor[:int((.8)*len(minor))]
         test = major[int((.8)*len(major)):] + minor[int((.8)*len(minor)):]

In [15]: from nltk.classify import NaiveBayesClassifier
         classifier = NaiveBayesClassifier.train(training)

In [16]: classifier.show_most_informative_features()

         Most Informative Features
                             road = True            major : minor  =      6.9 : 1.0
                             rash = True            major : minor  =      5.6 : 1.0
                       concussion = True            major : minor  =      5.6 : 1.0
                            hands = True            major : minor  =      5.6 : 1.0
                             deep = True            major : minor  =      4.4 : 1.0
                             arms = True            major : minor  =      4.4 : 1.0
                           impact = True            major : minor  =      4.4 : 1.0
                             bone = True            major : minor  =      4.4 : 1.0
                            taken = True            major : minor  =      4.4 : 1.0
                              The = True            major : minor  =      3.4 : 1.0

In [17]: example1 = "Broke my leg multiple stitches"

         print(classifier.classify(format_sentence(example1)))

         major

In [18]: from nltk.classify.util import accuracy
         print(accuracy(classifier, test))

         0.4782608695652174
```

Figure [12]. Train the model and test classifier with NaiveBayesClassifier in nltk library.

Summary

The dataset has categorical data with option of “Other” and free from text responses. Regular expression to extract year from free from Date_Time column, visualization with groups, Chi-Square test and Google API to search places helped parsing the free form text data and answering the questions below.

Is there a change in police involvement over time since early reporting?

The percentage of police involvement over years does not show any increase or decreasing trend. As seen in figure [5], it pretty much remained below 60%. Data parsing of Date_Time was done using regular expressions in Python.

Are there specific scenarios where police do not take report?

Chi-Square tests shown in figure [6], performed on variables above 35 years old and being male/female, proven to be not affecting if a police officer takes report if one present.

Map the incidents reporting harassment

Figure [8] shows the locations where harassments were reported. These maps were generated using Google Maps Platform and Bokeh plotting on Google Maps.

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

Use Natural Language Processing and classification techniques to classify if injuries are severe or not. Injuries is a free form text column.

NaiveBayesClassifier from natural language toolkit nltk has been used to classify injury free form text and accuracy was very poor at 0.48. This model needs to be tuned with changing features set and selecting better training/testing data set.

What time most crashes are occurring

As shown in table in figure [2], the greatest number of crashes occurred in full daylight.

References

[1]. "WABA History (1972-1992) | Cycling Infrastructure | Washington". *Scribd*, 2018, https://www.scribd.com/doc/218463126/WABA-History-1972-1992. Accessed 9 Dec 2018.

[2]. County, Action et al. "Washington Area Bicyclist Association". *Washington Area Bicyclist Association*, 2018, https://www.waba.org/. Accessed 9 Dec 2018.

[3]. contributors, Bokeh. "Welcome To Bokeh — Bokeh 1.0.2 Documentation". *Bokeh.Pydata.Org*, 2018, https://bokeh.pydata.org/en/latest/. Accessed 9 Dec 2018.

Appendix

Survey Link to understand the data better:

https://org.salsalabs.com/o/451/p/salsa/web/questionnaire/public/?questionnaire_KEY=1200

Data Dictionary with explanation of categorical questions

'supporter',

'supporter_city',

'supporter_state',

'supporter_zip',

'Age',

'Attorney_Yes_No',

Have you retained an attorney?

Attorney Yes/No

'Citation_Details',

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

'Citation_Yes_No',

Were you issued a citation for an infraction?

Citation Yes/No

'Compensation_Yes_No',

If you were injured, did you receive compensation for your injuries?

Compensation Yes/No

'Crash_Description',

'Crash_Location',

'Crash_Type',

Door - Parked/idling motor vehicle door opened into path of cyclist

Right Hook - Motorist made right turn from adjacent lane across path of cyclist traveling straight

Left Hook - Motorist made left turn across path of cyclist traveling straight

Passing - Motor vehicle struck cyclist from behind or sideswiped while passing

Harassment / Assault - Motorist intentionally distracted, intimated, harassed or assaulted you

Other - All other crash types

*If you are unsure, please choose "Other"

'Cyclist_Statement_Yes_No',

If a police report was taken, did the officer record your statement?

Cyclist Statement Yes/No

'Date_Time',

Please provide the date & time of the crash.

Date/Time

'Follow_Up_Yes_No',

'Gender',

'Injuries',

'Lighting',

What was the lighting at the time of the crash?

Lokesh Merupula

AIT-580, Fall 2018

Prof. Harry J. Foxwell

LIGHTING

Full Daylight Dawn/Dusk Night - No street lighting Night - Some street lighting

'No_Statement',

'Police_Dept.',

'Police_Report_Yes_No',

Was a police report taken?

Police Report Yes/No

'Police_Yes_No',

Were the police called to the scene of the crash?

Police Yes/No

'Vehicle_Type',

'Vehicle_Type___Other',

'WABA_Member_Yes_No',

'Weather'