

# Homework 10 - Alligators

Ammar Plumber, Elaina Lin, Kim Nguyen, Meghan Aines, and Ryan Karbowicz

4/27/2021

The options for each name are the following:

1. I have texted in the last 7 days
2. I have met during my time in MBDS
3. Is my friend
4. I ask for advice/help
5. Asks me for help/advice

Options for the introversion question:

1. Introverted
2. Middle ground
3. Extraverted

```
# Load packages and an xls file
library(readxl)
library(network)
library(igraph)
library(tidyverse)
hw10<- read_excel("hw10.xlsx")
node<- read_excel("hw10.xlsx", skip = 1)
colnames(node) <- c("ID", 1:40, "trait")
```

##1. Prepare the data for the network analysis (create a separate edge and node list)

```
#create the edge list
edge_prep <- node %>% pivot_longer(col = -c(ID, trait), names_to = "in-tie", values_to = "value") %>%
  rename (`out-tie` = ID)

edgelist <- separate_rows(edge_prep, value, sep = ",") %>%
  rename (`tie-type` = value)
as.character(edgelist$`tie-type`)
```

```
##      [1] NA  NA  NA  NA  NA  NA  NA  NA  NA  "4" NA  "4" "1" NA  NA  NA  NA  NA  NA  "1" "2" "3" "4" "5" NA  "
##      [30] NA  NA  NA  NA  NA  NA  NA  "4" NA  NA  NA  NA  "1" "2" "3" "4" "5" NA  "1" "3" "4" "5" NA  NA  NA
##      [59] NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
```

