DS6501

A2 Social Data Analytics

Dr Abubakar Siddique

Andrew Graff 2231290

Executive summary

We are dealing with Social Network Analysis and exploring the dynamics of interactions among members of a London street gang. We discover links by using analysis techniques to better understand relationships and connections within a network, identifying patterns that provide us with a good insight into network behaviors, trends, and associations. We delve into the fields of measuring statistics and calculations supported by visual representation of data adding weight to our social data analysis.

Table of Contents

T1

a. We have successfully loaded the two data sets of nodes and links (StreetGang).

b. We have created an igraph object based on the data files.

c. Inspect the attributes of the network and describe nodes and links.

There are 315 edges (indicating the links between nodes) We can see weight and width

```
> edge_attr(StreetGang_net) #examine edge (links) attributes
$weight
  [1] 1 1 2 1 1 2 3 2 2 3 1 1 1 2 2 3 1 1 1 1 1 1 1 1 1 3 2 1 2 1 2 2 1 1 3 2 1 1 2 2 1 1 3 2 2 2 4 4
 [49] 3 1 3 2 2 1 1 3 1 1 1 1 1 2 2 2 2 2 4 3 1 1 2 3 1 2 2 1 1 2 1 1 1 1 1 1 1 3 1 1 1 1 3 3 1 2 2 1
 [97] 1 2 1 1 1 1 1 2 3 1 2 2 1 1 2 1 1 2 2 1 1 1 2 1 1 3 2 1 1 1 1 2 1 1 1 2 2 1 1 1 1 1 1 1 1 2 4 1
[145] 1 1 2 1 3 3 2 1 2 2 1 1 1 1 1 1 2 1 2 3 2 1 1 2 2 2 1 1 1 2 2 1 1 1 2 1 1 1 1 3 1 1 2 1 1 1 1 1
[193] 1 1 2 1 1 1 3 2 1 1 4 2 1 1 1 1 1 1 2 2 2 2 2 2 1 2 3 2 2 4 2 2 2 3 2 1 2 1 1 2 1 1 1 1 1 1 2 2
[241] 2 1 1 1 1 1 1 3 1 1 1 2 1 1 1 1 1 1 1 1 4 4 4 1 1 1 1 2 1 4 4 1 1 4 1 1 1 1 1 1 1 2 1 1 1 4 4 1
[289] 1 4 2 3 2 1 1 1 1 1 1 1 2 2 1 4 1 1 2 2 2 2 2 2 1 2 1

$width
  [1] 1 1 2 1 1 2 3 2 2 3 1 1 1 2 2 3 1 1 1 1 1 1 1 1 1 3 2 1 2 1 2 2 1 1 3 2 1 1 2 2 1 1 3 2 2 2 4 4
 [49] 3 1 3 2 2 1 1 3 1 1 1 1 1 2 2 2 2 2 4 3 1 1 2 3 1 2 2 1 1 2 1 1 1 1 1 1 1 3 1 1 1 1 3 3 1 2 2 1
 [97] 1 2 1 1 1 1 1 2 3 1 2 2 1 1 2 1 1 2 2 1 1 1 2 1 1 3 2 1 1 1 1 2 1 1 1 2 2 1 1 1 1 1 1 1 1 2 4 1
[145] 1 1 2 1 3 3 2 1 2 2 1 1 1 1 1 1 2 1 2 3 2 1 1 2 2 2 1 1 1 2 2 1 1 1 2 1 1 1 1 3 1 1 2 1 1 1 1 1
[193] 1 1 2 1 1 1 3 2 1 1 4 2 1 1 1 1 1 1 2 2 2 2 2 2 1 2 3 2 2 4 2 2 2 3 2 1 2 1 1 2 1 1 1 1 1 1 2 2
[241] 2 1 1 1 1 1 1 3 1 1 1 2 1 1 1 1 1 1 1 1 4 4 4 1 1 1 1 2 1 4 4 1 1 4 1 1 1 1 1 1 1 2 1 1 1 4 4 1
[289] 1 4 2 3 2 1 1 1 1 1 1 1 2 2 1 4 1 1 2 2 2 2 2 2 1 2 1
```

	name	Age	Birthplace	Residence	Arrests	Convictions	Prison	Ranking
1	1	20	1	0	16	4	1	1
2	2	20	2	0	16	7	1	2
3	3	19	2	0	12	4	1	2
4	4	21	2	0	8	1	0	2
5	5	24	2	0	11	3	0	2
6	6	25	3	1	17	10	0	2
7	7	20	4	1	8	1	0	2
8	8	21	1	0	15	6	1	3
9	9	20	1	1	9	3	0	3
10	10	23	1	1	12	4	1	3
11	11	21	1	0	16	8	1	3
12	12	25	3	0	5	3	0	3
13	13	21	3	0	19	9	1	3
14	14	19	3	0	23	9	0	3
15	15	21	3	0	12	9	1	4
16	16	19	3	0	14	7	1	4

There are 54 nodes, indicating 54 people and we have 8 columns each indicating significant information about the node.

There are **Name, Age, Birthplace, Residence, Arrests, Convictions, Prison and Ranking**.

The later 4 columns have a number representing a count (amount of times)

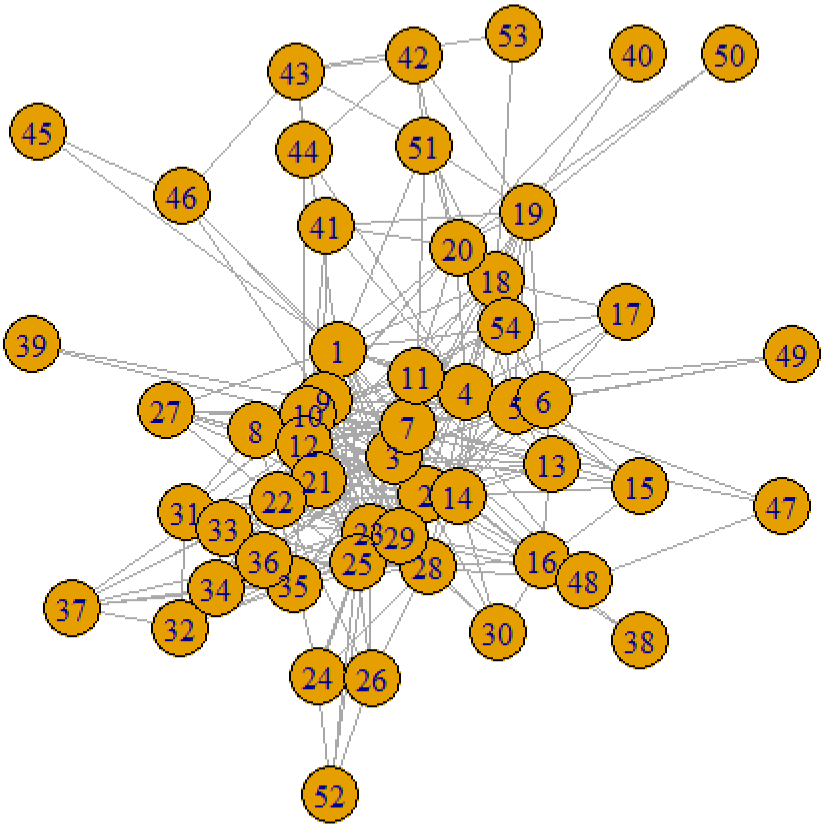
There are number values as attributes. The name column is numbers to a total of 54 entries. The age is the actual age, birthplace and residence is represented by a number.

