Universiteit Leiden
Opleiding Computer Science

Obtaining Insights into Criminal Contact Networks

Name: Nick van den Bosch

Date: 27/11/2018

1st supervisor: Frank Takes
2nd supervisor: Cor Veenman

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

A lot of investigating in law enforcement is done case by case, without looking at the bigger picture. We use social network analytics to gain insights on suspects and relations between cases in a criminal contact network. First we do so by looking for important criminals using centrality metrics and metadata. Then we attempt to find ways to break up the network by removing these central targets in order to test its resilience. We find the community structure of the network and validate this using statistics associated to individual nodes and look if there is a criminal hierarchy in a network such as this one. One of our findings is that the network consists of some central actors that act as bridges in the network, that distort the flow of communication when removed. This suggests that using social network analysis, law enforcement can potentially cause disruption within communication networks of criminals.

Acknowledgements

This master thesis was done at LIACS, in collaboration with the Dutch police at Amsterdam-Amstelland, under the supervision of Frank Takes and Cor Veenman (University) and Dominique Roest (Police). I would like to thank them for their help, advice and supervision during this project.

Contents

1 Introduction

This section contains background information about the topic and domain of the thesis, leading to the questions that will be investigated and concluding with an overview of the rest of the thesis.

1.1 Background

A lot of processes in the line of work of law enforcement require manual actions to complete. Examples of this are filing police reports, extracting data from devices and determining new leads to further investigate cases. Another process that still requires a fair amount of manual action is data analysis. Data analysis is the key to successfully use information, by performing actions on raw data you transform this raw data into intelligence. At the dutch police in Amsterdam-Amstelland, some of this process is already automated by software that is developed for investigations, but there are still steps to be made in performing high quality automated data analysis.
Authorities such as the police generally have a number of investigations running simultaneously, each of which containing a variety of data. Some of these investigations are completely unrelated to each other, but some might have an overlap, such as common suspects, or multiple investigations that relate to each other by having the same type of crime. These shared data attributes between investigations can be analyzed to bring to light new information that can be helpful when trying to solve these cases.

One of these attributes that one could examine are social circles, i.e. people's contacts. In this day and age everyone uses mobile devices to communicate with each other. We use mobile phones to call, text and e-mail our social contacts. A lot of this social interaction is stored in our devices, such as logs of calls, texts, e-mails and contact information. Using this data we can see who is connected with who through their contacts, and we can create a network of people and the ones they interact with on their devices, called a social network.

Network analysis has been developing for several decades and is very popular in the social sciences [1], amongst others. It is used to handle the large amounts of big data that are available nowadays, as a fair amount of network analysis methods scale well to bigger datasets. An example of social network analysis is where a network is used to predict and evaluate the outbreak of diseases. In [3], Christakis et al. show an analysis of the transmission of an influenza pandemic, where they characterize factors that impact the spreading of this disease using a social network.

When creating such a social network of a group of people that have relations to each other, we would get an interconnected network from which we can extract information using social network analysis methods. An example of this is that one can distinguish groups such as mutual friends, family or coworkers by comparing connectivity between nodes in this network. This is referred to as a community, when a group of nodes are more connected internally than they are with nodes outside of this group. Certain people might get classified into groups that are to be expected, or people can be found in groups that one would not expect. Besides classifying specific people in these so called communities, the community structure of a network as a whole can give insight on the data.

Another interesting measure is that one could determine people that are central to the network, which can be expressed in a variety of different ways. Examples of this are whether a person has a lot of connections in general, or whether a person is connected to people with a lot of connections. Using these so called network centrality measures [4] we will study the social circles of suspects and criminals, to provide additional insight to relations within or between investigations. Perhaps there is a central figure that connects a big part of the crime happening in Amsterdam, or a person high up in the criminal circuit trying to stay hidden, communicating through accomplices. By using measures such as centrality and community detection one could get insights like these in these networks.

Data of this nature can often be incomplete, as criminals attempt to work in secrecy as much as they can. A prime example of this is their methods of contact. While most people have one phone with all of their regular contacts in it, criminals tend to have multiple phones, sometimes with only one or not a single contact in it. They do this to hide their operation and means of contact as much as possible. This means multiple devices can belong to the same entity, which can lead to an incorrect interpretation of the criminal network resulting from such data.

1.2 Research Question

This leads us to the proposed research question:
What insights can we get on suspects and relations between cases by analyzing data of criminal contact using social network analysis?

We can divide this research question into several sub-questions:

1. Can we identify multiple devices to be the same entity?
2. Can we identify the most 'important' criminals using network analysis?
3. Can find 'key players' that break up the criminal network when they are arrested?
4. Can we detect groups of criminals that perform the same type of crime?
5. Is there a hierarchical structure in this criminal network?

1.3 Thesis Overview

After this introductory section, the thesis is divided in five other sections. Section 2 describes work related to this research and gives an insight in the domain of law enforcement data and criminal networks. Section 3 discusses the statistics of the data that was used. It also goes into the preprocessing and refining that was done to ensure that the data was transformed in a way that was suitable for social network analysis and that the quality of the data was up to scientific standards. Section 4 describes the methods that were used to analyse the data. Section 5 presents the application of these methods to the data and shows the results of the experiments, answering the research questions. However, the first research question will be discussed in Section 4, as a lot work was done in preprocessing and this involved this problem of entity resolution that includes the first research question. Section 6 contains our conclusions and suggestions for future work.

2 Related work

In this section we will highlight some of the previous work in the field of social network analysis, specifically on criminal networks, and how this relates to our research.

2.1 Criminal Network Characteristics

There are many challenges to overcome when investigating and analyzing criminal network data, compared to other data. Criminals often have different behaviour because of what they do, they try to work in secrecy as much as possible to avoid authorities, which can results in data intelligence problems. Morselli [2] states that criminal networks differ from regular network in certain ways, because they "face a constant trade-off between security from law enforcement and efficiency of their operation. Criminal networks are not simply social networks operating in criminal contexts. The covert settings that surround them call for specific interactions and relational features within and beyond the network [5]".
Adderley et al. [6] and Sparrow [7] state that the data available by law enforcement agencies such as the police have several aspects that differ from standard social network data:

- Incompleteness - Criminals work in secrecy and do not want to be identified by authorities, therefore the data is inevitably incomplete. This can result in things such as missing links between people or missing suspects in a network.

- Incorrectness - The data held by authorities could contain incorrect information, either by criminal intent by tampering with data or due to human error while processing the evidence and entering the data in the system.

- Network Dynamics - Criminal Networks are dynamic, as many networks are. Meaning they are likely to evolve over time.

- Fuzzy Boundaries - The boundaries of a criminal network are ambiguous. Crimes within organizations are often interrelated with other organizations. This means that individuals do not necessarily have to belong to one community of criminals but can have ties to multiple organizations.

When experimenting we will take these different aspects into account and interpret the results from our experiments accordingly.

2.2 Criminal Communities: Crime Gangs

In their research, Oatley and Crick describe the community structure in a criminal network by identifying crime gangs in the UK [8]. They identify that if the network exhibits global clustering this indicates the presence of groups of criminals in the same gang. Within these gangs, local clustering tells something about the structure within the gang. They describe an internal gang structure of four different roles within a gang:

- Leader - Responsible for recruiting, sanctions 'missions' for enforcers.
- Provider - An individual either internal or external that supplies the gang with firearms and/or ammunition.
- Enforcer - An individual that is an active gunman for the gang.
- Runner/Dealer - A member who distribute or supply drugs, usually on the leaders behalf.

One of the goals of this research is to see if the community structure of the crime in Amsterdam is similar to internal gang structure that was found by Oatley and Crick or if we find different results.

2.3 Identifying Key Players in a Social Network

In [9], Borgatti proposes two methods to define sets of key players in a social network, KPP-POS and KPP-NEG.

The KPP-NEG method is used to determine central entities in the network, so called key players. These entities are not just central in the network, they have the property that if they are removed from the network this causes maximal fragmentation, breaking up the network.

His method involves a greedy optimization algorithm that uses fitness features based on key player metrics such as the Herfindahl index [10].

It starts with selecting a set of random nodes and computing the fitness value of this set. After which it iteratively perform two steps until the stopping criterium is matched:

- Loop over each node in the random set and each node not in the random set, and compute the fitness difference of the random set if these nodes were swapped.
- If the fitness difference is smaller than the stopping criterium, stop. Else swap the nodes with biggest fitness difference and update the fitness of the random set.

Fragmenting the network by removing key nodes is something that could prove useful in law enforcement. It can be beneficial to determine key criminals that interconnect the different components of the network, because arresting these individuals, this could potentially result in groups of criminals losing contact with one another.

3 Data

The following section describes the process of extracting the data, refining it, ensuring the quality of the data and the methods used for analyzing the social network created with this dataset.

3.1 Data Extraction

For this research we used confidential data extracted from devices, primarily mobile phones, that have been collected for investigations by the dutch police in Amsterdam. The police extracts information from phones that are found or taken from suspects to use the information to aid investigations. These are primarily from potential suspects, but could also be devices found in and around houses that have been raided or investigated.