##	[88]	NA	NA	"2"	"1"	NA	"1"	NA	NA	NA	NA	"1"	"2"	"3"	"4"	"5"	NA	"1"	"3"	"4"	"5"	NA	NA	"1"	"2"
##	[117]	NA	NA	NA	NA	NA	NA	NA	"1"	"3"	"4"	"5"	NA	"1"	"2"	"3"	"4"	"5"	NA	NA	NA	NA	NA	NA	NA
##	[146]	NA	NA	NA	"1"	"2"	"3"	"4"	"5"	NA	"1"	"2"	"3"	"4"	NA	NA	NA	"1"	"2"	"3"	"4"	"5"	NA	NA	NA
##	[175]	NA	NA	NA	NA	NA	"1"	"2"	"3"	"4"	"1"	"2"	"3"	"4"	NA	NA	NA	NA	NA	NA	"2"	"2"	"3"	"1"	"2"
##	[204]	"1"	"2"	"3"	"4"	"5"	NA	NA	"1"	"2"	"3"	"1"	"2"	"3"	"4"	"5"	NA	NA	NA	NA	NA	NA	NA	NA	NA
##	[233]	NA	NA	"3"	NA	NA	"3"	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	"3"	NA	NA	NA	"3"	NA
##	[262]	NA	NA	"1"	"2"	"1"	NA	NA	"2"	NA	"4"	NA	NA	NA	NA	"1"	NA	"1"	"1"	NA	"1"	"2"	"1"	NA	"2"
##	[291]	NA	"2"	"1"	"3"	"4"	NA	"2"	NA	NA	NA	"2"	NA	NA	NA	NA	NA	"1"	"2"	"3"	"5"	"2"	"3"	"4"	NA
##	[320]	"5"	"1"	"2"	"3"	"4"	"5"	"4"	"4"	NA	"4"	NA	"1"	"3"	"4"	NA	NA	"1"	"2"	"3"	"4"	"5"	"3"	NA	"2"
##	[349]	"2"	"3"	"4"	"5"	"3"	"4"	"1"	"3"	"4"	"5"	"2"	"3"	NA	NA	NA	"1"	"2"	"3"	"4"	"1"	"2"	"3"	"4"	"1"
##	[378]	"2"	"3"	"4"	"5"	"1"	"3"	"4"	"5"	"1"	"3"	"4"	"5"	"2"	"2"	"3"	"4"	NA	NA	NA	NA	NA	NA	NA	NA
##	[407]	"4"	NA	"1"	NA	NA	"1"	"3"	NA	NA	NA	NA	NA	"4"	NA	NA	NA	NA	NA	NA	"1"	"4"	"5"	"4"	NA
##	[436]	NA	"5"	"4"	NA	NA	NA	"5"	NA	NA	NA	NA	NA	NA	"4"	NA	NA	NA	"1"	"3"	"4"	"5"	NA	NA	NA
##	[465]	"3"	"4"	"5"	NA	NA	"4"	"1"	"5"	"1"	"3"	"4"	"5"	NA	NA	NA	NA	"1"	"2"	"3"	"4"	"5"	"3"	"4"	NA
##	[494]	"5"	"1"	"3"	"4"	"5"	"4"	"5"	NA	NA	NA	NA	NA	NA	NA	NA	NA	"1"	"2"	"3"	"4"	"5"	"1"	"2"	NA
##	[523]	"3"	"4"	NA	NA	NA	NA	NA	NA	NA	NA	NA	"1"	"2"	"3"	"4"	"5"	"1"	"2"	"3"	NA	"2"	"5"	NA	NA
##	[552]	"3"	"4"	"5"	"1"	"2"	"3"	"4"	"5"	"1"	"2"	"3"	"4"	"5"	NA	"1"	"2"	"3"	"4"	"5"	NA	NA	"2"	"3"	NA
##	[581]	NA	NA	NA	"1"	"2"	"3"	"2"	NA	NA	NA	"2"	NA	"2"	"4"	"2"	"2"	"4"	NA	NA	"1"	"2"	"4"	NA	"2"
##	[610]	"2"	"4"	"2"	"3"	"4"	"2"	NA	NA	"1"	"2"	"4"	NA	NA	"1"	"2"	"3"	"4"	"1"	"2"	"3"	"4"	NA	"2"	"2"
##	[639]	"2"	NA	NA	NA	NA	"1"	"2"	"3"	"4"	NA	NA	"1"	"2"	"3"	"4"	"5"	"1"	"2"	"3"	"4"	"5"	NA	"4"	"1"
##	[668]	NA	NA	NA	NA	NA	"2"	"2"	NA	"4"	"4"	"5"	"2"	NA	NA	NA	"2"	"2"	"4"	NA	"1"	"2"	"3"	"4"	"1"
##	[697]	NA	"1"	"2"	"3"	"4"	"5"	NA	NA	"4"	"5"	NA	"1"	NA	NA	"5"	NA	"1"	"4"	"1"	"4"	"5"	"4"	"1"	"4"
##	[726]	NA	NA	NA	NA	"1"	"3"	"4"	"5"	"4"	"5"	NA	"1"	"2"	"3"	"4"	"1"	"3"	"4"	"5"	NA	NA	NA	NA	"1"
##	[755]	"3"	"4"	"1"	"3"	"4"	"5"	NA	NA	NA	"1"	"3"	"4"	"5"	"1"	"1"	"3"	"4"	"5"	NA	NA	NA	NA	NA	"1"
##	[784]	NA	"1"	"2"	"3"	"4"	"5"	"2"	"3"	NA	"4"	NA	NA	"1"	"1"	"3"	"4"	"5"	NA	NA	"3"	"4"	NA	NA	"1"
##	[813]	"5"	NA	"1"	"3"	NA																			

##2. Calculate measures of centrality and similarity in networks depending on the type of tie? Which tie type generates the most similar network in terms of introversion? Which tie type is the most dissimilar?

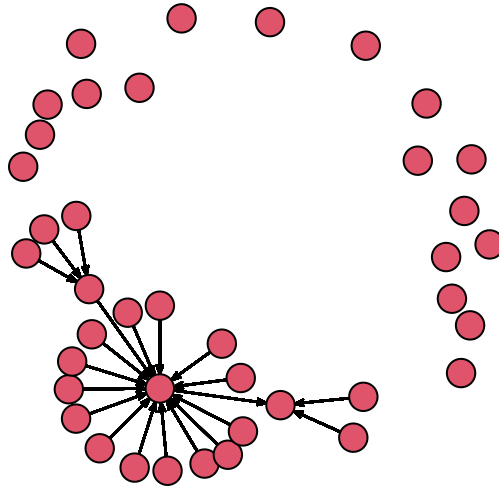
Tie type 1 (I have texted in the last 7 days) generates the most similar network in terms of introversion, while tie type 3 generates the most dissimilar.

```
classnormsnet <- network(edgelist, vertex.attr = nodelist, matrix.type = "edgelist", ignore.eval = FALSE)
summary(classnormsnet)
```