Attributes.txt: Elaborates on information about the birthplace, residence and weight.

```
Link Attribute: Weight  = 1 (friends),
= 2 (co-offend together)
                        = 3 (co-offend together, serious crime)
                        = 4 (co-offend together, serious crime, kin)

Node Attributes: Age, Birthplace (1 = West Africa, 2 = Caribbean, 3 = UK, 4 = East African), Residence (1 = resident of housing estate
0 = non-resident), Arrests (number of arrests), Convictions (number of convictions), Prison (1 = served time in prison, 0 = has not
served time), Ranking (value of 1 to 5 - assigned by police where 1 is the highest ranking).
```

d. Plot network using default settings and describe main problem with this plot in terms of it's readability.



There are many numbers.

Lines are not significant to see relationships between the nodes and links.

It is quite scrambled and clutter.

Vertices are the end points (nodes)

Edges are the links (arrows)

We learned there are 315 edges also known as links and also 54 vertices known as nodes.

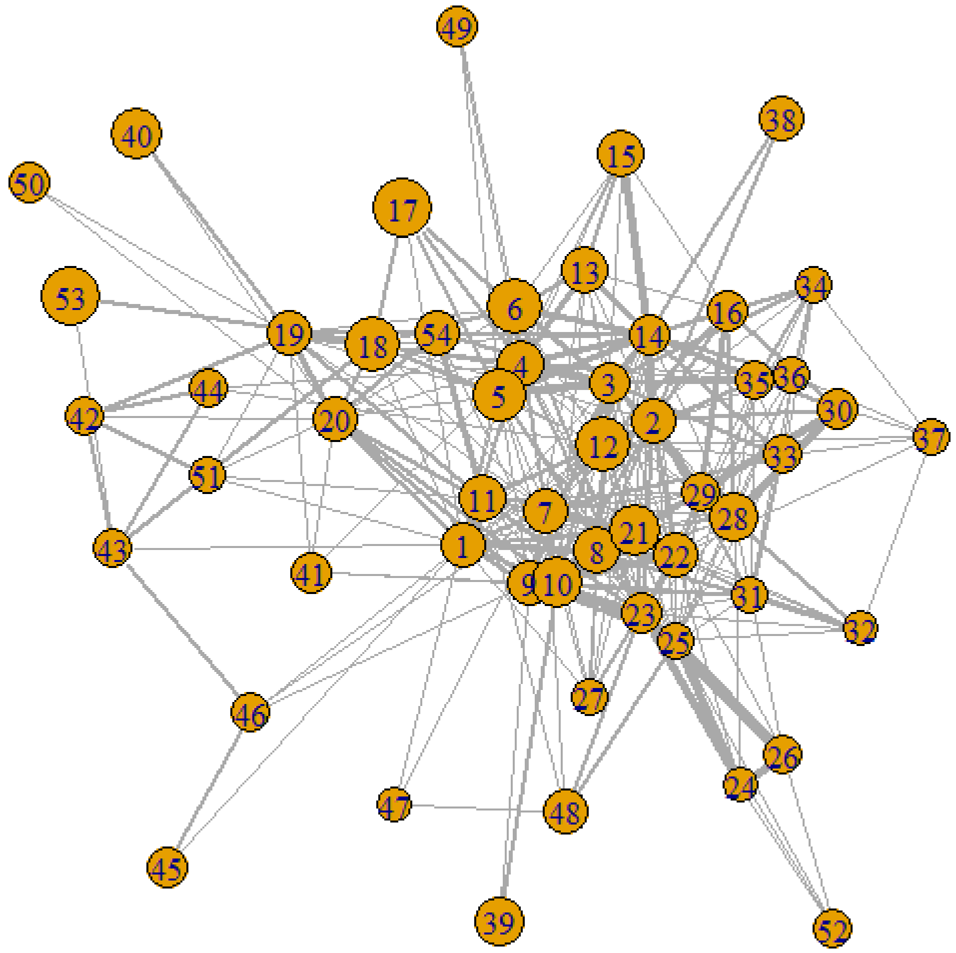
According to this data, each node (number) we see represents an individual and the many links between each node establishing a connection of some sort.

e. Plot the network again using the following attributes settings:

Widths of the links between nodes set to the value of their weight attribute

Size of each node should equal to the age attribute divide by 2

(We are given the below network plot graph)



Describe how the network plot has improved?

We can see the edges have spread out by showing length between other nodes. The further the links between nodes the greater the weight as indicated in the code.

According to the attributes file

```
Link Attribute: Weight  = 1 (friends),
= 2 (co-offend together)
                        = 3 (co-offend together, serious crime)
                        = 4 (co-offend together, serious crime, kin)
```

We have an understanding the higher the weight therefore the thicker the width of the links between nodes. As per information it means 4 (co-offend together, serious crime and kin). The size of the nodes has changed as per coding around setting node size to age attribute and diving this by 2. Seeing the network plot now, we get a fair idea of the age just by measuring size of nodes compared to other nodes. For example, node 17 must be older than node 24.

T2.

a. Explore measures of centrality within the network by calculating the degree, betweenness, and closeness measures of each node

```
> degree(StreetGang_net)
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
25 22 22 21 19 16 25 15 21 22 18 25 11 24  7  8  5 14 13 15 19 24 23  6 23  6  8 18 17  5 12  7
33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54
10  8 12 12  7  2  2  2  5  5  6  4  2  5  3  5  3  2  7  4  2  6
```

```
> betweenness(StreetGang_net, directed=T, weights=NA)
           1            2            3            4            5            6            7
149.74463684  59.03602715  43.59225557  99.92031728  66.18359123  34.42306577 104.03338170
           8            9           10           11           12           13           14
  3.42991526  66.50302003  66.62109209  47.63484040  82.10542905   5.74201575  88.22854563
          15           16           17           18           19           20           21
  1.42850172   4.04833083   0.00000000  61.33552039  65.17993124  87.75095445  14.80113736
          22           23           24           25           26           27           28
 62.01336149  76.81906046   0.98876263  76.75829597   0.98876263   0.30931656  37.90985242
          29           30           31           32           33           34           35
 19.17126619   0.09090909   6.37217580   1.97116674   5.04441876   1.00815557   6.85400527
          36           37           38           39           40           41           42
  6.85400527   1.79700344   0.00000000   0.00000000   0.00000000   0.88181818   4.75555556
          43           44           45           46           47           48           49
 13.23370799  10.21166759   0.00000000   6.93367733   0.54394552   2.91328527   0.00000000
          50           51           52           53           54
  0.00000000   8.12601857   0.00000000   0.38596491   3.32133104
```

```
> closeness(StreetGang_net, mode="all", weights=NA)
          1           2           3           4           5           6           7           8
0.012345679 0.011494253 0.011363636 0.011627907 0.011494253 0.010989011 0.012345679 0.010416667
          9          10          11          12          13          14          15          16
0.011627907 0.011764706 0.010989011 0.012048193 0.009345794 0.011363636 0.008620690 0.008849558
         17          18          19          20          21          22          23          24
0.008474576 0.010309278 0.009615385 0.009900990 0.010869565 0.011494253 0.011494253 0.008000000
         25          26          27          28          29          30          31          32
0.011494253 0.008000000 0.009615385 0.010752688 0.010638298 0.008130081 0.009345794 0.008130081
         33          34          35          36          37          38          39          40
0.009259259 0.008547009 0.009615385 0.009615385 0.008064516 0.007751938 0.007407407 0.006711409
         41          42          43          44          45          46          47          48
0.008620690 0.007194245 0.008064516 0.008547009 0.007575758 0.008620690 0.007936508 0.008474576
         49          50          51          52          53          54
0.007633588 0.006711409 0.008547009 0.007518797 0.006944444 0.008849558
```