For each available device we have extracted their list of contacts. For each of these contacts we have the name of the contact, the phone number of the contact, the name of the item containing the contact and the name of the investigation corresponding to that item. By linking our dataset to a different system of the police we also obtained another a domain related measure we call crime weight or crime severity. The crime weight of an individual is determined by the highest crime weight of any of the cases they are involved in. This means that if they are involved in a lot of cases, there is a higher chance that they are involved in a more severe case, resulting in a higher crime weight. This crime weight consists of a number between 1 and 6, where 1 are relatively small crimes and 6 are the most severe crimes one could imagine. There are also some nodes classified with crime weight 7, which is the case when there was no information about the severity of the crime.

Representing the raw data in a network resulted in a network with 343,200 nodes and 469,677 edges.

3.2 Refining the Data

To ensure that our research with this dataset leads to valid results, we refined the data by removing excess information and normalizing certain attributes.

3.2.1 Servicenumbers

A lot of the items in this dataset have lists of contacts that contain servicenumbers. These numbers are irrelevant for doing social network analysis and can create confusion when analyzing the network. A very common servicenumber such as the emergency service can be in a lot of people's contacts, but it does not say anything about whether these people are connected through this number or not. These service numbers will also likely have a lot of connections in the network, as they are commonly found in people's phones, but this does not mean they are a significant entity in this criminal contact network.

For these reasons we excluded all servicenumbers using a list of known servicenumbers. We have also manually extended this list by analyzing entities with a very high degree in the network, being entities with a lot of connections, and checking odd numbers for whether they are a servicenumber or not.

Removing these servicenumbers also causes some phones to have a completely empty list of contacts. These are likely phones that have just been bought and only have servicenumbers of their mobile operator in their phone. As a result they are single nodes in the network that do no interact with anyone and do not add any value to the data. These nodes were also excluded from this research. After removing these servicenumbers from the data this resulted in a network with 342,438 nodes and 461,476 edges.

3.2.2 Normalization

A lot of contacts with phone numbers in mobile devices are stored by saving the number after being called or receiving a text from someone. One then click on this number and save it as a contact, however, one can also manually save contacts and input the phone number themselves. In a large dataset like the one used in this research there are a lot of cases of people entering a phone number for a contact incorrectly. There are also multiple ways to correctly enter the same phone number, for example, in the Netherlands a correct cellphone number would be 0612345678. Another correct way of saving the same number in the Netherlands would be +31612345678 or 0031612345678, as the country code for the Netherlands is 31. If you would call either of these three numbers in the Netherlands you would connect to the same phone number.

Seeing as we want to differentiate contacts based on their phone number we normalized these different forms of numbers to one format and cleaned all phone numbers of incorrect symbols, typing errors and incorrectly formatted phone numbers. Table 1 shows examples of the actions of normalization and cleaning of the phone numbers to get them to one format. Each of these numbers is normalized to the standard phone number format we use for this research, which is 31612345678 for this example.

3.3 Data Quality

Each item in our data has a name consisting of an item number and what type of phone it is, for example 1234567 IPhone 6. The idea behind this was that this number preceding the type of phone was to be unique so each device had its own unique name. In practice this was not the case and there were item names corresponding to multiple devices, as well as that the information from some devices was extracted and stored multiple times using different software or different versions of the same software.

Table 1: Phone number normalization and cleanup examples

Action	Example
Remove leading 31+	31+31612345678
Remove ++	31++612345678
Remove spaces	31 6 12 34 56 78
Remove dashes	31-612345678
Remove leading +	+31612345678
Remove leading ’	’31612345678
Reformat 06-numbers	0612345678
Reformat 0031-numbers	0031612345678
Replace 031	031612345678
Replace/reformat (0)	(0)612345678
Replace leading 31(0)6	31(0)612345678
Replace leading 0	031612345678
Reformat/Clean number	316+31612345678+3123456

For doing social network analysis on this data we want to distinguish people, also referred to as entities, with a unique identifier. This is so each node in the social network corresponds to exactly one entity and there are no groups of multiple nodes representing the same entity or multiple entities represented by the same node. Ideally, all the nodes in the network should be unique entities, however in practice, seeing as information from some devices gets extracted multiple times and stored separately we could have entities that are represented by multiple nodes. We want to minimize this to keep the quality of the network high and the information extracted from the network as accurate as possible. This problem of linking records of the same entity by means of a unique identifier is called entity resolution [11] and we took several measures to solve this problem to improve the quality of the data, as discussed in Section 4.2 and 4.3.

4 Methodology

In this Section we will go into the methods that were used to answer the research questions described in Section 1.3.
Section 4.1 contains social network definitions that will be used in the sections thereafter. Section 4.2 is about methods to deal with the problem of entity resolution. Section 4.3 goes into the process of extracting the own numbers how they contribute to entity resolution. Section 4.4 is about identifying/ranking important individuals. Section 4.5 is contains methods to analyze the community structure of a network.

4.1 Social Network Definitions

Social network analysis is the process of investigating social structures through the use of networks and graph theory [12]. This can be done by representing the data in a network structure, where entities in this network are characterized as nodes that are connected by their relations/interactions characterized as edges or links. When representing data in this way one can extract additional information from this data using network statistics and algorithms.

A social network can be represented as a graph $G = (V, E)$ with a set of socially relevant objects V, also referred to as nodes, entities or vertices, connected by a set of relations E, also referred to as links or edges. Figure 1 shows an example of such a network with 6 nodes and 6 edges.
This network is undirected and unweighted. That the network is undirected means that every connection has no orientation. An edge (v, w) is identical to the edge (w, v) for $v, w \in V$. An example of this is a friendship between

two people. An example of a relation in a directed network would be a transaction of one person to another in a financial network. That the network is unweighted means that every link has the same weight, so each relation is of the same strength.
This example network connects all nodes in one component, which is not always the case. If for example the edges v, w and v, x were removed the network would consist of two components, where there would be one component of two nodes and one component of four nodes. The component with the majority of the network's nodes is called the giant component.
The network that is analyzed for this research is a network based on the contact between phones of criminals that have been extracted by the police in Amsterdam.

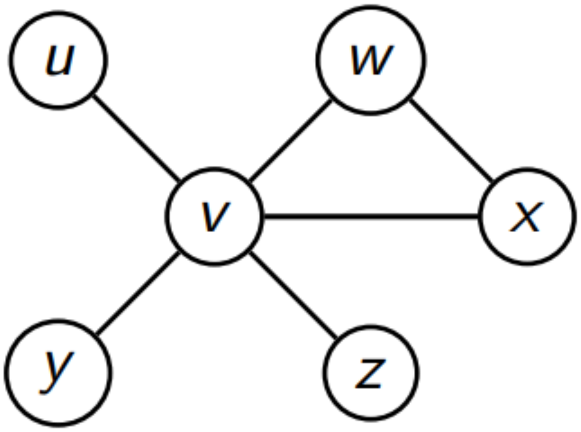

Figure 1: Network with 6 nodes and 6 edges.
Source: `http://liacs.leidenuniv.nl/~takesfw/SNACS`

4.2 Entity Resolution

To deal with the problem of entity resolution discussed in Section 3.3, we compared features of the devices, where the main feature is the list of their contacts. We did a pairwise comparison of all phones in the dataset to compute the percentage of overlap in their contacts. If a pair of phones has at least one contact in common, it is then taken in consideration, storing the size of each of the contact lists, the number of contacts in common and the percentage of overlap. To compute this percentage of overlap experimented with several methods of determining set-similarity, such as the Jaccard Index [13], the Srensen-Dice coefficient or simply taking the average of both the overlap of phone 1 in phone 2 and the overlap of phone 2 in phone 1. We came to the conclusion that in this domain, the latter option was a very inaccurate representation of the overlap between the 2 lists of contacts, and that the Srensen-Dice coefficient resulted in a much higher percentage of overlap than the Jaccard Index when there were only a few contacts in common. The Jaccard Index seemed to give an good representation of how much phones overlapped but from a domain perspective this was not accurate enough. As per requested, to determine whether 2 phones belonged to the same entity we computed both the overlap of phone 1 in phone 2 and phone 2 in phone 1. Combining this overlap with meta information such as the name of the item, that commonly consists the type of phone, or the name of the case where it was used in, the police can accurately determine whether devices belong to the same entity. Figure 2 shows a visual of the frequency of occurrence of percentual overlap in pairs of phones that have one or more contact in common. We see that most phones have a relatively low percentage of overlap in contacts, and there are interesting outliers at 50% and 100%. The peak at 100% is due to devices being extracted multiple times, or several devices that belong to the same person.