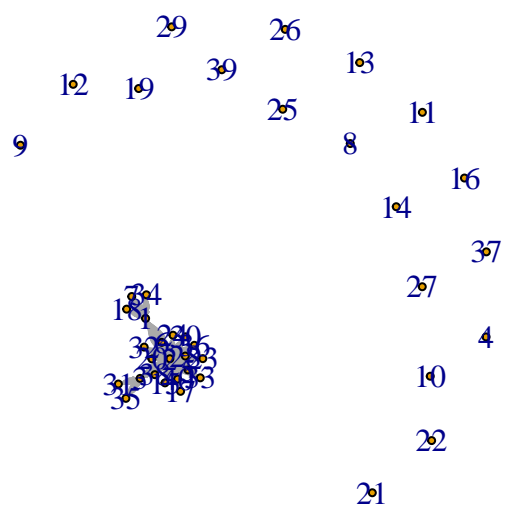
```
## Network attributes:
##   vertices = 40
##   directed = TRUE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges = 1246
##   missing edges = 0
##   non-missing edges = 1246
##   density = 0.7987179
##
## Vertex attributes:
##
##   ID:
##     numeric valued attribute
##     attribute summary:
##       Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1.00   10.75   20.50   20.50   30.25   40.00
##
##   trait:
##     numeric valued attribute
##     attribute summary:
##       Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       1         2         2         2         2         3        19
##   vertex.names:
##     character valued attribute
##     40 valid vertex names
##
## Edge attributes:
##
##   in-tie:
##     character valued attribute
##     attribute summary:
##       the 10 most common values are:
## 18  5  6 28  2 31 29 22 19 34
## 53 50 48 47 44 43 42 41 38 38
##
##   tie-type:
##     character valued attribute
##     attribute summary:
##       0  1  2  3  4  5
## 551 143 133 131 176 112
##
## Network adjacency matrix:
##   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 33 34 35 36
## 1  0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 2  0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```



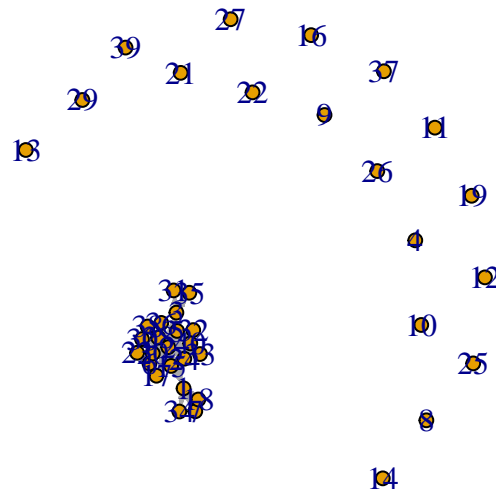
```
plot(classnormsnet, vertex.cex = 3)
```



```
# graphing the network  
network_class <- graph_from_data_frame(d = edgelist, vertices = nodelist, directed = TRUE)  
plot(network_class, edge.arrow.size = 0.1, vertex.size = 3)
```



```
plot(network_class, edge.arrow.size = 0.1, vertex.size = 6)
```



### Edgelist based on Tie-Type, Nodelist based on trait

```
# edgelist based on tie-type, nodelist based on trait
edgelist1 <- edgelist %>% filter(`tie-type` == 1)
edgelist2 <- edgelist %>% filter(`tie-type` == 2)
edgelist3 <- edgelist %>% filter(`tie-type` == 3)
edgelist4 <- edgelist %>% filter(`tie-type` == 4)
edgelist5 <- edgelist %>% filter(`tie-type` == 5)
nodelist_introvert <- nodelist %>% filter(trait==1)
nodelist_extrovert <- nodelist %>% filter(trait == 3)
network_class_tietype1 <- graph_from_data_frame(d = edgelist1, vertices = nodelist, directed = TRUE)
network_class_tietype2 <- graph_from_data_frame(d = edgelist2, vertices = nodelist, directed = TRUE)
network_class_tietype3 <- graph_from_data_frame(d = edgelist3, vertices = nodelist, directed = TRUE)
network_class_tietype4 <- graph_from_data_frame(d = edgelist4, vertices = nodelist, directed = TRUE)
network_class_tietype5 <- graph_from_data_frame(d = edgelist5, vertices = nodelist, directed = TRUE)
```

**Density** Density is the number of ties relative to the number of possible ties

From greatest to least, the rank of tie types in terms of density is as follows:

1. I ask for advice/help (tie type 4)
2. I have texted in the last 7 days (tie type 1)
3. I have met during my time in MBDS (tie type 2)
4. Is my friend (tie type 3)

5. Asks me for help/advice (tie type 5)