b. Repeat analysis performed in (a) to display nodes in descending order of their value for each centrality measure.

```
> order(degree(StreetGang_net), decreasing=TRUE)
 [1]  1  7 12 14 22 23 25  2  3 10  4  9  5 21 11 28 29  6  8 20 18 19 31 35 36 13 33 16 27 34
[31] 15 32 37 51 24 26 43 54 17 30 41 42 46 48 44 52 47 49 38 39 40 45 50 53
> order(betweenness(StreetGang_net, directed=T, weights=NA), decreasing=TRUE)
 [1]  1  7  4 14 20 12 23 25 10  9  5 19 22 18  2 11  3 28  6 29 21 43 44 51 46 35 36 31 13 33
[31] 42 16  8 54 48 32 37 15 34 24 26 41 47 53 27 30 17 38 39 40 45 49 50 52
> order(closeness(StreetGang_net, mode="all", weights=NA), decreasing=TRUE)
 [1]  1  7 12 10  4  9  2  5 22 23 25  3 14  6 11 21 28 29  8 18 20 19 27 35 36 13 31 33 16 54
[31] 15 41 46 34 44 51 17 48 30 32 37 43 24 26 47 38 49 45 52 39 42 53 40 50
```

Identify the top 3 nodes with the highest value for each centrality measure. (Centrality is the focus of behaviour of the nodes in network)

```
> topdeg
   name degree betweeness  closeness
1     1     25  149.74464 0.01234568
7     7     25  104.03338 0.01234568
12   12     25   82.10543 0.01204819
> topbet
  name degree betweeness  closeness
1    1     25  149.74464 0.01234568
7    7     25  104.03338 0.01234568
4    4     21   99.92032 0.01162791
> topclo
   name degree betweeness  closeness
1     1     25  149.744464 0.01234568
7     7     25  104.03338 0.01234568
12   12     25   82.10543 0.01204819
```

Degree: Node 1 (24)

Betweenness: Node 1 (149)

Closeness: Node 1 (0.01)

Node 1 with the highest degree score of 25 tell us this person has many links/hops to other nodes in the network. This is a highly connected individual.

Node 1 with the highest betweenness score of 149.74464 tells us the number of times this node lies on the shortest path between other nodes. So, we can identify this node as influencing the flow of information around a system.

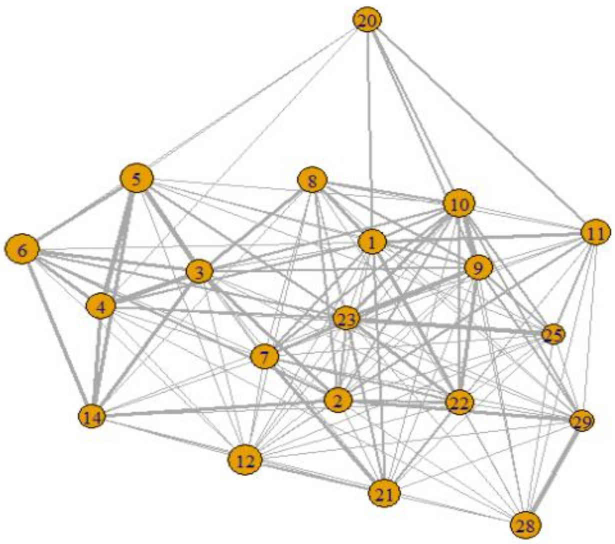
Node 1 with the highest closeness score of 0.1234568 tells us this node is more closer to all the other nodes. This is a total sum of all shortest paths (Distance between other nodes in the network).

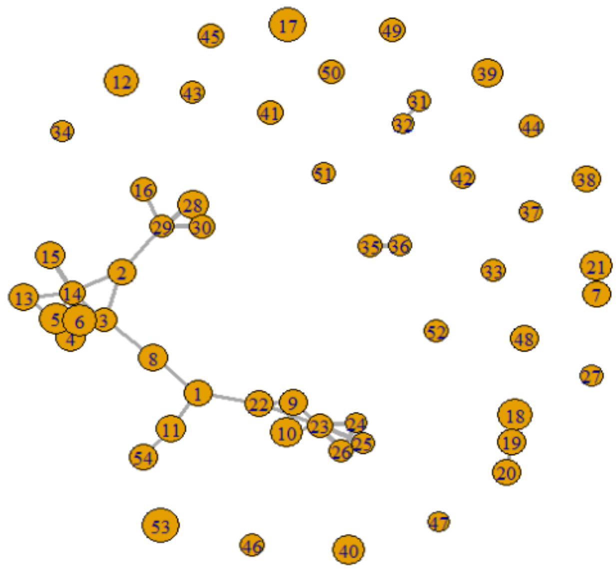
These three nodes could be of interest to law enforcement due to their links with other nodes based on connections, flow of information and close distance between. In highly organized crime, a high degree of links would be of great interest.

T3

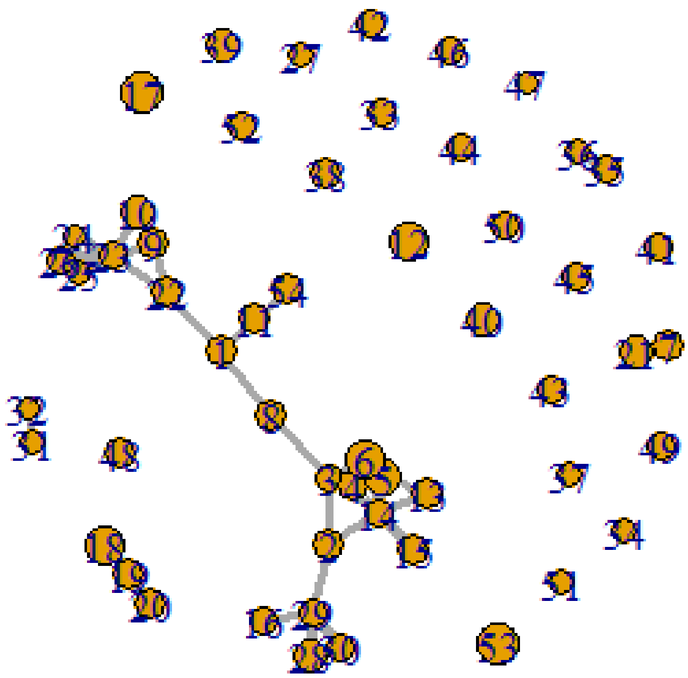
a. Simplify the network by

Removing nodes with a degree less than 15   Removing edges with a weight less than 3





Plot adjusted network 'layout_nicely'



(b) Briefly describe the network plotted in (a).

Your description should discuss the groupings within the network and you should identify nodes that act as bridges between these groups.

We have simplified the network from its original network data plot where there were many links and nodes. Removing nodes with a link less than 15, has given us readability in contrast of being able to identify the stronger nodes with connections/links to other nodes.

Based on this analysis and analysis performed in T2, which node is likely to be the most influential gang member.