4.3 Extracting Own Numbers

Other than the methods described in Section 4.2, we dealt with the problem of entity resolution by determining the phone numbers corresponding to the devices in the data, further referred to as ‘own numbers’. This data originally was extracted from phones using the extraction software that the police uses,

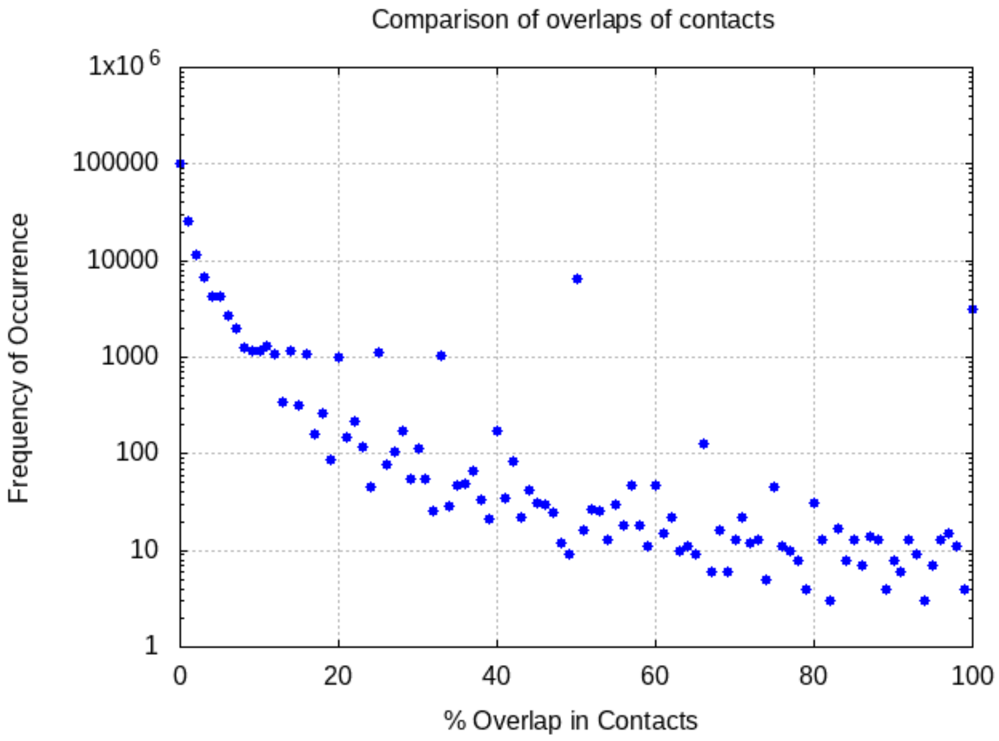

Figure 2: Contact overlap in pairs of phones

but it was stored in files with a lot of other information. These own numbers were not extracted from these As stated in Section 3.3 we want unique identifiers for each entity in our social network and the item names were not exactly unique. As the own numbers are neccesary to identify telephone nodes, we created a parser that finds all the extracted files of a device, finds corresponding police reports that can possibly contain this information and then parses these files to retrieve the phone's own number. Of all devices in their database, of 31% of these devices the own number was extracted.

Another incentive to retrieve the own numbers of the devices is if we do not have this information when making a social network of phones and their contacts using this data, we would only have links between phones and contacts, and not a single connection between two phones. This also results in multiple nodes representing the same entity. A phone in the dataset could have the own number 31612345678 and this number could be in the list of contacts of a different phone. If we do not know the own number of the phone, we would have a node for the phone with some identifier, and a node for the contact 31612345678, even though they are in fact the same entity.

Not having these own numbers would result in a very misleading social network, as there would be certain nodes with only outgoing links, these being the phones. And there would be certain nodes with only incoming links, being the contacts. Figure 3 is a visual of how one phone with its contacts would look like.

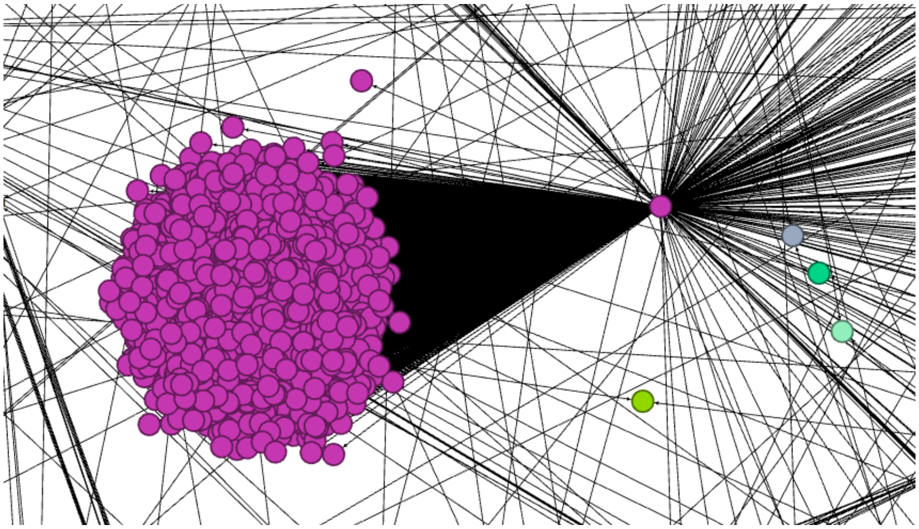

Figure 3: Phone with its contacts when no own number is known

In a network such as this one, if the own numbers are unknown there is a high chance that a person is represented by more than one node. This is because the phones that get extracted are nodes in the network defined by a unique identifier instead of their own number. The contacts in these phones get defined by their phone number and there is no way to know whether a 'phone node' with its identifier is the same entity as a 'contact node' with its phone number. This means there will be a lot of redundant nodes and edges and the network will not be a realistic representation of the situation. Figure 4 shows an part of the network without solving this entity resolution problem by defining these own numbers. It shows that the network almost exclusively consists of these 'phone' nodes with their big cluster of contacts, with very limited edges to other nodes and zero connections between phones.

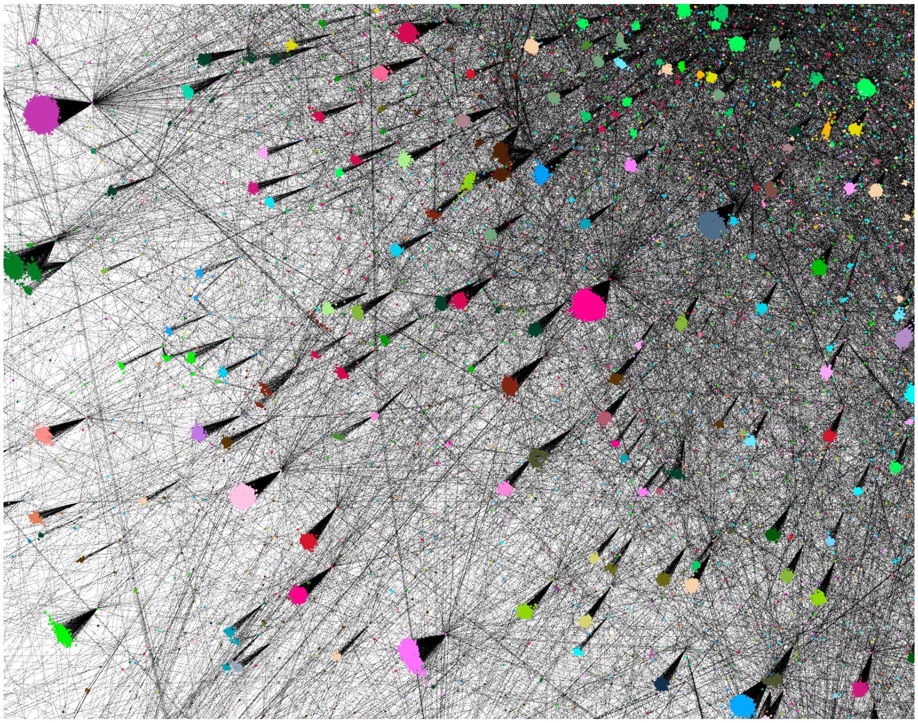

Figure 4: Structure of the network without own numbers

By determining these own numbers we were able to create a much more realistic representation of the data that was presented to us. Although, when looking at the network visualization we still find nodes with a high amount of overlap that potentially are the same entity, which can be seen in figures 5.

In these figures, the pattern of overlap in contacts seems too interesting to be random. We see that there are three nodes in the bottom of the figures that are connected with a large number of contacts above, and the overlap of these contacts is interesting. The node in the middle is connected with all of these contacts, and both the node on the left and the node on the right are connected with about half, where they both have a group of nodes that does not overlap, and a group in the middle that does overlap. We expected this to be devices that belong to the same entity, or devices from the same investigation. However, for most of the occurrences of this pattern this was not the case. This means that after one performs network analysis on a network such as this, there are still patterns that stand out, which can be interesting to be manually investigated by the police.

4.4 Entity Ranking

When extracting information in the domain of crime data, an important measure in network analysis is how the network is connected. Which people are 'key' in a network, and how to determine these important individuals. One way of ranking these entities in a network is looking at how central they are in the network compared to others. By combining these rankings and potentially adding domain specific information one can determine the importance of people in a network.

There are many ways to determine the centrality of a node in a network compared to other nodes. For this research we used a handful of centrality measures that will be described in this section. These will be used to answer two of the research questions that were mentioned in the introduction in Section 1.2.