```
# density based on tie type  
edge_density(network_class_tietype1, loops = F)
```

```
## [1] 0.09166667
```

```
edge_density(network_class_tietype2, loops = F)
```

```
## [1] 0.08525641
```

```
edge_density(network_class_tietype3, loops = F)
```

```
## [1] 0.08397436
```

```
edge_density(network_class_tietype4, loops = F)
```

```
## [1] 0.1128205
```

```
edge_density(network_class_tietype5, loops = F)
```

```
## [1] 0.07179487
```

## Reciprocity

From greatest to least, the rank of tie types in terms of reciprocity is as follows:

1. Is my friend (tie type 3)
2. I have met during my time in MBDS (tie type 2)
3. I have texted in the last 7 days (tie type 1)
4. Asks me for help/advice (tie type 5)
5. I ask for advice/help (tie type 4)

```
#reciprocity  
reciprocity(network_class_tietype1)
```

```
## [1] 0
```

```
reciprocity(network_class_tietype2)
```

```
## [1] 0
```

```
reciprocity(network_class_tietype3)
```

```
## [1] 0
```



```
reciprocity(network_class_tietype4)
```

```
## [1] 0
```

```
reciprocity(network_class_tietype5)
```

```
## [1] 0
```

### Transitivity

From greatest to least, the rank of tie types in terms of transitivity is as follows:

1. I have met during my time in MBDS (tie type 2)
2. Is my friend (tie type 3)
3. I ask for advice/help (tie type 4)
4. I have texted in the last 7 days (tie type 1)
5. Asks me for help/advice (tie type 5)

```
# transitivity  
transitivity(network_class_tietype1, type="global")
```

```
## [1] 0
```

```
transitivity(network_class_tietype2, type="global")
```

```
## [1] 0
```

```
transitivity(network_class_tietype3, type="global")
```

```
## [1] 0
```

```
transitivity(network_class_tietype4, type="global")
```

```
## [1] 0
```

```
transitivity(network_class_tietype5, type="global")
```

```
## [1] 0
```

### Diameter

The following tie types have a diameter of 5:

1. I have texted in the last 7 days (tie type 1)
2. I have met during my time in MBDS (tie type 2)

3. Is my friend (tie type 3)

The following tie types have a diameter of 4:

4. I ask for advice/help (tie type 4)
5. Asks me for help/advice (tie type 5)

```
# Diameter
diameter(network_class_tietype1, directed=F, weights=NA)
```

```
## [1] 4
```

```
diameter(network_class_tietype2, directed=F, weights=NA)
```

```
## [1] 4
```

```
diameter(network_class_tietype3, directed=F, weights=NA)
```

```
## [1] 4
```

```
diameter(network_class_tietype4, directed=F, weights=NA)
```

```
## [1] 4
```

```
diameter(network_class_tietype5, directed=F, weights=NA)
```

```
## [1] 3
```

### Closeness

Closeness measures how close people are to everyone else

From closest to least close, the rank of tie types is as follows:

1. I have met during my time in MBDS (tie type 2)
2. I ask for advice/help (tie type 4)
3. I have texted in the last 7 days (tie type 1)
4. Asks me for help/advice (tie type 5)
5. Is my friend (tie type 3)

```
#calculating closeness
mean(closeness(network_class_tietype1, mode="all", weights=NA))
```

```
## Warning in closeness(network_class_tietype1, mode = "all", weights = NA): At centrality.c:2784 :clos
## well-defined for disconnected graphs
```

```
## [1] 0.0009559908
```

```
mean(closeness(network_class_tietype2, mode="all", weights=NA))
```

```
## Warning in closeness(network_class_tietype2, mode = "all", weights = NA): At centrality.c:2784 :clos  
## well-defined for disconnected graphs
```