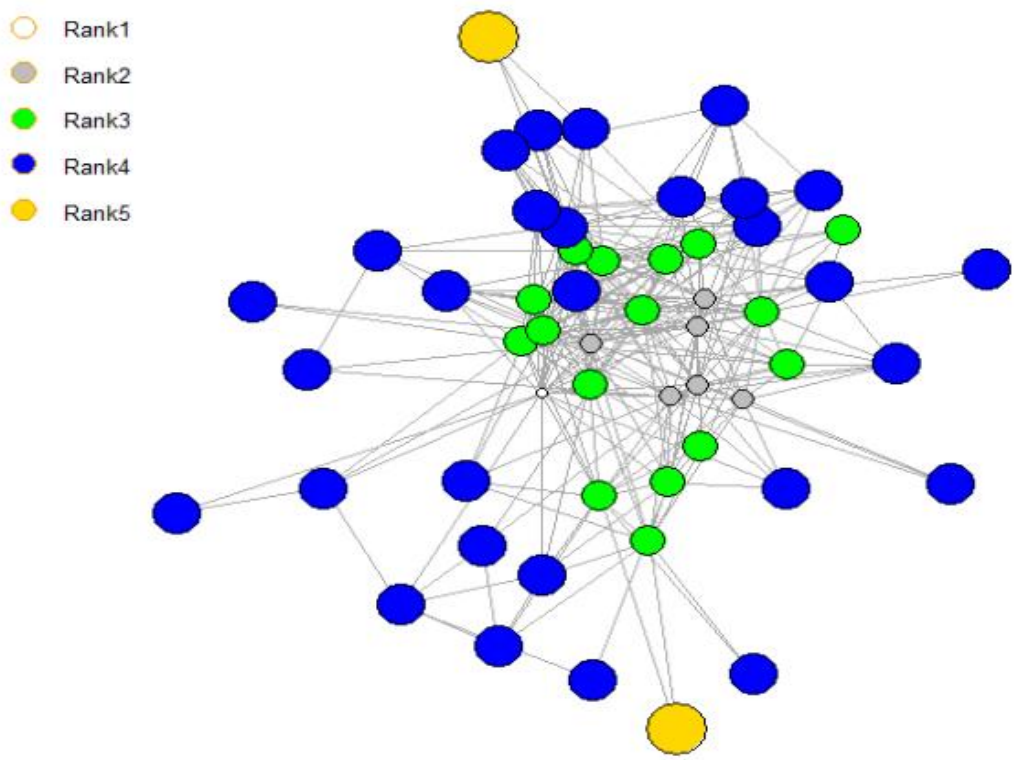
The most influential gang member is node 1. Our analysis from attaining the highest scoring for degree, betweenness and closeness provided us with the information that node 1 was top across these three areas.

The above analysis plotting the network has supported this scoring as we can visually see node 1 being connected to other nodes (appearing as bridge). This network was simplified by removing links of less than 15 (so only strong linked nodes remain) and keeping only nodes with weight of 4 (removing weight less than 3).

This means the connected nodes we see in the image above are all linked by co-offending together, serious crime and kin.

T4

(a) Set the colour of each node in the network based on the ranking attribute.
- Plot this network



Briefly describe the pattern observed in terms of the placement of nodes with the same ranking

The pattern observed from the placement of the nodes in the same ranking is that they are closely bundled together.

The highest-ranking nodes (Rank 1: White) is located in the center of the network. From this observation, we can see there are only two nodes with a Rank 5 (Gold) that are located at opposite ends of the network these are the lowest ranked nodes. The second highest ranking Rank 2 (Gray) node is compiled in the center area of the network barricaded by nodes of Rank 3 (Green) and Rank 4 (Blue)

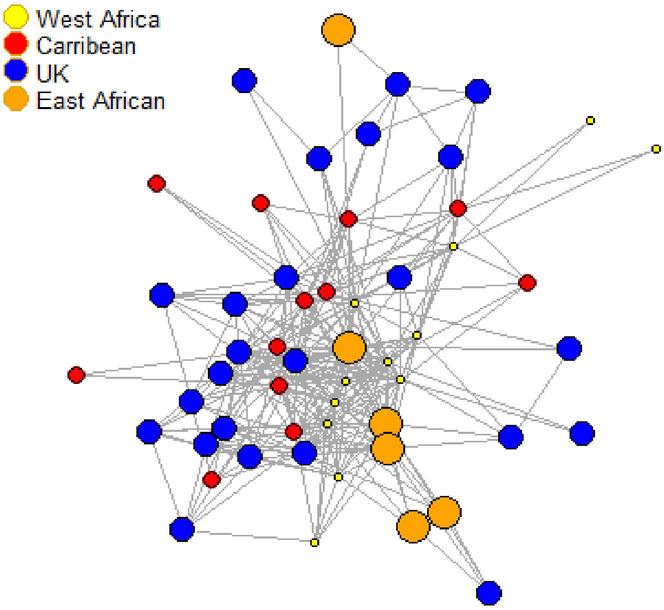
How does a gang member's ranking (assigned by Police) appear to relate to the degree of the nodes and the seriousness of co-offending with other gang members.

The gang member's ranking relates to the degree of the nodes by a similar measurement and that is the person with the highest links. In organized crime, it is often the highest rank/crime leader to give commands and communicate at a large capacity to other people. This relates to the degree of the nodes as for the node with most links to other nodes.

The ranked nodes as depicted in the graphs formulate a hierarchy structure.

It is interesting because we can see one white node (Rank 1) that is in the middle and very far from the Rank 5 nodes. As with businesses/organizations it is common for a call handler to be in a different room than the CEO.

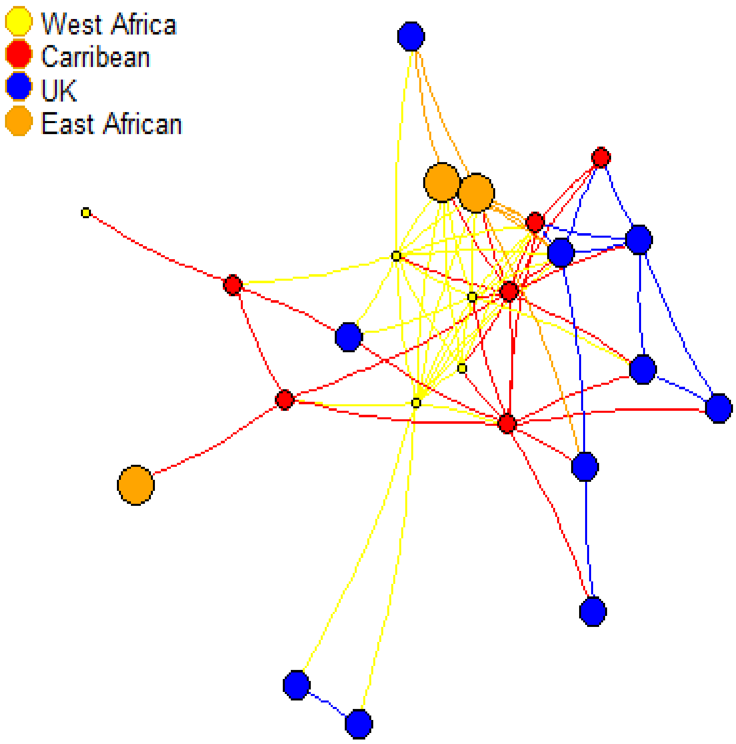
(b) Using the network created in T1, reassign the colour of each node within the whole street gang based on birthplace.



Simplify this network so it includes only those gang members who have served time in prison.

