Denver Crime

Radjamin Hukom, rhukom@bellarmine.edu

ABSTRACT

This paper seeks to understand the factors surrounding crime in Denver that is available to be analyzed in the Denver Crime data. Through this investigation, the paper seeks to be able to understand whether certain factors correlate with crime or not.

I. INTRODUCTION

Using the database of Denver crime acquired at kaggle, I want to investigate what are the different factors that affect the type of crime that is being done in Denver through logistic regression machine learning. Some of the factors that I would be looking into would be the location of the crime, the crime_id, offense_id, time, among others.

The dependent variable would mainly be the type of crime that was perpetrated, meaning, it would be a logistic regression as it is a discrete variable. Other possible outputs would be the district or precinct id showing whether certain types of crimes or other factors affect where the crime would be perpetrated.

Some interesting parts of this project would be (hopefully) a map that shows where the crime was perpetrated, a timetable showing where crime often occurred, and when it is often reported among other things. This is a big dataset.

II. BACKGROUND

The data set documents criminal offenses in the city and county of Denver. It is a data set that is continually renewed. However, the data set is not whole as certain information is omitted due to legal requirements. The data is not without errors as it relies on data that cannot always be verified. The data set omitted sexual assault, child abuse, juvenile perpetrators, victims, and witnesses of certain crimes due to legal restrictions.

It is not stated explicitly in the data card why this information is collected other than for documentation purposes. It is also not clear what potentially useful information in the data set that is being sought by the city of Denver. However, by looking at the data set, a clear directive to investigate which factors correlating with (or even causing) crime seems obvious.

III. EXPLORATORY ANALYSIS

This data set contains 386865 rows and 20 columns with various data types.

Table 1: Data Types

Variable Name	*Data Type*	*Additional Notes*
Incident_id	Float64	Not usable in logistic regression due to large numbers
Offense_id	Float64	Not usable in logistic regression due to large numbers
Offense_code	Int64	Not usable in logistic regression due to large numbers
Offense_code_extension	Int64	
Offense_type_id	Object (string)	A further elaboration on offense_category_id
Offense_category_id	Object (string)	
First_occurrence_date	Object (string → datetime64[ns])	
Last_occurrence_date	Object (string → datetime64[ns])	175556 null values
Reported_date	Object (string → datetime64[ns])	
Incident_address	Object (string)	
Geo_x	Float64	15503 null values
Geo_y	Float64	15503 null values
Geo_lon	Float64	15769 null values
Geo_lat	Float64	15769 null values

District_id	Object (string)	57 null values
Precinct_id	Int64	
Neighborhood_id	Object (string)	689 null values
Is_crime	Int64 (binary)	
Is_traffic	Int64 (binary)	
Victim_count	Int64 (binary)	

IV. METHODS

A. Data Preparation

First, due to programming language issues, I needed additional code to enable the program to be able to read the dataset. Then, to enable to make the visualizations that I later needed, I converted the date columns into datetime64[ns] from string to make the AM and PM also show up.

Then I dropped 'incident_address', 'is_crime', 'is_traffic', 'victim_count'. I wanted to drop as few columns aas possible in the beginning. I dropped the first one because I saw that there were too many variations in that column. The last three I dropped due to is_crime being all 1s, is_traffic being all 0s, and victim_count being all 1s.

To eliminate the null values, I started with district_id. I noticed that almost all was using integer values except U which I presumed to denote unknown which I converted to 0 to be able to convert district_id to int from string. I decided to drop neighborhood_id due to there being 78 different neighborhoods. Then, I separate first_occurrence_date into fid, fid_time, and fid_part, which denotes the date, the time, and the AM or PM of the crime. Then, I did the same for last_occurrence_date (lod), and reported_date (rd). After rearranging the columns, I decided to drop last_occurrence_date as I don't want to remove that big of a chunk of my data and I also don't want the hassle to clean it, there's no good value to put in for empty values as there's not enough knowledge available for me to know what the empty values meant. Tried researching it, can't come up with a good answer. Wanted to put current date time, but then need a function to continually update to current time, which doesn't really makes sense. I decided to drop precinct_id to there being too many unique inputs in that column.