```
## [1] 0.0008753162
```

```
mean(closeness(network_class_tietype3, mode="all", weights=NA))
```

```
## Warning in closeness(network_class_tietype3, mode = "all", weights = NA): At centrality.c:2784 :clos  
## well-defined for disconnected graphs
```

```
## [1] 0.0009559908
```

```
mean(closeness(network_class_tietype4, mode="all", weights=NA))
```

```
## Warning in closeness(network_class_tietype4, mode = "all", weights = NA): At centrality.c:2784 :clos  
## well-defined for disconnected graphs
```

```
## [1] 0.0009131176
```

```
mean(closeness(network_class_tietype5, mode="all", weights=NA))
```

```
## Warning in closeness(network_class_tietype5, mode = "all", weights = NA): At centrality.c:2784 :clos  
## well-defined for disconnected graphs
```

```
## [1] 0.0007701825
```

## Betweenness

Betweenness is the shortest paths between nodes that go through a given node

From most betweenness to least betweenness, the rank of ties in terms of betweenness is as follows:

1. Is my friend (tie type 3)
2. I ask for advice/help (tie type 4)
3. Asks me for help/advice (tie type 5)
4. I have texted in the last 7 days (tie type 1)
5. I have met during my time in MBDS (tie type 2)

```
mean(betweenness(network_class_tietype1, directed=F, weights=NA))
```

```
## [1] 6.8
```

```
mean(betweenness(network_class_tietype2, directed=F, weights=NA))
```

```
## [1] 5.625
```

```
mean(betweenness(network_class_tietype3, directed=F, weights=NA))
```

```
## [1] 6.8
```

```
mean(betweenness(network_class_tietype4, directed=F, weights=NA))
```

```
## [1] 6.2
```

```
mean(betweenness(network_class_tietype5, directed=F, weights=NA))
```

```
## [1] 2.9
```

## Degree

Degree looks at the number of ties in a network

From greatest to least, the rank of tie types in terms of degree is as follows:

1. I ask for advice/help (tie type 4)
2. I have texted in the last 7 days (tie type 1)
3. I have met during my time in MBDS (tie type 2)
4. Is my friend (tie type 3)
5. Asks me for help/advice (tie type 5)

```
mean(degree(network_class_tietype1, mode="all"))
```

```
## [1] 7.15
```

```
mean(degree(network_class_tietype2, mode="all"))
```

```
## [1] 6.65
```

```
mean(degree(network_class_tietype3, mode="all"))
```

```
## [1] 6.55
```

```
mean(degree(network_class_tietype4, mode="all"))
```

```
## [1] 8.8
```

```
mean(degree(network_class_tietype5, mode="all"))
```

```
## [1] 5.6
```

## Centrality Degree

Information Centrality is the number of all paths between nodes that go through a given node

From greatest centralization to least centralization, the rank of tie types is as follows:

1. I ask for advice/help (tie type 4)
2. I have met during my time in MBDS (tie type 2)
3. Is my friend (tie type 3)
4. I have texted in the last 7 days (tie type 1)
5. Asks me for help/advice (tie type 5)

```
centr_degree(network_class_tietype1, mode="in", normalized=T)
```

```
## $res
## [1] 24 96 23 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [40] 0
##
## $centralization
## [1] 2.369872
##
## $theoretical_max
## [1] 1560
```

```
centr_degree(network_class_tietype2, mode="in", normalized=T)
```

```
## $res
## [1] 24 77 32 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [40] 0
##
## $centralization
## [1] 1.889103
##
## $theoretical_max
## [1] 1560
```

```
centr_degree(network_class_tietype3, mode="in", normalized=T)
```

```
## $res
## [1] 20 84 27 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [40] 0
##
## $centralization
## [1] 2.069872
##
## $theoretical_max
## [1] 1560
```

```
centr_degree(network_class_tietype4, mode="in", normalized=T)
```

```
## $res
## [1] 27 118 31 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [30] 0 0 0 0 0 0 0 0 0 0 0 0
##
## $centralization
## [1] 2.912821
##
## $theoretical_max
## [1] 1560
```

