Interim Report
Prediction of Crime Patterns in Chicago

Submitted By:
Ravisutha Sakrepatna Srinivasamurthy
Saroj Kumar Dash

November 7, 2017

Contents

1 Introduction **1**

2 Objective **1**

3 Approach **1**
3.1 Building network 1
3.2 Methods to predict next crime pattern 1
3.3 Progress 2
3.4 Timeline 2

Appendices **3**

1 Introduction

In this project we are analyzing the Chicago Crime Data [4]. Following section explains our objective, intended method for achieving the objective and progress until today.

2 Objective

Our objective is to predict crime pattern in Chicago (lets say for next year) by using data from previous years with the help of network analysis techniques.

3 Approach

We are currently contemplating on different methods by which we can achieve our objective. But before applying any method, we are facing some design challenges; namely how to construct a network and what forms a node etc. We have various datasets such as education dataset, income dataset, rider dataset, etc. Our first challenge is to come up with a network using these datasets.

3.1 Building network

We are currently considering crime data for the year 2015. We will use data from this year to test out all intended methods and to analyze them . And then we will extend it to other years. In 2015, there were 263,477 reported crimes. Each crime will be treated as a node. There are total of 401 crime types according to Chicago Police website[5]. And crime can happen either in morning, or afternoon or night or in the early morning. Based on these, we have 401 types crime that can happen in 4 different times of a day. We will consider each of these combination as a node. Hence we will end up with 1604 more nodes. Again, we will divide the crime based on months. This will add 12 * 401 = 4804 more nodes to our network. Each crime is associated with an area. Chicago can be divided based on many attributes such as ward number, precinct, district, etc. Our choice depends on the other datasets as explained below.

Along with crime dataset, we will be using education, rider, income and police beats datasets. But some of these datasets are based on wards and some are based on precincts etc. We have to somehow unify these datasets and then we have to incorporate in our network. We are currently working on this.

3.2 Methods to predict next crime pattern

We are currently contemplating three methods to achieve our objective.

- First method is to build network for different years (let's say 2001 and 2002). Then compare these to networks and analyze the changes. We are hoping to use Laplace inverse matrix[2] to find the dissimilarity in the network and analyze the network based on this matrix. And based on these analysis predict the crime trends for future years.
- Second method is to predict the spread of crime from one area to another area. This is similar to network analysis of how an epidemic disease spreads in a network[3]. This analysis will help in predicting the spread of crime to other areas.

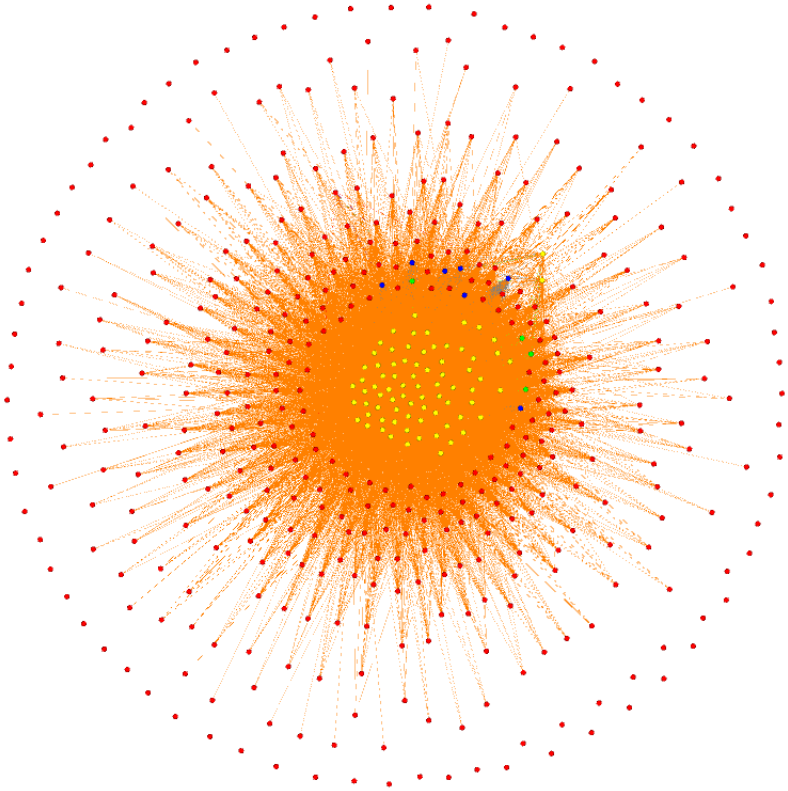

Figure 1: Visualization of crime network. Red nodes are crime types, yellow nodes are the community, green nodes are time of the day and blue nodes represents weekday. This is a preliminary network. Our network will change after we incorporate changes as described in the above sections.

- And finally, in a given year, we want to see what influences the observed crime pattern. For this we intend to use role discovery algorithm that was presented by one of our peer in the class. This may or may not work but we are curious about what this algorithm may come up with.

3.3 Progress

We are currently cleaning the dataset and trying to build the network. We got a network just by using the crime dataset(Figure 1). Nodes for this network are community and crime types.