Plot this network and describe the interactions observed between gang members. Your description should discuss evidence of any grouping based on ethnicity and whether gang members of differing ethnicity interact with one another

A legend should be added to explain the colour coding you have defined



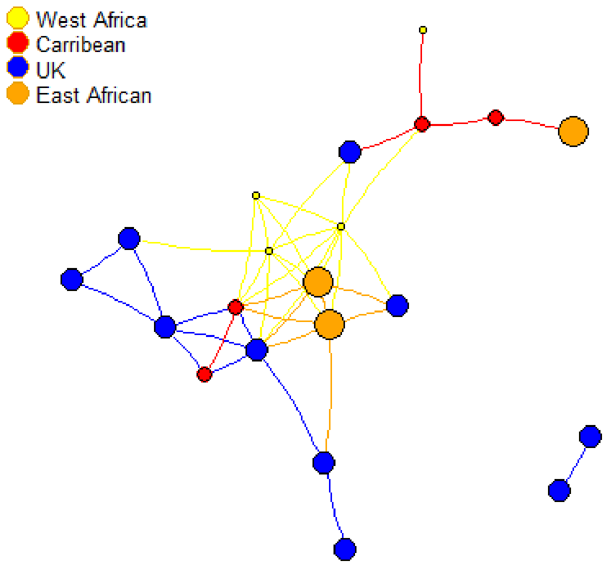
As per image to the left, we can only identify nodes of birthplace and time served in prison.

UK being a dominant place of birth (Blue 10), followed by Carribean (Red 6), West Africa (Yellow 5) and lastly East African 4 (Orange 3).

The interactions and groupings observed here from the birthplace (interpreted as ethnicities) are mixed. It does appear UK interacts with UK and Caribbean. West Africa with West Africa, has ties with Carribbean, and only few UK.

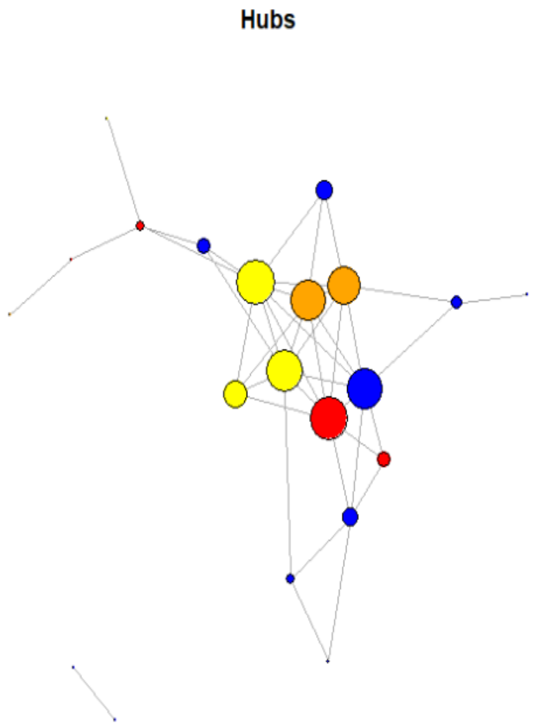
East Africa smaller in size with very little connections.

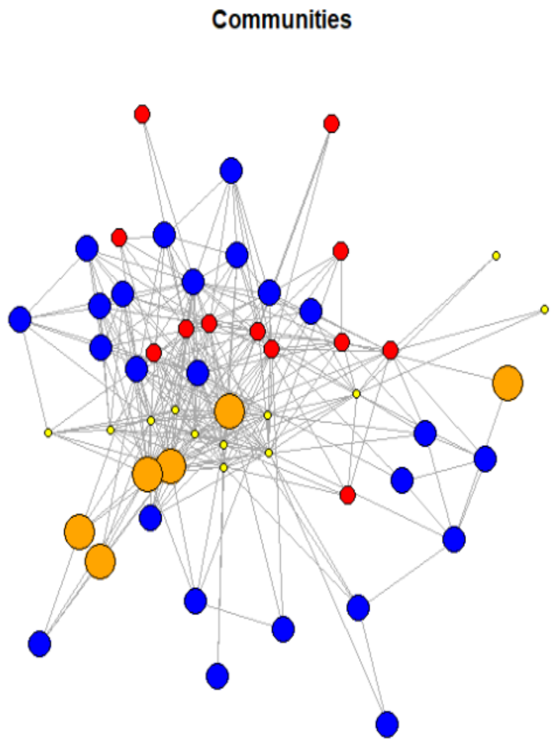
(c) Using the network created in (b) delete nodes where gang member ranking is less than 3.



Now determine the hub score of gang members within this network. Using this network create a two-panel plot. In the first panel, plot the network where the size of each node is set to 15 times the value of the hub score. In the second panel, display the communities within the network using the cluster_optimal() function.

Describe these two communities in terms of their ethnicity and also in terms of their interactions (i.e. friends, co-offenders etc.), both within each community and between these two communities.





We can see two clusters. The hub is significantly proportionately smaller to the right. Hub has one large red node with a few yellow nodes and blue nodes surrounding the network. These are reflected by the scoring. We can see the higher scoring nodes yellow, red, and blue on the left. On the right is red and second blue with a few smaller yellow.

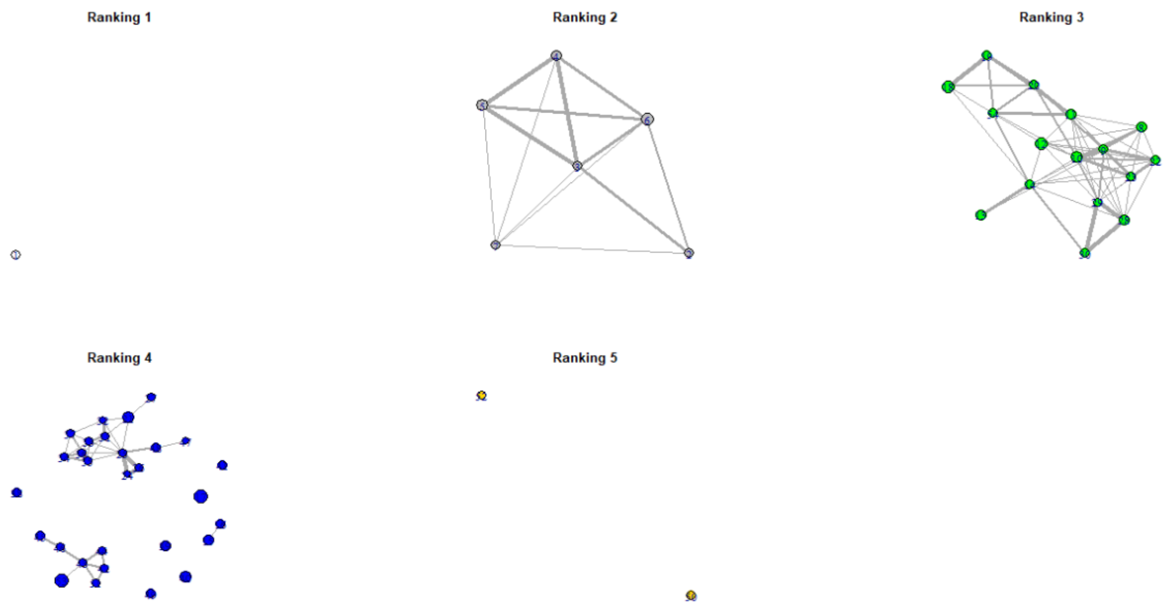
This could indicate the red node has close ties with neighboring nodes compared to the rest of the network. On the right, the community's network is presented with a large cluster of connections and can see many blue nodes and large green slightly centered amongst the group.