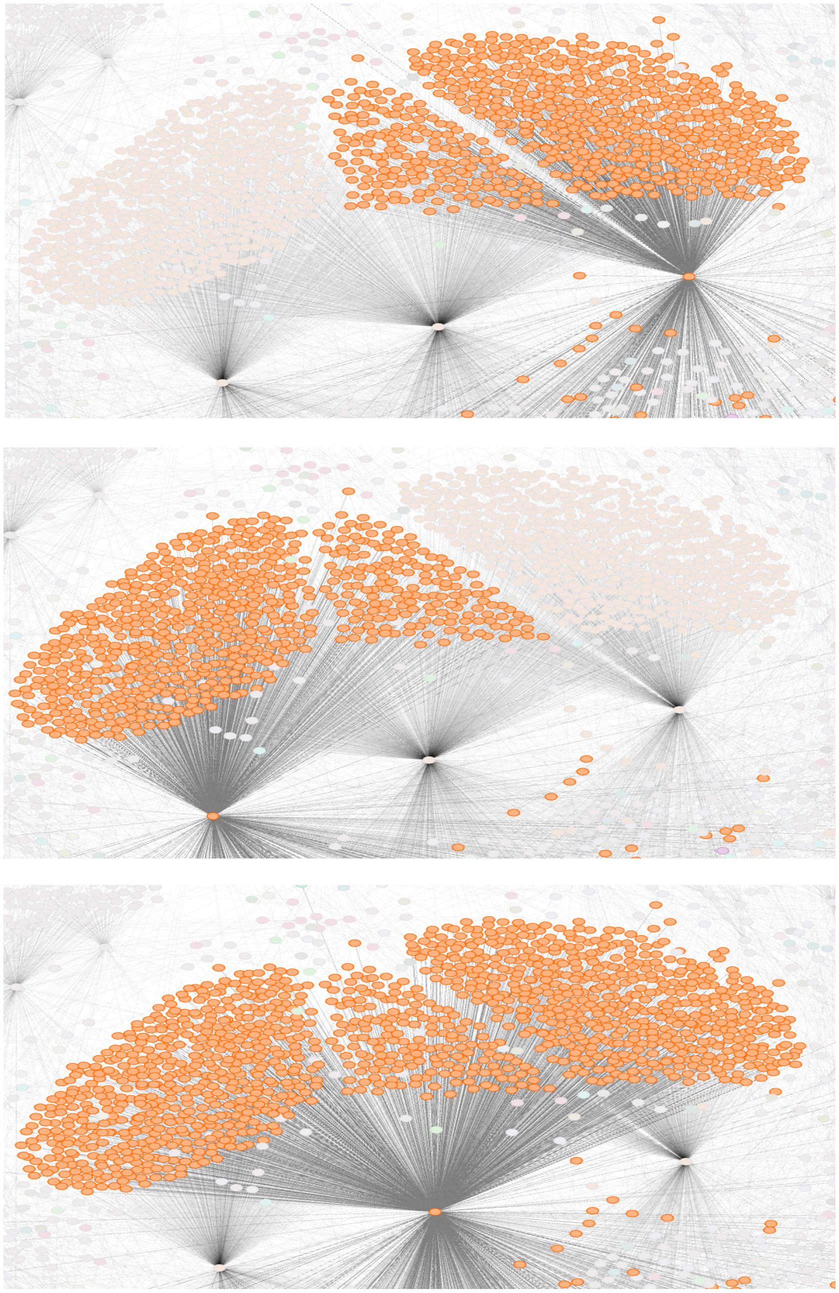

Figure 5: Interesting overlap pattern

4.4.1 Degree Centrality

The degree centrality of a node is conceptually the simplest centrality measure. It is based on the degree, the number of links the node has. If a node is connected to many other nodes in a network it has a more central position in terms of local power, as in this case it has contact with numerous people in the network. Criminals that only know a few other criminals are generally less interesting than criminals with a lot of connections in the criminal circuit. The degree centrality $C_d(v)$ of a node v is equal to the degree of the node, defined as $C_d(v) = deg(v)$ where $deg(v)$ is its degree. Nodes with a higher degree have a higher degree centrality.

4.4.2 Closeness Centrality

The closeness centrality of a node is the average length of the shortest path between this node and all other nodes in a network. The closer a node is to all other nodes, the smaller the size of the shortest paths will be and the more central this node will be in the network.

It is defined as $C_c(v) = \frac{1}{(\frac{1}{n-1}\sum_{w \in V} d(v,w))}$

Where $C_c(v)$ is the closeness centrality of node v and $\sum_{w \in V} d(v, w)$ is the average shortest path length from node v to any other node w in the network. A higher closeness centrality means the node is more central in a network.

4.4.3 Betweenness Centrality

The betweenness centrality of a node is determined by the number of times this node is included in the shortest path between two other nodes. This is based on people being an important factor in the connection to others. If a node connects a large number of other nodes this node is central to the network.

Betweenness centrality of node u is defined as $C_b(u) = \sum_{v,w \in V} \frac{\sigma_u(v,w)}{\sigma(v,w)}$

Where $\sigma(v, w)$ is the number of shortest paths from node v to node w and $\sigma_u(v, w)$ is the number of these shortest paths that run through node u. Nodes with higher betweenness are more central in a network.

4.4.4 Eccentricity Centrality

The eccentricity centrality of a node is based on the concept of node eccentricity. This is the length of the longest shortest path from this node, or the distance to the node furthest away from the considered one.
The eccentricity $e(v)$ of node v is defined as $e(v) = \max_{w \in V} d(v, w)$
The eccentricity centrality of a node is then obtained by simply taking the inverse of the eccentricity: $C_e(v) = \frac{1}{e(v)}$
Nodes with a higher eccentricity centrality are more central in a network.

4.4.5 PageRank Centrality

PageRank centrality is based on the principle of the PageRank [14] algorithm that google uses to rank websites in their search engine results. It is a variant of eigenvector centrality [15] that uses the structure of incoming links of a node.

It is defined by repeatedly applying $P_c(i) = \alpha \sum_{j=1}^{n} a_{ji} \frac{P_c(j)}{out(j)} + \beta$ until convergence, then setting $C_p(i) = P_c(i)$

Where α is a constant, β is equal to $\frac{1-\alpha}{n}$ where n is the number of nodes and $a_{ji} = 1$ is if there is an edge from j to i.

4.4.6 Detecting Key Players

As described in Section 2.3, Borgatti proposes two methods to define sets of key players in a social network. Entities are considered key players when they are central in a network in such a way, that when they are removed from the network this causes a high degree of fragmentation. This means that the network gets more disconnected after removing these entities than it was before. Borgatti proposes a way to randomly select nodes, compute the fitness value of these nodes, which is determined by the fragmentation that is caused by removing them from the network, and in this way find a set of nodes with maximal fragmentation.

While Borgatti's method is random, there is a centrality measure that is likely closely related to network fragmentation: betweenness centrality. As described in Section 4.4.3, betweenness centrality is the measure that indicates how many times a node is used as a bridge in the shortest path from one node to another. We propose that by removing nodes with high betweenness values, one removes these bridges and this may cause fragmentation of the network.

4.5 Community Detection

When looking at the community structure of a network there are groups of nodes that are more strongly connected with each other than with the rest of the network. Finding these groups, so called communities, can be a computationally difficult task as the size of these groups is typically not known beforehand.

Finding communities can give additional information about a network. In the field of crime data one could for example find gangs, or criminals performing the same type of crime in certain locations. In most cases these communities are detected by community detection algorithms and there are a variety of algorithms that can perform these computations. In this work we have used the Louvain [16] method.

The Louvain method for community detection is a method based on an iterative process of modularity optimization. Modularity, defined as a value between -1 and 1 measures the density of edges inside communities compared

to edges between communities.
Maximizing this value theoretically results in the best possible grouping of the nodes in a network. Instead of trying all possible combinations of nodes in groups, the Louvain method first finds small communities by maximizing modularity locally on all nodes. These small communities are then represented as one node and this process is then repeated until no modularity increase can occur.
The general definition of modularity (for a weighted graph) is defined below. However, for our experiments we worked with an unweighted network, meaning all the weights are equal to 1.

$Q = \frac{1}{2m} \sum_{ij} \, [a_{ij} - \frac{k_i k_j}{2m}] \; \delta(c_i, c_j)$

Where

- a_{ij} represents the edge weight between i and j
- k_i and k_j are the sums of the weights of the edges attached to nodes i and j respectively
- m is the sum of all edge weights in the graph
- c_i and c_j are the communities of nodes i and j
- δ is a simple delta function that is 1 if $i = j$ and 0 if $i \neq j$

An iteration of the Louvain method has two iterative phases. First, each node is assigned to its own community. Then for each node i the algorithm calculates the change in modularity when i is removed from its own community and moved to the community of each neighbour j of node i. This change in modularity, ΔQ, can be calculated using:

$\Delta Q = [\frac{\sum_{in} + 2k_{i,in}}{2m} - (\frac{\sum_{tot} + k_i}{2m})^2] - [\frac{\sum_{in}}{2m} - (\frac{\sum_{tot}}{2m})^2 - (\frac{k_i}{2m})^2]$

Where

- $\sum_{in}$ is the sum of all the weights of the links inside the community that node i is moving to
- $\sum_{tot}$ is the sum of all the weights of the links to nodes in the community
- k_i is the weighted degree of i
- $k_{i,in}$ is the sum of the weights of the links between i and other nodes in the community

Once ΔQ is computed for all communities that node i is connected to, it is moved into the community for which the highest modularity increase is attained. In case there is no improvement of modularity, i stays in its original community. Once there is no improvement in modularity for any node, the first phase of the algorithm is finished.

The second part of the algorithm is where all of the nodes in one community get grouped and now represent one node. A new network is built from these grouped nodes where links between nodes in the same community are represented as self loops on the new community node. Edges from multiple nodes in the same community to a node in a different community are represented by weighted edges. Once this process of rebuilding the simplified network is done, the first phase can be reapplied to this network.