We are currently trying to unify all other datasets so that it can incorporated in the network as explained in the above subsection.

3.4 Timeline

This is our expected timeline leading to the completion of the project.

- Cleaning the data and building the network should be complete by 11/12/2017
- Apply the three methods and analyze the network by 19/12/2017
- Predict the future crime trends by 11/22/2017
- Get feedback from Dr. Safro and make necessary changes. 11/24/2017
- Prepare the report and submit by 11/27/2017

Appendices

Appendix A :Make network python code:

```
#!/usr/bin/python3

#Author :    Ravisutha Sakrepatna Srinivasamurthy
#            Saroj Kumar Dash
#Project:    Analysis of Chicago Crime Data

import networkx as nx
import datetime
import csv
import numpy as np

class Crime_Network:
    def __init__(self, path):

        #Get the path for the required three data
        self.data_path = path[0];
        self.type_path = path[1]
        self.comm_path = path[2];

        #Build the crime type dictionary
        self.built_type ()

    def built_type (self):
        """Build a dictionary of crime types according to chicago police website."""

        with open(self.type_path, 'rt') as f:
            reader = csv.reader(f)

            self.a = {}
            i = 0
            j = 11
            for row in reader:
                if (i == 0):
                    i += 1
                    continue
                temp = row[0]
                if (temp[0] == '0'):
                    self.a[temp[1:]] = j
                else:
                    self.a[temp] = j
                j += 1

    def convert_date (self, t):
        """ Convert date and time into usable format. """
```

```
        [date, time, m] = t.split (" ")

        [month, date, year] = date.split ("/")

        month = int (month)
        date = int (date)
        year = int (year)

        d = datetime.date (year, month, date)
        weekday = d.weekday()

        [t, temp, temp] = time.split (":")
        t = int (t)

        if (m == "PM"):
            if (t >= 1 and t < 6):                  #1PM - 5PM
                time = 2
            elif (t >= 6 and t <= 11):              #6PM - 11PM
                time = 3
            else:                                   #12PM
                time = 1
        if (m == "AM"):
            if (t < 7):                                 #1AM - 6AM
                time = 4
            elif (t >= 7 and t <= 11):                  #7AM - 11:59AM
                time = 1
            else:                                       #12AM
                time = 4

        return (time, weekday)

    def build_network (self):
        """Build a adjacency matrix using the crime data."""

        self.A = np.zeros((490, 490))
        with open (self.data_path, "rt") as f:
            reader = csv.reader (f)

            i = 0
            for row in reader:
                #Skip first line
                if (i == 0):
                    i += 1
                    continue

                #Get date information
                date = row[2]
                (t, w) = self.convert_date (date)
                t = t - 1
```

```
                w += 4

                #Crime Type
                temp = row[4]

                if (temp[0] == '0'):
                    if (not(temp[1:] in self.a)):
                        c_type = 412
                    else:
                        c_type = self.a[temp[1:]]
                else:
                    if (not(temp in self.a)):
                        c_type = 412
                    else:
                        c_type = self.a[temp]


                #Community Type
                commu = int (row[13]) + 412

                self.A[commu][t] += 1
                self.A[commu][w] += 1
                self.A[commu][c_type] += 1

#Create graph using numpy array
def create_graph (self):
    """ Convert adjacency matrix to networkx graph. """

    G = nx.from_numpy_matrix (self.A)
    return (G)

#Write to file and view in gephi
def write_file (self, G):
    """ Write the network to a file. Helps in visualization. """
    nx.write_graphml(G, 'new.graphml')

#Add atributes
def add_attributes (self, G):
    """ Helper attributes for better visualization. """

    for i in range (np.shape(self.A)[0]):
        if (i < 4):
            G.nodes[i]["color"] = 'green'
        elif (i < 11):
            G.nodes[i]["color"] = 'blue'
        elif (i < 413):
            G.nodes[i]["color"] = 'red'
        else:
            G.nodes[i]["color"] = 'yellow'
```

```
path = []
path.append ("/home/ravi/Network_Science/Project/Data/Crimes_2015.csv")
path.append ("/home/ravi/Network_Science/Project/Data/IUCR.csv")
path.append ("/home/ravi/Network_Science/Project/Data/community.csv")

a = Crime_Network (path)
a.build_network ()
G = a.create_graph()
a.add_attributes(G)
a.write_file (G)
```

References

[1] Ahmed, N. K., Rossi, R. A., Willke, T. L., & Zhou, R. 2016, arXiv:1610.00844

[2] F. Fouss, A. Pirotte, J. m. Renders and M. Saerens, "Random-Walk Computation of Similarities between Nodes of a Graph with Application to Collaborative Recommendation," in IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 3, pp. 355-369, March 2007.

[3] https://www.theatlantic.com/magazine/archive/2013/10/violence-is-contagious/309459/

[4] https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2

[5] https://data.cityofchicago.org/Public-Safety/Chicago-Police-Department-Illinois-Uniform-Crime-R/c7ck-438e