The large green nodes could be of some seniority being at center, of means of connections to other node links. Yellow nodes are quite dense in the middle and the smallest of the network. This indicates the yellow nodes are tightly knit and could be family/close friends. Red could be associated as they are located only on one side of the network.

T5.

Simplify the network created in T1 based on the Ranking attribute. Create 5 networks in total – one for each ranking. Plot each network.

Discuss whether you generally agree that the assigned ranking value reflects the serious of the co-offending committed among gang members within each network.



Based on these 5 networks of different Ranks, we cannot agree from these networks that the assigned ranking value provides us with an interpretation of the seriousness of co-offending among each gang members from the network.

Ranking 4 has a dense but spread out network and the highest Ranking 1 is distanced with no links. There is only one Rank 1, a few more in Ranking 2 and 3 which shows a connection to everyone. Some thin edges to indicate connections are friends and thick to show co-offending and serious crime.

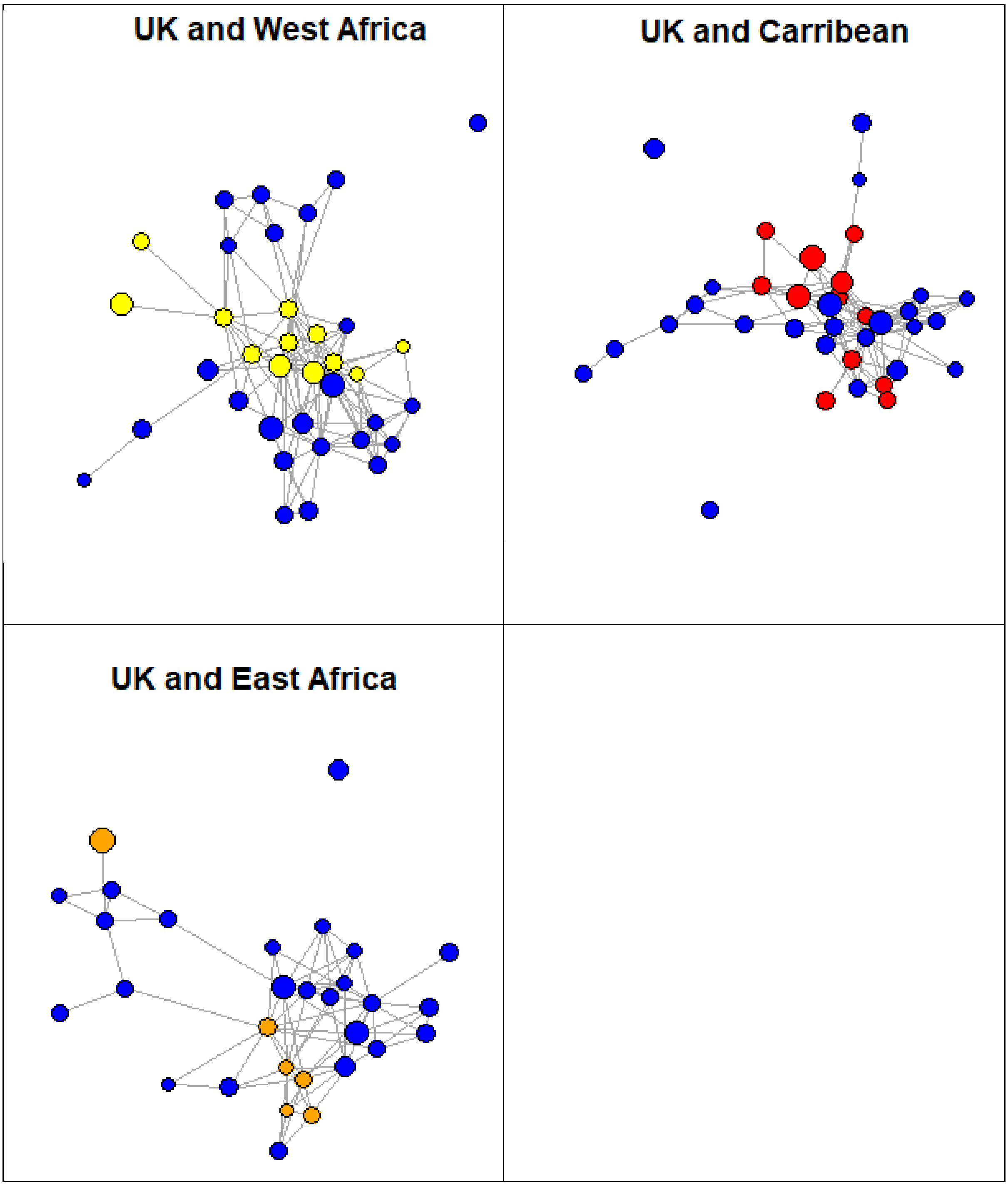
Ranking 2 has a thick edges between other nodes, this tells us the connections between the nodes are co-offending, serious crime, and prison.

Ranking 5 is distance with no edges/no links which is similar to Ranking 1. This adds to our analysis as the ranking value does not reflect the network diagram. Ranking 1 is highest rank and Ranking 5 is lowest rank however appearing the same in our diagram.

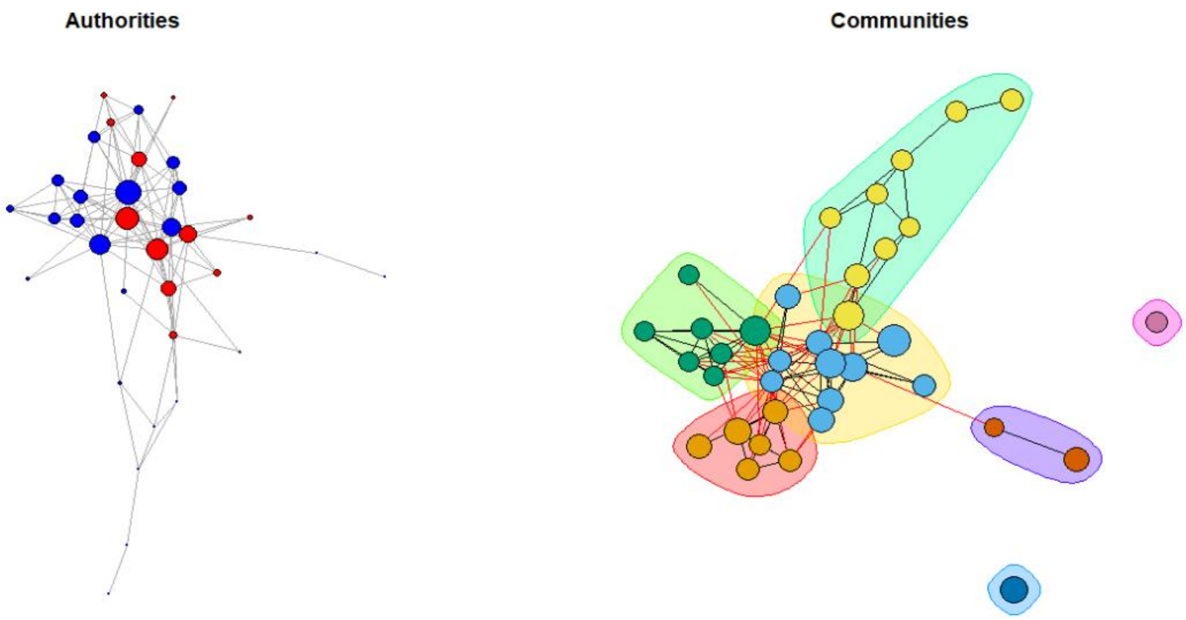
T6.