```
centr_degree(network_class_tietype5, mode="in", normalized=T)
```

```
## $res
## [1] 21 70 21 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [40] 0
##
## $centralization
## [1] 1.723077
##
## $theoretical_max
## [1] 1560
```

## Mean Distance

The mean distance between nodes, in order from least to greatest, is as follows:

1. Is my friend (tie type 3)
2. I ask for advice/help (tie type 4)
3. I have met during my time in MBDS (tie type 2)
4. I have texted in the last 7 days (tie type 1)
5. Asks me for help/advice (tie type 5)

```
mean_distance(network_class_tietype1, directed=T)
```

```
## [1] 1.575
```

```
mean_distance(network_class_tietype2, directed=T)
```

```
## [1] 1.583333
```

```
mean_distance(network_class_tietype3, directed=T)
```

```
## [1] 1.575
```

```
mean_distance(network_class_tietype4, directed=T)
```

```
## [1] 1.578947
```

```
mean_distance(network_class_tietype5, directed=T)
```

```
## [1] 1.392857
```

**Which tie type generates the most similar network in terms of introversion?  
Which tie type is the most dissimilar?**

In terms of introversion:

**Most similar network:** “I have texted in the last 7 days”

**Most dissimilar network:** “Is my friend”

```
node <- nodelist
node[is.na(node)] <- -1
net <- graph_from_data_frame(d = edgelist, vertices = node, directed = TRUE)
net1 <- graph_from_data_frame(d = edgelist1, vertices = node, directed = TRUE)
net2 <- graph_from_data_frame(d = edgelist2, vertices = node, directed = TRUE)
net3 <- graph_from_data_frame(d = edgelist3, vertices = node, directed = TRUE)
net4 <- graph_from_data_frame(d = edgelist4, vertices = node, directed = TRUE)
net5 <- graph_from_data_frame(d = edgelist5, vertices = node, directed = TRUE)
#filter edgelist for all 5 tie types...then calculate this measure for 5 networks and see which is most
#Similarity
assortativity(net1, V(net1)$trait, directed=T)
```

```
## [1] -0.5815749
```

```
assortativity(net2, V(net2)$trait, directed=T)
```

```
## [1] -0.6688433
```

```
assortativity(net3, V(net3)$trait, directed=T)
```

```
## [1] -0.6573574
```

```
assortativity(net4, V(net4)$trait, directed=T)
```

```
## [1] -0.6198843
```

```
assortativity(net5, V(net5)$trait, directed=T)
```

```
## [1] -0.6045513
```

##3 Do introverts tend to be at the periphery while extraverts are in the center of the network? Motivate.

We use both decision tree and regression to examine if introverts tend to be at the periphery while extraverts are in the center of the network. The input is the trait (level of introvert/extravert) and the output is the degree/betweenness.

Both the decision tree models are unable to predict any patterns. The regression model also does not show any significances. The small sample size is our limitation. We come to conclude that trait (introvert/extravert) does not predict where introverts and extraverts are in the network.

```
library(rpart)
library(rpart.plot)
library(caret)

#decision tree: betweenness and trait
tree_bt <- nodelist %>% mutate (between = betweenness(network_class, directed=F, weights=NA))
model_tree_bt <- rpart(between ~ as.factor(trait), method = "anova", data = tree_bt, cp = 0.00001)
rpart.plot(model_tree_bt)
```

12  
100%

```
#decision tree: degree and trait
tree_dg <- nodelist %>% mutate (degree = degree(network_class, mode="in"))
model_tree_dg <- rpart(degree ~ as.factor(trait), method = "anova", data = tree_dg, cp = 0.00001)
rpart.plot(model_tree_dg)
```



50  
100%

```
#regression: bewteenness and trait
summary(lm(between ~ as.factor(trait), data = tree_bt))

##
## Call:
## lm(formula = between ~ as.factor(trait), data = tree_bt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -67.0    -3.8    -3.8    -3.8   134.0
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.868e-14  2.356e+01   0.000  1.0000
## as.factor(trait)2  3.800e+00  2.581e+01   0.147  0.8846
## as.factor(trait)3  6.700e+01  3.331e+01   2.011  0.0595 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.8 on 18 degrees of freedom
## (19 observations deleted due to missingness)
## Multiple R-squared:  0.2597, Adjusted R-squared:  0.1775
## F-statistic: 3.158 on 2 and 18 DF,  p-value: 0.06676
```