Now, I'm left with geo locator columns. I tried to make a program to estimate the geo locators using the district_id or precinct_id. after the upteenth time of attempting to fix this 'bug' where it clearly says that the boolean function outputs True so it should do what I asked it to do, but it clearly didn't when I checked so I decided to dropna.

Then I made fid_day and rd_day, new columns that tell the day of the week the crime took place. For logistic regression, I converted offense_category_id into offense_category_num which is the integer version of the same information.

B. Experimental Design

```
x = ds[['incident_id', 'offense_id', 'offense_code_extension', 'geo_x', 'geo_y',
    'geo_lon', 'geo_lat', 'district_id']]
y = ds[['offense_category_num']]
```

Table 2: Experiment Parameters

Experiment Number	Parameters
1	All 8 raw features with 66/16.5/16.5 split for train, validate, and test
2	All 8 raw features with 80/10/10 split for train, validate, and test
3	All 8 raw features with 50/25/25 split for train, validate, and test

```
x= ds[['incident_id', 'offense_id', 'offense_code_extension','offense_category_num', 'geo_x', 'geo_y',
    'geo_lon', 'geo_lat']]
y= ds[['district_id']]
```

Table 3: Experiment Parameters

Experiment Number	Parameters
4	All 8 raw features with 66/16.5/16.5 split for train, validate, and test
5	All 8 raw features with 80/10/10 split for train, validate, and test
6	All 8 raw features with 50/25/25 split for train, validate, and test

C. Tools Used

The following tools were used for this analysis: Python v3.5.2 running the Anaconda 4.3.22 environment for Asus computer was used for all analysis and implementation. In addition to base Python, the following libraries were also

used: Pandas 0.18.1, Numpy 1.11.3, Matplotlib 1.5.3, Seaborn 0.7.1, SKLearn 0.18.1, datetime, and warnings. Provide a brief explanation of why you chose these tools.

All the ones listed above were tools that was used in class and were ones that I was familiar with in usage due to extensive tutelage under Dr. Sarkar. The datetime and warnings were used for convenience and to speed up the process as converting certain columns (like the date ones) into datetime format would make it easier for data manipulation and analysis while the warnings allow for me to tell the computer to forgo printing the warnings, meaning, faster and easier to read outputs.

CI. RESULTS

a. Classification Measures/ Accuracy measure

Figure 1: Experiment 1

	precision	recall	f1-score	support
0	0.00	0.00	0.00	4675
1	0.00	0.00	0.00	15295
2	0.00	0.00	0.00	299
3	0.00	0.00	0.00	18438
4	0.00	0.00	0.00	9196
5	0.00	0.00	0.00	7210
6	0.00	0.00	0.00	18477
7	0.00	0.00	0.00	144
8	0.00	0.00	0.00	4821
9	0.00	0.00	0.00	18223
10	0.00	0.00	0.00	2231
11	0.17	1.00	0.30	21259
12	0.00	0.00	0.00	2194
accuracy			0.17	122462
macro avg	0.01	0.08	0.02	122462
weighted avg	0.03	0.17	0.05	122462

Figure 2: Experiment 2

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2816
1	0.00	0.00	0.00	9346
2	0.00	0.00	0.00	175
3	0.00	0.00	0.00	11015
4	0.00	0.00	0.00	5632
5	0.00	0.00	0.00	4392
6	0.00	0.00	0.00	11161
7	0.00	0.00	0.00	86
8	0.00	0.00	0.00	2961
9	0.00	0.00	0.00	11010
10	0.00	0.00	0.00	1364
11	0.17	1.00	0.30	12960
12	0.00	0.00	0.00	1302
accuracy			0.17	74220
macro avg	0.01	0.08	0.02	74220
weighted avg	0.03	0.17	0.05	74220

Figure 3: Experiment 3

	precision	recall	f1-score	support
0	0.00	0.00	0.00	7050
1	0.00	0.00	0.00	23156
2	0.00	0.00	0.00	437
3	0.00	0.00	0.00	27763
4	0.00	0.00	0.00	13830
5	0.00	0.00	0.00	10957
6	0.00	0.00	0.00	28137
7	0.00	0.00	0.00	204
8	0.00	0.00	0.00	7347
9	0.00	0.00	0.00	27580
10	0.00	0.00	0.00	3488
11	0.17	1.00	0.30	32260
12	0.00	0.00	0.00	3339
accuracy			0.17	185548
macro avg	0.01	0.08	0.02	185548
weighted avg	0.03	0.17	0.05	185548