These two steps are repeated until there is no further increase of modularity. Now communities in the network have been identified and these can be analyzed to find nodes with similar connections.

5 Experiments

This section contains the experiments that we ran to answer the research questions defined in Section 1.2.
Section 5.1 describes the experimental setup used for this research. Section 5.2 shows characteristics of the network. The remaining sections contain the application of the methods described in Section 4 to answer the research questions.

5.1 Experimental Setup

Most of the network analysis methods that were used were programmed in python using the Networkx [17] package. We also used the Teexgraph [18] library for large scale network analysis. Visualizations were done using Gephi [19]. The experiments were conducted in a VMware virtual machine running Ubuntu that was connected to a local PostgreSQL database server containing the data.

5.2 Network Statistics

Table 2 shows characteristics of the criminal contact network.

This table shows that almost all nodes are connected with one another in the giant component. The experiments that we ran were all ran on the giant component of the network. Roughly 98% of every entity in the dataset is connected, which is unexpected considering main connecting points such as servicenumbers have been removed beforehand. One would think that, because criminals tend to work in secrecy as much as possible there would be less connectivity in a network such as this one. A giant component of this magnitude shows that this criminal network is more interconnected than expected.
The remaining 2%, 6793 nodes, consist of 799 components. While the giant

Table 2: Network Statistics

Measure	Value
Number of Nodes	340,262
Number of Edges	434,864
Number of Components	800
Nodes in Giant Component	333,469
Edges in Giant Component	428,603
Number of Triangles	76,371
Average Clustering Coefficient	0.04010
Average Degree	2.556
Average Shortest Path Length	6.263

component will likely consist of organized crime, the rest of the components will likely consist of smaller groups of criminals or people that work alone or in sufficient secrecy.

The table also shows that the triangle count of the network is relatively low. We speak of a triangle when in a group of three nodes there are edges from each node to the other. When comparing this statistic to a network such as Facebook or Twitter, we see that for this network the triangle count is much lower than the number of edges and nodes. A part of the reason for this is that not all of the own numbers were known, and triangles can only form with phones for which the own number is known. This is because without own numbers there would only be links from ‘phone’ nodes to ‘contact’ nodes and there is no way to create a triangle in this fashion. Another reason for this is because in general, social interactions show homophily (people tend to connect to people that are similar to themselves) and transitivity [20], but this shows that for this crime network these relations apply in a smaller manner. This form of non-transitive connectivity is also shown in the relatively low average clustering coefficient of 0.04010. As the cluster-

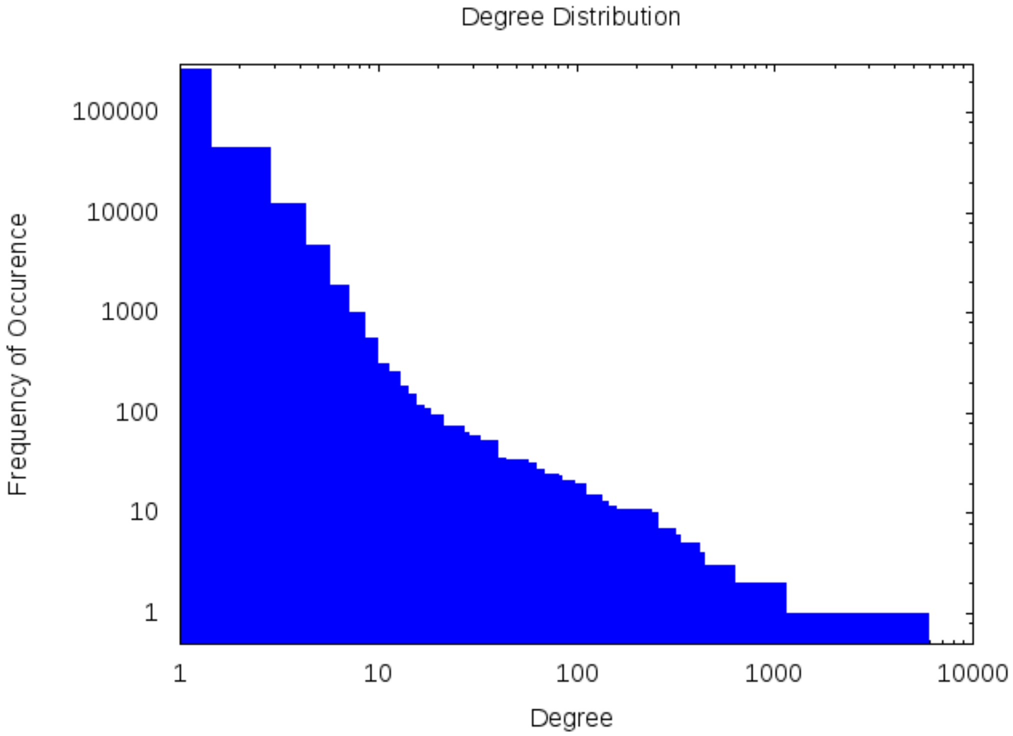

Figure 6: Degree Distribution

ing coefficient is dependant on the number of triangles, this is to be expected.

Figure 6 shows the distribution of degree of the network.
We found that when looking at the degree distribution this criminal network follows a power-law, where there are a large number of people in the network with a low degree, and the higher the degree of the nodes get, the smaller the number of nodes with this degree gets. This means the degree is not a normal distribution but is more skewed, meaning there are very few people with an extremely high degree and a lot of people with an extremely low degree, as opposed to a normal distribution.
Another interesting measure that can be read from Table 2 is the average

shortest path length. For a network of this magnitude it is interesting to see that the average shortest path length is only 6.263. This means that from any node to in the network to another, on average, it takes 6.263 edges.
Considering that the average clustering coefficient is only 0.04010, there have to be nodes that are connected to a large number of people for the average shortest path length to be 6.263. These so called 'hub' nodes are an important factor to interconnecting the network. We can see the presence of these nodes in Figure 4, where some of these hub nodes have over a thousand connections. Having a dense core with hub nodes, a small average shortest path length and a high clustering coefficient are properties that can indicate that the network is a small world network. This network has two out of three of these properties, but the clustering coefficient is lower than the expected value for one of a small world network. This is most likely due to that the own numbers of phones are not all known in the network. Because not all of these own numbers are known, some connections between phones are missing, which results in a lower triangle count, which in turn results in a lower clustering coefficient. If all the data was available there is a good chance that this network would be a small world network.

5.3 Centralities

Another way to obtain insights is to look at network statistics and domain related attributes and see how they relate to each other. One way to do this is a scatter plot. This is a visual of two measures plotted against each other, showing how these measures relate to one another. Making a plot for each pair of these measures results in a scatter plot matrix, which can be seen in Figure 7.
For each node, from left to right the measures that can be seen in the scatter plot matrix are:

- Degree centrality C_d

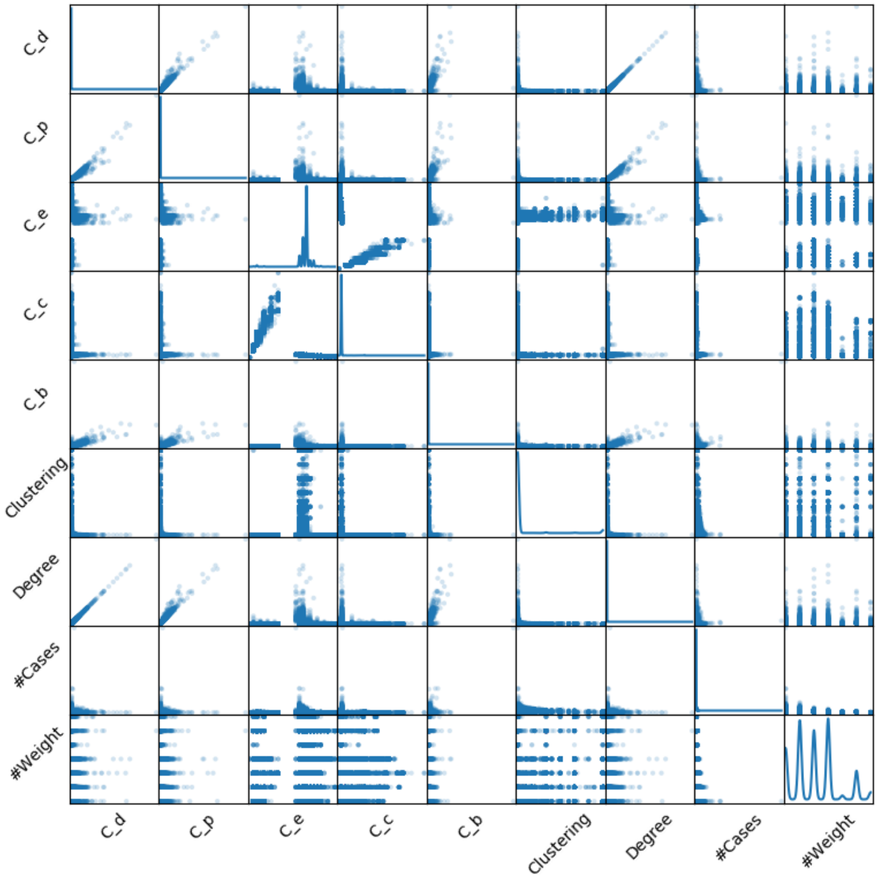

Figure 7: Matrix of relation between node-specific metrics

- PageRank centrality C_p
- Eccentricity centrality C_e
- Closeness centrality C_c
- Betweenness centrality C_b

- Average clustering coefficient
- The number of connections (degree)
- The number of cases an entity is involved in
- The severity of the crimes as defined in Section 3.1

Many of the scatter plots in this matrix are network measures related to other network measures, but the most interesting scatter plots are those of how network measures relate to domain specific measures.
While we only have two domain specific measures in this matrix, as one may expect, these correlate. The number of cases and the severity of the crimes of an individual relate to each other, such that if an individual is involved in a lot of crimes, the severity of their most severe crime is higher than someone that is involved in fewer crimes.