(a) Create three networks that show the criminal interactions between UK gang members and each other ethnicity i.e West African and UK, Caribbean and UK and finally, East African and UK. Use the network created in T1 as the initial starting point. UK blue.

Remove links with a weight value equal to 1 (i.e. friends) then plot each network.



(b) Using the network created in (a) for UK and Caribbean interactions, calculate the Authority scores of gangs members. Set the size of each node to 10 times the value of authority score. Create a two panel plotting window and plot this network in the first panel. Now identify the communities with this network using the cluster_optimal() function and plot these communities within the second panel



(c) Based on the networks created in (a) and (b), what evidence is there that supports the hypothesis of the research paper (available on Moodle) that co-offending occurs mostly among gang members of the same ethnicity? Is there evidence contrary to this hypothesis? You should ignore any isolates and nodes too small to distinguish their colour coding within the first panel plot in (b)

The evidence we can see here which supports the hypothesis, is the grouping of clusters by allocation of nodes of birthplace and ethnicity. We can see yellow bundled together, green, red, blue, and orange in their own territorial areas.

We can see the hubs with high scores

Cluster optimal is an analysis tool that helps identify the groupings of these ethnic backgrounds. We see the distinct groups in the network that are more connected to other groups. In the plotted graph, the UK is at the center with ties as per intersecting fields to east/west Africa and Caribbean.

Contrary to the hypothesis, the analysis performed using the various different methods has concluded substance towards the theory of co-offending occurring does happen mostly among gang members of the same ethnicity.

Conclusion

We have used a variety of practical applications to better understand the social structure data of a London Street Gang. Using igraph had helped visualize these networks to see the interactions and plotting of social structures with nodes and vertices. Recycling data to enable adjustments for clarity and discovery has overall signified key patterns. The most important and highly influential node was 1, followed by 7 and 12. During analysis, the bigger picture was observed and seen how the London Street Gang did not only associate with themselves, but select few. The second biggest ethnic group was West Africa.

The rankings were interesting as the highest rank was at the centre of network connections and the lowest rank were outside the network.

This concluded the passing of information, connection to other nodes of similar attributes and grouping using the cluster optimal tool.