Figure 4: Experiment 4

	precision	recall	f1-score	support
0	0.00	0.00	0.00	161
1	0.00	0.00	0.00	18603
2	0.00	0.00	0.00	17174
3	0.22	1.00	0.37	27490
4	0.00	0.00	0.00	14327
5	0.00	0.00	0.00	14477
6	0.00	0.00	0.00	27203
7	0.00	0.00	0.00	3027
accuracy			0.22	122462
macro avg	0.03	0.12	0.05	122462
weighted avg	0.05	0.22	0.08	122462

Figure 5: Experiment 5

	precision	recall	f1-score	support
0	0.00	0.00	0.00	58
1	0.00	0.00	0.00	5637
2	0.00	0.00	0.00	5156
3	0.22	1.00	0.36	8214
4	0.00	0.00	0.00	4352
5	0.00	0.00	0.00	4485
6	0.00	0.00	0.00	8301
7	0.00	0.00	0.00	907
accuracy			0.22	37110
macro avg	0.03	0.12	0.05	37110
weighted avg	0.05	0.22	0.08	37110

Figure 6: Experiment 6

	precision	recall	f1-score	support
0	0.00	0.00	0.00	233
1	0.00	0.00	0.00	28311
2	0.00	0.00	0.00	25974
3	0.22	1.00	0.37	41638
4	0.00	0.00	0.00	21767
5	0.00	0.00	0.00	21799
6	0.00	0.00	0.00	41263
7	0.00	0.00	0.00	4563
accuracy			0.22	185548
macro avg	0.03	0.12	0.05	185548
weighted avg	0.05	0.22	0.08	185548

b. Discussion of Results

My models that were trained to output the district_id seems to be more accurate with accuracy score of .22 compared to the .17 for the ones meant to output the offense_category_id. It might be due to the number of variables that was available to support each possible output being so lopsided, more lopsided for the offense_category_id than for the district_id.

c. Problems Encountered

The project went surprisingly well even though I spent a disproportionate amount of time on the preprocessing. However, I found that I learned a lot about data preprocessing through the process, so I am not too sad about that. However, what bugged me the most is the bug that I encountered that ultimately made me decide to dropna. Another problem that I encountered is my models being so low in accuracy. After further discussion with my instructor, Dr. Sarkar, in the future I should modify my preprocessing process in light of my goal of logistic regression ML. In light of it, I should have focused more on a smaller amount of columns, a smaller amount of possible outputs (say top 10 if they have more than 15) for each column, and then using dummyvariable, made all the columns into binary columns as instructed in class instead of what I did, which is more for visualization.

d. Limitations of Implementation

My models SHOULD NOT be used to predict the target data due to its low accuracy (below .5). However, the model that was discussed in the previous section might possibly be better. However, due to the lopsided amount of data supporting each variable (even in the top 10), I have reason to believe that it might just increase the accuracy to around .5, which is usable but not advised, certainly more usable than the ones I produced.

e. Improvements/Future Work

As previously discussed, in the future I should modify my preprocessing process in light of my goal of logistic regression ML. In light of it, I should have focused more on a smaller amount of columns, a smaller amount of possible outputs (say top 10 if they have more than 15) for each column, and then using dummyvariable, made all the columns into binary columns as instructed in class instead of what I did, which is more for visualization.

Also, I would have liked more time devoted to the creation of an interactive map to be able to show if crime is more frequent in certain areas to make use of the geo locators. Sadly, I had to abandon that idea due to time constraints. I have found the answers, but not the time to implement them.

CII. CONCLUSION

In conclusion, my models are bad and needs more work. My visualizations were good as I was able to glean useful information from them. However, ultimately my goals were not visualization but logistic regression, which means I failed this time. Additionally, in the visualization area, I failed to section off more time for map creation. There were a lot of good and new things that I learned through this project, but I ultimately failed to meet my personal objectives.