Besides this correlation of domain specific measures, this scatter plot matrix does not show other obvious correlations between domain specific measures and network measures. We do measure a correlation between betweenness and crime weight and some correlations between betweenness and other network measures such as degree and PageRank centrality. This means that some nodes with a high betweenness are also significant to the criminal network in a different way, which can make them important entities for the police to watch or investigate.

5.4 Ranking ‘Important’ Entities

There are several ways to approach finding ‘Important’ nodes in a network. The first is applying network analysis methods and determining central nodes that are crucial for connecting the network. Another way is to look at domain specific measures. For this research we did not have a lot of domain specific information, but we did have some case related information such as for each node in which cases they were involved.

By linking the database where our dataset is from to another system that the police had where they have information about the severity of the crimes, we found the 'crime weight' for each node. This crime weight consists of a number between 1 and 6, where 1 are relatively small crimes and 6 are the most severe crimes one could imagine. There are also some nodes classified with crime weight 7, which is the case when there was no information about the severity of the crime. Figure 8 shows a diagram of the distribution of crime weight in this network.

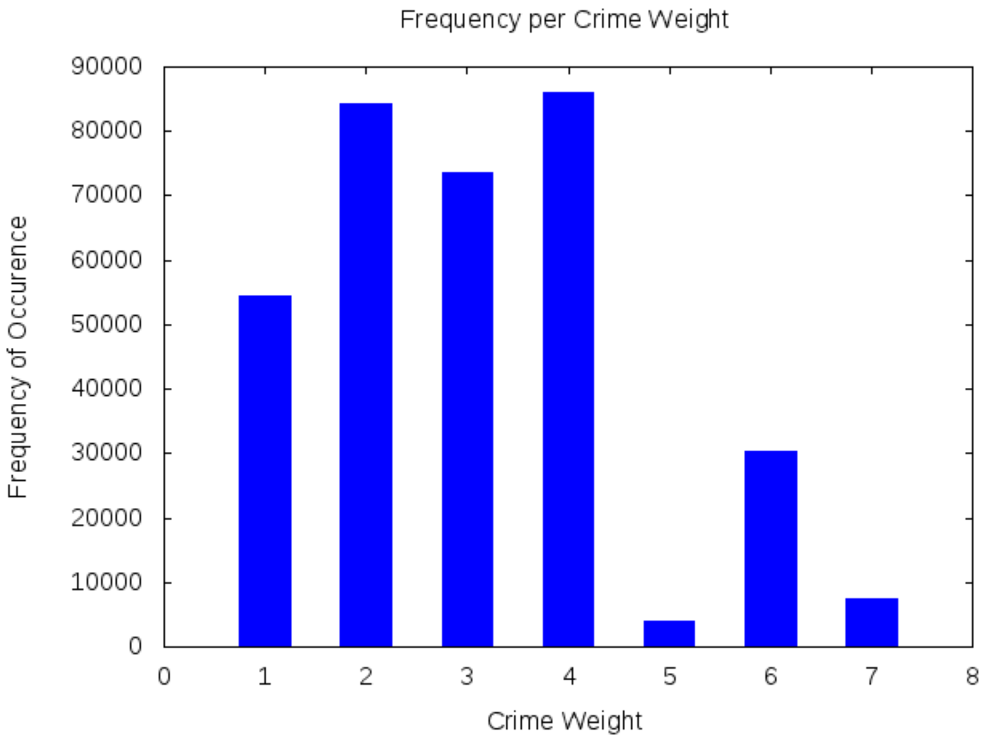

Figure 8: Crime weight distribution

In this figure, each node is ranked by their highest crime weight. This is because generally, more severe crimes determine a criminal. For example if someone is involved in shoplifting and also involved in a murder, it would be strange to qualify this person as a thief, rather than a murderer.

Besides this domain specific metric we also looked at centrality rankings to determine important nodes. For each node we computed each of the five centralities described in Section 4.4. For each centrality measure we ranked the nodes by ordering them by their centrality value. We then looked at the top 20 of each centrality measure to see if we can find nodes of importance, that are highly ranked in multiple centrality measures. As these centralities are known to correlate[21], we expect some amount of overlap between the top ranked entities of this network. We find that when looking at the top five of each centrality ranking, five entities show up in more than one centrality ranking. Two entities even rank in the top 5 in three different measures. When looking at the top 10 we find that nine entities show up in more than one ranking, of which five show up in three different rankings. This tells us that even though some centralities are correlated in a way, there are in fact entities that could be interesting to investigate.

We also combined these methods of ranking entities, where we looked at centralities in comparison to case weight. This is to find out whether case weight correlates with centrality measures to determine if severe criminals are also more central in the criminal network of Amsterdam. Figure 9 shows how two commonly used centralities compare to a domain specific measure such as case weight.

In this figure the centralities are normalized to a range between 0 and 1. We see that the majority of nodes in this centrality range have a low crime weight as the majority of the points in the figure are blue. We see that the lefthand side of the figure is mostly blue and dark blue, meaning nodes with a very low degree are generally criminals with a low crime weight. This makes sense as criminals that are not deep in the criminal circuit yet and do not have as many connections, probably perform less severe crimes. We also see that when the betweenness of nodes rises this does not really correlate with crime weight as in this figure the nodes with higher betweenness are still mostly colored blue.

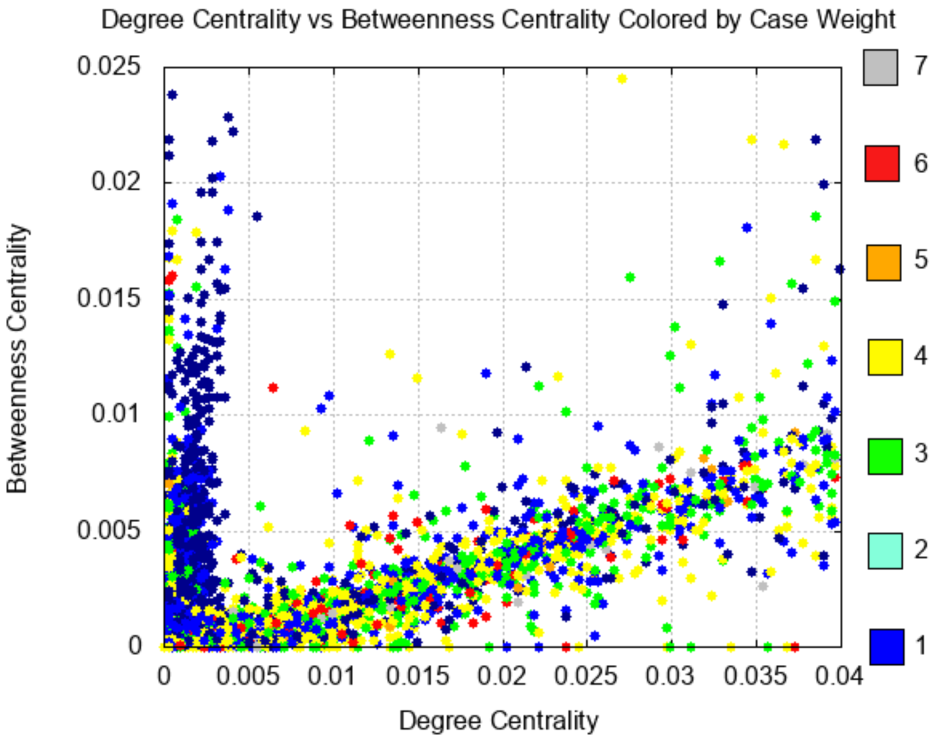

Figure 9: Degree centrality vs betweenness centrality, colored by case weight

There is a correlation visible in the figure, between degree centrality and betweenness centrality, as was also seen in the scatter plot matrix. The higher the degree of the node, the higher the betweenness of that node. And we also see that on the righthand side, where the degree and betweenness values start going up, the crime weight also goes up as that side shows more green, yellow, orange and red nodes.