FINAL SCRIPT

```
#ASSIGNMENT 2 sOCIAL NETWORK ANALYSIS

#T1

# Installing packages("igraph")
library(igraph) # Load the igraph package
library(igraphdata)
library(dplyr)

Gnodes <- read.csv("StreetGangNodes.csv") # Importing the data sets into R Studio
Glinks <- read.csv("StreetGangLinks.csv")

head(Gnodes) # Examine the data Vertices nodes
head(Glinks) # Examine the edges links

# Creating an igraph object based on the data files
StreetGang_net <- graph_from_data_frame(d=Glinks, vertices=Gnodes, directed=F) #Directed =
F (for false so undirected indicated by no arrows)

# Inspecting the attributes of the network
E(StreetGang_net) #(315) edges also links
V(StreetGang_net) #(54) vertices also nodes
edge_attr(StreetGang_net) #examine edge (links) attributes
vertex_attr(StreetGang_net) #examine names vertices (nodes)
V(StreetGang_net)$Age #check attributes

#The following commands extract an edge list and an adjacency matrix from our igraph network.
as_data_frame(StreetGang_net, what="edges") #as_edgelist(StreetGang_net, names=T)
as_data_frame(StreetGang_net, what="vertices") #as_adjacency_matrix(StreetGang_net,
attr="weight")

plot(StreetGang_net)#plotting simple network

#Widths of the links (edges) between nodes set to the value of weight attribute
E(StreetGang_net)$width <- E(StreetGang_net)$weight
V(StreetGang_net)$size <- V(StreetGang_net)$Age/2 #Size of each node equal age attribute
divide by 2

plot(StreetGang_net) #adjusted network with above attributes and specifications
#========================================================================
====

#T2
#(a) Calculate degree, betweenness and closeness measures of each node

degcent<-degree(StreetGang_net) #number of links each node has
betcent<-betweenness(StreetGang_net, directed=T, weights=NA)
clocent<-closeness(StreetGang_net, mode="all", weights=NA)
```

```
#saving above data to dataframe
centdf <-
data.frame(name=names(degcent),degree=(degcent),betweeness=betcent,closeness=clocent)

#(b) repeat analysis performed in (a) to display nodes in descending order of their value for each
centrality measure.
# identify the top 3 nodes with the highest value for each centrality measure.

order(degree(StreetGang_net), decreasing=TRUE) #attaining order of degree
order(betweenness(StreetGang_net, directed=T, weights=NA), decreasing=TRUE)
order(closeness(StreetGang_net, mode="all", weights=NA), decreasing=TRUE)

degsort<-centdf[order(-centdf$degree),] #able to order and create df for centrality values
betsort<-centdf[order(-centdf$betweeness),] #specifying the attribute value
closort<-centdf[order(-centdf$closeness),]

topdeg <-head(degsort,3) #sorting the top scoring values of our centrality measure
topbet <-head(betsort,3) #top three numbers
topclo <-head(closort,3)

#Highest scoring nodes in the three different areas (degree, betweness, closeness)
topdeg
topbet
topclo

#=========================================================================
=====

#T3
#(a) Simplify network
#Remove nodes with a degree less than 15
StreetGang_net.deg <- delete.vertices(StreetGang_net, degree(StreetGang_net)<15)
plot(StreetGang_net.deg)
#Remove edges with a weight less than 3
StreetGang_net.sp <- delete_edges(StreetGang_neg, E(StreetGang_neg)[weight < 3])
plot(StreetGang_net.sp)
#Plot adjusted network 'layout nicely'
plot(StreetGang_net.sp,layout = layout_nicely)

#=========================================================================
===========

#T4
#Set colour of each node in network based on the ranking (Blue, Red)

V(StreetGang_net)$Ranking

colrs <-c("white","gray","green","blue","gold") #colours we add to frame
V(StreetGang_net)$color <-colrs[V(StreetGang_net)$Ranking]
V(StreetGang_net)$size <-V(StreetGang_net)$Ranking*3#ranking multiply three for the size of
node
```

```
V(StreetGang_net)$label<-NA #no label required
E(StreetGang_net)$width<-E(StreetGang_net)$weight/6#diving the weight of edges by three
E(StreetGang_net)$arrow.size<-.2 #size of arrow
E(StreetGang_net)$edge.color<-"black" #colour of edge
plot(StreetGang_net,layout = layout_nicely) #plot setout format
legend(x="topleft", y=-1.1, c("Rank1","Rank2","Rank3","Rank4","Rank5"),
pch=21,col="777777",pt.bg=colrs, pt.cex=2, cex=.8, bty="n",
    ncol=1) #legend and assignment of values

#===============================================================================
==========

#T4
V(StreetGang_net)$Birthplace #gives us values 1,2,3,4
V(StreetGang_net)$Prison

BPcolrs <-c("yellow","red","blue","orange")
V(StreetGang_net)$color <-BPcolrs[V(StreetGang_net)$Birthplace]
V(StreetGang_net)$size <-V(StreetGang_net)$Birthplace*3
V(StreetGang_net)$label<-NA
E(StreetGang_net)$width<-E(StreetGang_net)$weight/6
E(StreetGang_net)$arrow.size<-.2
E(StreetGang_net)$edge.color<-"black"

plot(StreetGang_net)
legend(x="topleft", y=-1.1, c("West Africa","Carribean","UK","East African"),
pch=21,col="777777",pt.bg=BPcolrs, pt.cex=2, cex=.8, bty="n",
    ncol=1)

prison <- delete.vertices(StreetGang_net, V(StreetGang_net) #deleting vertices of no prison time
[V(StreetGang_net)$Prison == 0])

edge.start <-ends(prison,es=E(prison), names=F)[,1]#creating new edges and assigning prison
df
edge.col<-V(prison)$color[edge.start] #adding colour to the edges to signify connections
plot(prison, edge.color=edge.col, edge.curved=.1,layout = layout_nicely)
legend(x="topleft", y=-1.1, c("West Africa","Carribean","UK","East African"),
pch=21,col="777777",pt.bg=BPcolrs, pt.cex=2, cex=.8, bty="n",
    ncol=1)

#===============================================================================

#T4
#(C)     Using the network created in (b) delete nodes where gang member ranking is less than 3.

prison2 <- delete.vertices(prison, V(prison)
               [V(prison)$Ranking < 3])#adjustment of ranking

edge.start <-ends(prison2,es=E(prison2), names=F)[,1]
edge.col<-V(prison2)$color[edge.start]
plot(prison2, edge.color=edge.col, edge.curved=.1)
```

```
legend(x="topleft", y=-1.1, c("West Africa","Carribean","UK","East African"),
pch=21,col="777777",pt.bg=BPcolrs, pt.cex=2, cex=.8, bty="n",
    ncol=1)

#Now determine the hub score of gang members within this network.
#Using this network create a two-panel plot.
#In the first panel, plot the network where the size of each node is set to 15 times the value of
the hub score.
#In the second panel, display the communities within the network using the cluster_optimal()
function.

hs <-hub.score(prison2,weights=NA)$vector #creating hub score using prison2 network
par(mfrow=c(1,2)) #plot two network format
plot(prison2, vertex.size=hs*15,main="Hubs") #creating visual plot with title
clp <- custer_optimal(StreetGang_net) #cluster creation so we can see community behavior
plot(StreetGang_net,main="Communities") #plotting network

#======================================================================
====

#T5. Simplify the network created in T1 based on the Ranking attribute.
#Create 5 networks in total – one for each ranking. Plot each network.

StreetGang_net <- graph_from_data_frame(d=Glinks, vertices=Gnodes, directed=F)
E(StreetGang_net)$width <- E(StreetGang_net)$weight
V(StreetGang_net)$size <- V(StreetGang_net)$Age/2 #Size of each node equal age attribute
divide by 2

rankings <- unique(V(StreetGang_net)$Ranking) #ranking attribute being created in rankings
subgraphs<-list() #creating subgraph list to create a total of 5 networks
for (ranking in rankings) {vrank <-V(StreetGang_net)[V(StreetGang_net)$Ranking == ranking]
grank<-induced_subgraph(StreetGang_net, vrank) #new data frame for stored sub graphs to
implement networks
subgraphs[[as.character(ranking)]]<- grank} #function to plot rankings ref rstudio

colors1 <- c("white","gray","green","blue","gold")
par(mfrow = c(2, 3))
for (i in 1:length(subgraphs)) {plot(subgraphs[[i]], main = paste("Ranking", names(subgraphs)[i]),
vertex.color = colors1[i])
} #arrays of stored data in rankings distributing a total of 5 as per ranking attributes and counting
system

#======================================================================
=====

#T6 (a)

#UK and West Africa

StreetGang_net <- graph_from_data_frame(d=Glinks, vertices=Gnodes, directed=F)
E(StreetGang_net)$width <- E(StreetGang_net)$weight!=1
```

```
V(StreetGang_net)$size <- V(StreetGang_net)$Age/2
StreetGang_net_df <- get.data.frame(StreetGang_net, what = "edges")
V(StreetGang_net)$Birthplace

BPcol <-c("West Africa" = "yellow","Carribean" = "red","UK" ="blue","East Africa" = "orange")
V(StreetGang_net)$color <-BPcol[V(StreetGang_net)$Birthplace]
V(StreetGang_net)$label<-NA
E(StreetGang_net)$width<-E(StreetGang_net)$weight!=1
E(StreetGang_net)$arrow.size<-.2

except24 <- delete.vertices(StreetGang_net, V(StreetGang_net) #creating new network removing unneeded countries
              [V(StreetGang_net)$Birthplace == 2]) #removing birthplace 2

except25 <- delete.vertices(except24, V(except24) #reintegrating formula to simplify networking accordingly
               [V(except24)$Birthplace == 4]) #removing birthplace 4

plot(except25, main ="UK and West Africa")
legend(x="topleft", y=-1.1, c("West Africa","Carribean","UK","East African"),
pch=21,col="777777",pt.bg=BPcol, pt.cex=2, cex=.8, bty="n",
    ncol=1) #plotting only the data we need to measure rankings with UK and other countries
#================================================================================

#UK and Carribean

BPcol <-c("West Africa" = "yellow","Carribean" = "red","UK" ="blue","East Africa" = "orange")
V(StreetGang_net)$color <-BPcol[V(StreetGang_net)$Birthplace]
V(StreetGang_net)$label<-NA
E(StreetGang_net)$width<-E(StreetGang_net)$weight!=1
E(StreetGang_net)$arrow.size<-.2


except14 <- delete.vertices(StreetGang_net, V(StreetGang_net)
               [V(StreetGang_net)$Birthplace == 1])

except15 <- delete.vertices(except14, V(except14)
               [V(except14)$Birthplace == 4])

plot(except15, main = "UK and Carribean")
legend(x="topleft", y=-1.1, c("West Africa","Carribean","UK","East African"),
pch=21,col="777777",pt.bg=BPcol, pt.cex=2, cex=.8, bty="n",
    ncol=1)

#===================================================================================

#UK and East Africa

BPcol <-c("West Africa" = "yellow","Carribean" = "red","UK" ="blue","East Africa" = "orange")
```

```
V(StreetGang_net)$color <-BPcol[V(StreetGang_net)$Birthplace]
V(StreetGang_net)$label<-NA
E(StreetGang_net)$width<-E(StreetGang_net)$weight!=1
E(StreetGang_net)$arrow.size<-.2


except12 <- delete.vertices(StreetGang_net, V(StreetGang_net)
                [V(StreetGang_net)$Birthplace == 1])

except13 <- delete.vertices(except12, V(except12)
                [V(except12)$Birthplace == 2])

plot(except13, main = "UK and East Africa")
legend(x="topleft", y=-1.1, c("West Africa","Carribean","UK","East African"),
pch=21,col="777777",pt.bg=BPcol, pt.cex=2, cex=.8, bty="n",
    ncol=1)

#==========================================================================
==================

#(b) using UK and carribean interactions, calculate authority scores of gang members.
#set size of node to 10 x value of authority score
#create two panel plotting window and plot network in first window
#identify communities in this network using cluster_optimal() function and plot these
communities in second panel

as <- authority_score(except15,weights=NA)$vector#creating authority score
clp<-cluster_optimal(except15)#creating cluster to see and identify communities in a network
par(mfrow=c(1,2)) #two networks to plot
plot(except15, vertex.size=as*10, main = "Authorities") #authorities plot
plot(clp,except15,main="Communities") #communities plot
```