When examining correlations between each of the centrality measures and the severity of crimes to see if more severe criminals are in fact more central in the network, we find that this is not necessarily the case. The results of Figure 9 showed a relation between crime weight and centrality in a small range of the two centralities (0 to 0.04 and 0 to 0.025). When looking at the relation between centralities and crime weight, a boxplot gives an accurate representation of this relation, as you can see the distribution of a centrality measure seperate for each crime weight in one figure. Figure 10 shows a similar pattern of distribution of the degree for each crime weight, other than for the crime weights of 5 and 7 where there are significantly less nodes with this crime weight and a high degree. This is probably due to the absence of nodes with crime weights of 5 and 7 that we saw in Figure 8. For the other centrality measures we see a similar trend. This is the case for four out of the five centralities we used in this research, for which boxplots can be found in the appendix, Figures 12, 13, and 14. For closeness centrality however, we found that nodes with a lower crime weight tend to have a higher closeness centrality, which can be seen in Figure 11. This means that people with a lower crime weight tend to have a smaller average shortest path length, meaning their distance to others in the network is shorter. This may indicate that the bigger criminals are hiding in remote parts of the network.

5.5 Identifying Key Players

As stated in Section 4.4.6, identifying key players that cause the network to fragment when removed are likely to have high betweenness centrality values. We experimented with removing the nodes that have the highest betweenness in this network and observed the results this had on the fragmentation of the network. For this we use two measures, the number of components that are in the network after removing this node or these nodes, and the impact it has on the average shortest path length of the network. These measures are specifically interesting in this field, as fragmenting a criminal network or increasing the difficulty of criminals contacting each other can have an impact on the criminal circuit.

As shown in Table 3, we found that even when removing a single node, this being the node with the highest betweenness, we already disconnect

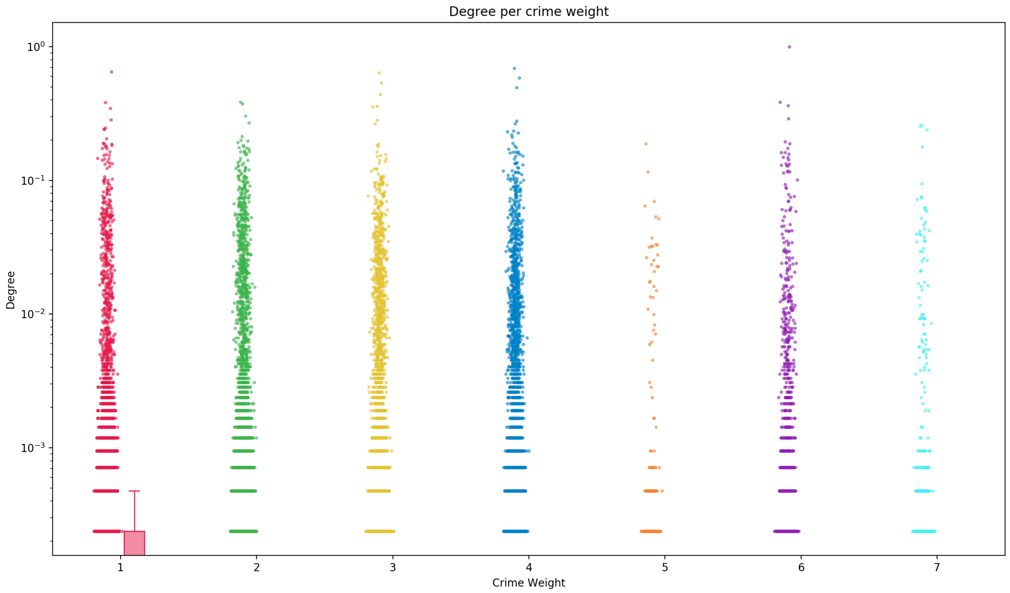

Figure 10: Boxplot of degree centrality vs crime weight

the network in such a way that it falls apart in five components, one large component and four small ones. When removing the ten nodes with the highest betweenness centrality, this shatters the network, resulting in 15853 components. Again, one main component and a lot of smaller components. After removing these nodes, we took a sample of this newly created network with 10.000 nodes and their respective edges and calculated the new average shortest path length. After taking the average of 10 approximations per new giant component, we see a minor increase in average shortest path length. This means that by removing a relatively small group of so called key players, the average shortest path length increases slightly, and the network falls apart into thousands of separate components.

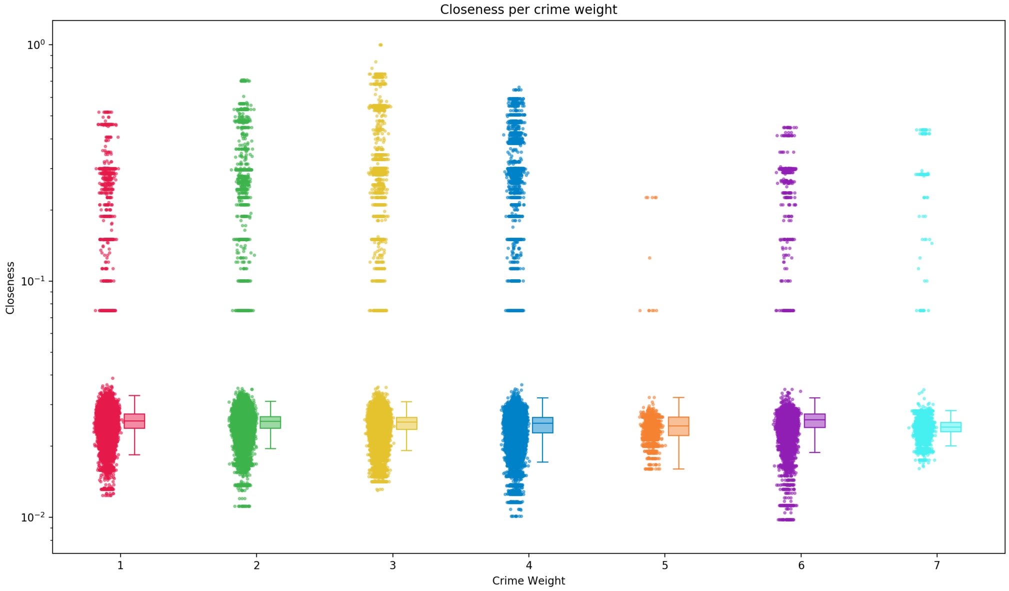

Figure 11: Boxplot of closeness centrality vs crime weight

Table 3: Fragmentation by removing key players

Top #nodes removed	#Components	#Nodes	#Edges	Path Length
0	1	333462	428591	8.18
1	5	333214	428200	8.20
10	15853	315912	406497	8.32
25	27993	303568	387170	8.35

5.6 Community Structure

To determine what the optimal division into communities is for a network using the Louvain Method, one has to experiment with the resolution parameter to see for which resolution the modularity is optimal. Table 4 shows that for a resolution of 1.25 we get the optimal value for modularity, with

190 communities for this network.

Table 4: Tuning of Resolution parameter

Resolution	#Communities	Modularity
0.50	343	0.91537
0.75	262	0.91662
1.00	215	0.91671
1.25	190	0.91705
1.50	172	0.91550
1.75	160	0.91458

For various nodes in this network we have information about the type of crime they are involved in. This means that when dividing the network in communities, we can investigate if these modularity based communities consist of people that perform the same type of crimes. In this way we can validate the network measure of community structure by using domain specific metadata. There's a total of 12 main crime types, with 675 subtypes. For this community analysis we will be looking at subtypes.
As shown in Table 5, we found that 75 of the communities have one crime type that covers the majority of the community. For 52 of the communities it is the case that 75% or more of the nodes in the community have the same crime type and 12 communities consist of only nodes with exactly the same crime type.

Table 6 shows that the type of crime that is most common in some communities, almost exclusively shows up only in that community. For 24 communities we find that their most common crime type has 90% or more of its occurrences in this community, meaning there is 10% or less of this crime type being performed in different communities. For 51 of the communities we find that their most occurring crime type occurs in this community for more than 50% of the total occurrences in the network. Meaning that the

Table 5: Crime distribution per community

% Same Crime Type	# Communities
>51	75
>75	52
100	12

majority of the occurrences of this crime type is in one community.

Table 6: Community distribution per crime

% Same Community	# Communities
>90	24
>51	51

5.7 Hierarchy

To determine if there is a hierarchical structure within a network, one can look at the assortativity of its nodes. Assortativity is described as a preference that nodes tend to attach to nodes that are similar in a way. In the case of a hierarchy, this is likely to be seen in the degree of nodes. Seeing as not all own numbers are known, it is likely that that the network is disassortative, as 'phone' nodes are connected to a lot of contacts and to fewer other phones, mainly resulting in connections between high degree nodes and low degree nodes. This network has an assortativity coefficient of -0.21 from which we can conclude that this network is disassortative. Low degree nodes are connected to high degree nodes and nodes of degree 50 and higher are rarely connected to other nodes of degree 50 and higher, but mainly to other very low degree nodes. This could be because of the 'phone'nodes being connected to the 'contact'nodes as described in Section 4.3. It could also be an indication of a hierarchical structure within the network, where not every criminal has contact with as many other criminals, but there are specific people who have contact with a lot of people and others who have very little contacts.

6 Conclusion & Discussion

In this work, we presented methods to analyze data of criminal contact using social network analysis to get insights on suspects and relations between cases. A lot of work was done in preprocessing the data and getting an accurate representation of the criminal network using this dataset. By doing this preprocessing and linking our dataset to additional metadata available by the police, we managed to identify a lot of nodes in the network that were the same person. Removing these nodes ensures the quality of our results.
We found that this criminal network has properties of a small world network, which it likely is, but due to absence of data it is missing links resulting in a low clustering coefficient.

We found there are no actors in this network that have a high centrality value for different centrality measures. Our analysis also showed that there is little correlation between centrality measures and the severity of crimes committed by actors in the network. This means we can identify potentially important targets for the police to arrest based on centrality measures and crime weight, but it is hard to automatically classify them in this way.
We found an effective way of breaking up a criminal network by locating key players that fragment the network when removed by choosing targets with high betweenness. This results in fragmentation and a significant increase in average shortest path length, hindering contact in the network.

We showed that community detection algorithms to some extent show similar results in grouping people based on their crime type. We found some communities solely consisting of people performing the same type of crime and some crime types that almost exlusively showed up in certain communities. Using these communities the police can find criminals in groups where they may not expect them to be, and create links between cases that were not previously present.
The network showed to have low assortativity, where there were a lot of nodes

connected to nodes with a very different degree, which can be explained due to the nature of the data, or by the presence of a hierarchical structure in the network.
Summarizing, we have researched entities, their relations, and relations between cases in a criminal network and found interesting information that could be used in future police investigations.

For future work, the data could be refined even more, as the results will get more realistic once more entity resolution is performed. An example of how this could be done is taking names of contacts in consideration, figuring out by which aliases criminals go and minimizing the number of nodes representing the same entity. Once temporal data is available, link prediction is also something that would be interesting to research for this data. As it would be valuable information to the police to know which criminals are likely to work with one another.

References

[1] Marin, A., & Wellman, B. (2011). Social network analysis: An introduction. The SAGE handbook of social network analysis, 11-25.

[2] Morselli, C., Gigure, C., & Petit, K. (2007). The efficiency/security trade-off in criminal networks. Social Networks, 29(1), 143-153.

[3] Christakis, N. A., & Fowler, J. H. (2010). Social network sensors for early detection of contagious outbreaks. PloS one, 5(9), e12948.

[4] Freeman, L. C. (1978). Centrality in social networks conceptual clarification. Social networks, 1(3), 215-239.

[5] Morselli, C. (2009). Inside criminal networks. New York: Springer.

[6] Adderley, R., Badii, A., & Wu, C. (2008). The automatic identification and prioritisation of criminal networks from police crime data. Intelligence and security informatics, 5-14.

[7] Sparrow, M. K. (1991). The application of network analysis to criminal intelligence: An assessment of the prospects. Social networks, 13(3), 251-274.

[8] Oatley, G., & Crick, T. (2014, August). Measuring UK crime gangs. In Advances in Social Networks Analysis and Mining , 2014 IEEE/ACM International Conference on (pp. 253-256). IEEE.

[9] Borgatti, S. P. (2006). Identifying sets of key players in a social network. Computational & Mathematical Organization Theory, 12(1), 21-34.

[10] Rhoades, S. A. (1993). The herfindahl-hirschman index. Federal Reserve Bulletin, 79, 188.

[11] Getoor, L., & Machanavajjhala, A. (2012). Entity resolution: theory, practice & open challenges. Proceedings of the Very large Data Bases Endowment, 5(12), 2018-2019.

[12] Otte, E., & Rousseau, R. (2002). Social network analysis: a powerful strategy, also for the information sciences. Journal of information Science, 28(6), 441-453.

[13] Real, R., & Vargas, J. M. (1996). The probabilistic basis of Jaccard's index of similarity. Systematic biology, 45(3), 380-385.

[14] Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Stanford InfoLab.

[15] Bonacich, P. (2007). Some unique properties of eigenvector centrality. Social Networks, 29(4), 555-564.

[16] Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. Journal of statistical mechanics: Theory and experiment, 2008(10), P10008.

[17] Hagberg, A., Swart, P., & S Chult, D. (2008). Exploring network structure, dynamics, and function using NetworkX. Proceedings of the 7th Python in Science Conference.

[18] Takes, F.D., teexGraph, 2016, GitHub repository
https://github.com/franktakes/teexgraph, Accessed July 2017

[19] Bastian, M., Heymann, S., & Jacomy, M. (2009). Gephi: an open source software for exploring and manipulating networks. International conference on web and social media, 8(2009), 361-362.

[20] Kolda, T. G., Pinar, A., Plantenga, T., Seshadhri, C., & Task, C. (2014). Counting triangles in massive graphs with MapReduce. Society for Industrial and Applied Mathematics Journal on Scientific Computing, 36(5), S48-S77.

[21] Valente, T. W., Coronges, K., Lakon, C., & Costenbader, E. (2008). How correlated are network centrality measures?. Connections (Toronto, Ont.), 28(1), 16.

Appendix

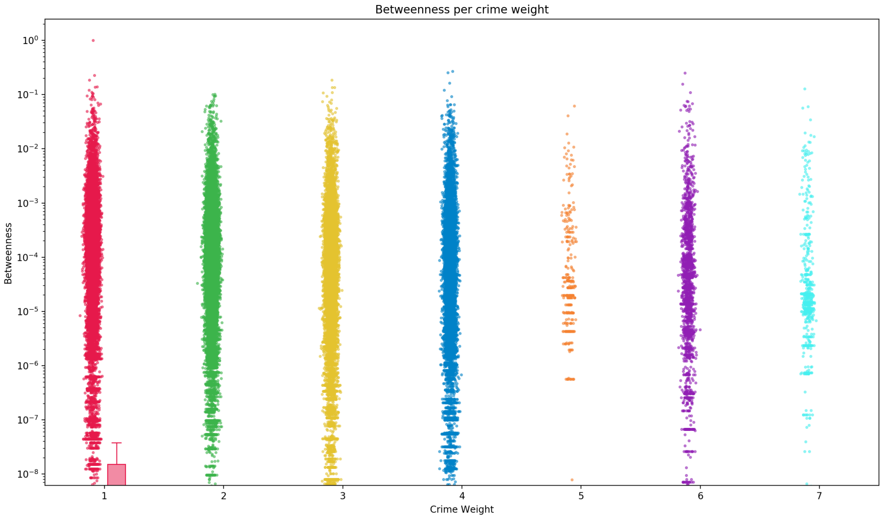

Figure 12: Boxplot of betweenness centrality vs crime weight

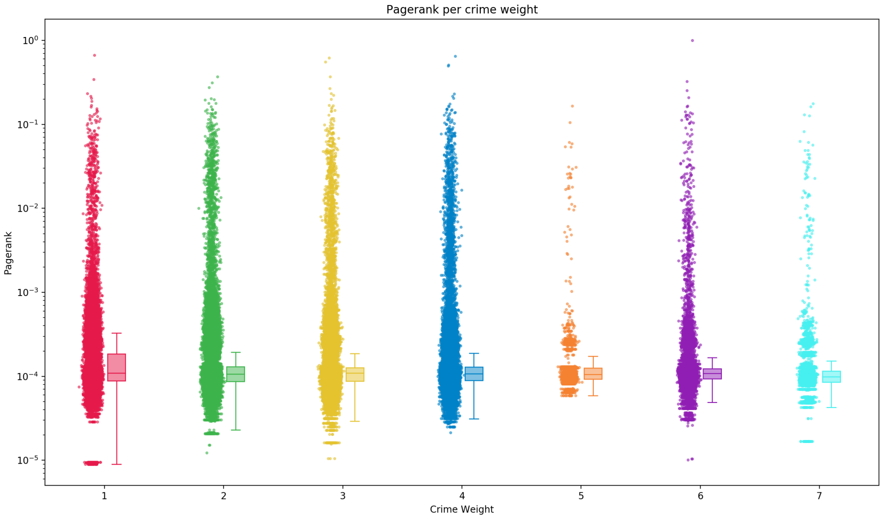

Figure 13: Boxplot of PageRank centrality vs crime weight

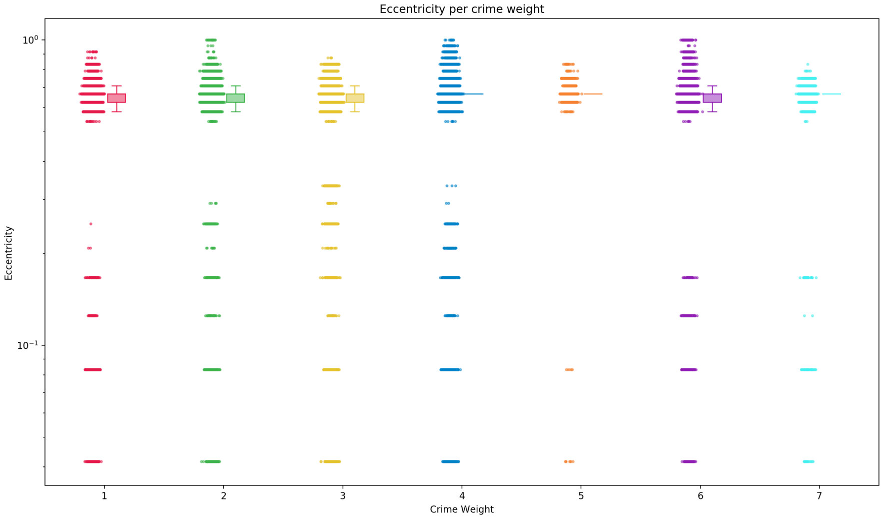

Figure 14: Boxplot of eccentricity centrality vs crime weight