Assignment Nº2

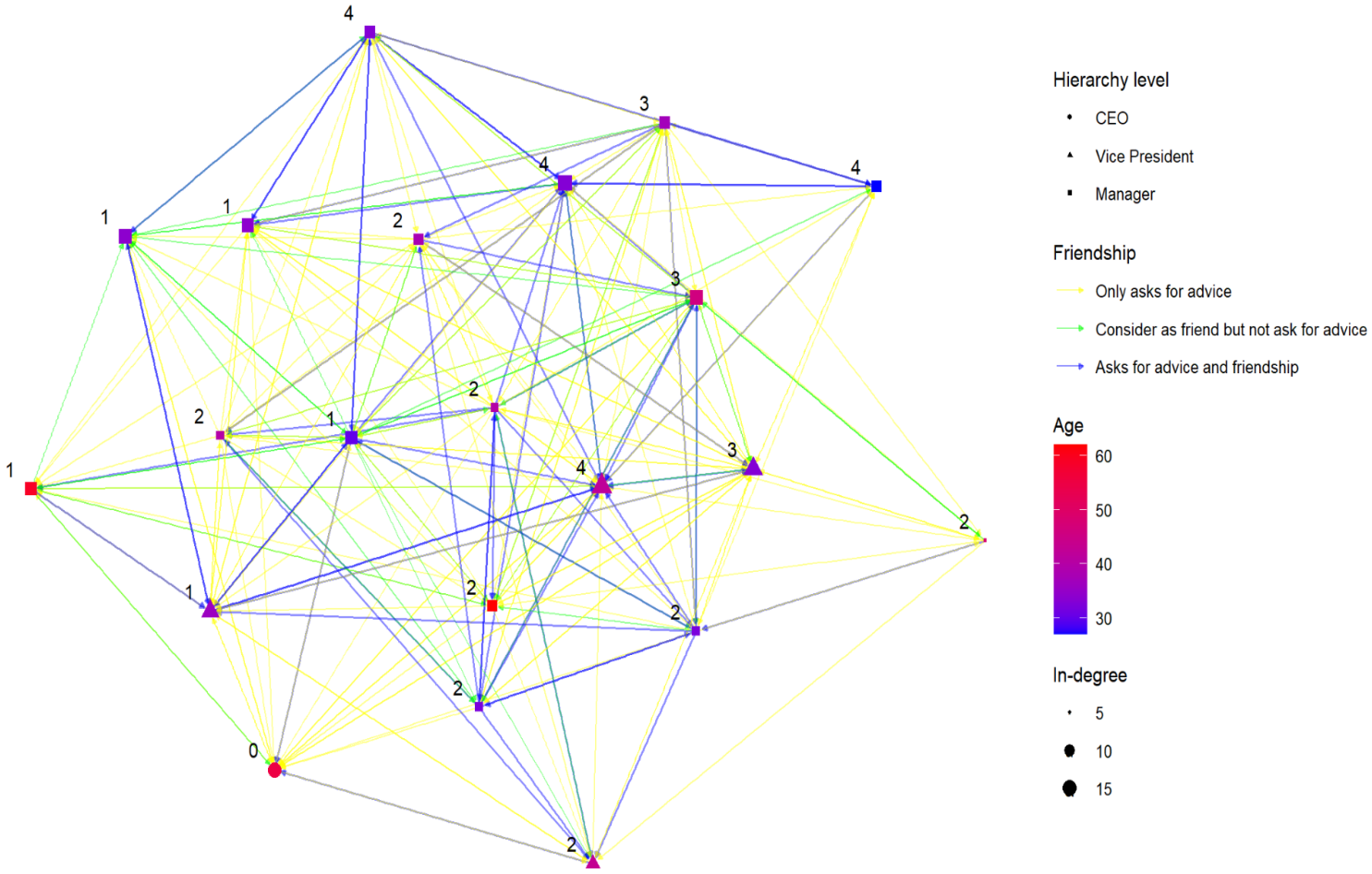
851-0252-06L Introduction to Social Networks: Theory, Methods and Applications FS2023

Daniel Repérant
Julia Salustowicz
Maria Eugenia Gil Pallares
Zhengxu Li

Task 1: Network Hypotheses (10 points)

(a) Describe and visualize the data set using the techniques you have come across in the course. In how far is this different from other networks that you have studied? What might explain the differences? What was your motivation to choose this network? (max. 300 words)

High-tech Managers (Advice)



We opted to use the **ht_advice** dataset, which was obtained from a high-tech manufacturing company with 100 employees and 21 managers. Each manager was asked to identify who they ask for advice, and information such as age, tenure, corporate hierarchy level, and the department was collected.

We chose this dataset because of its moderate size and structure, which is neither a complete graph with numerous edges nor a sparse graph with disconnected components (a directed graph with 21 nodes and 190 edges, resulting in a density of 0.45). Additionally, this dataset can be used in conjunction with *ht_friends*, where the same managers reported friendship relationships with their colleagues. After combining these two datasets, we represent the different friendship relationships with the size and color of the edges (yellow: advice relationship, green: friendship, blue: advice+friendship). The shape of the nodes will indicate the hierarchy level, and each node will be labeled with the corresponding department. Finally, with the color of the nodes and their size, we describe the age and in-degree.

A first glimpse at the graph shows that, even though we would assume that managers would not have a high in-degree (frequency of being asked for advice), there is no obvious difference between managers and vice presidents.

We would like to emphasize that this dataset is not drastically different from those we have previously encountered. The graph is connected, with no isolated vertices, and is relatively dense (compared to other graphs we have seen) with a reciprocity ratio of 0.47.

The reason for the density of this graph could be attributed to the company's medium size and frequent interactions between employees, who frequently exchange opinions and communicate with both superiors and subordinates. This cooperation results in a high number of interactions.

Code:

```
#  We get the matrix of the graphs from the High-Tech company: advice and friendship
library(mully)
library(sna)
library(igraph)
library(ggraph)
library(networkdata)
net1 <-as.matrix(ht_advice, matrix.type = c("adjacency"))
net2 <-as.matrix(ht_friends, matrix.type = c("adjacency"))

# a new matrix of the right size, prefilled with ties in net1
net3 <- net1
```

```
# where ties exist in the second network, we give a new value
net3[net2 == 1] <- 2
# where ties exist in both networks, we replace with a new, unique value
net3[net1 == 1 & net2 == 1] <- 3
# So we have that the least weight will correspond to the interactions where only
# advice is being asked, but there is no friendship. In the case of weight 2,
# the person asking for advice considers the other person as a friend, but does not ask for
advice from him.

g=graph_from_adjacency_matrix(net3,mode="directed",weighted=TRUE)

# We set the attributes from the original graphs
g <- set_vertex_attr(g, "age", value = vertex_attr(ht_advice, "age"))
g <- set_vertex_attr(g, "level", value = vertex_attr(ht_advice, "level"))
g <- set_vertex_attr(g, "dept", value = vertex_attr(ht_advice, "dept"))


set.seed(1)
layout.graph <- create_layout(g, layout = 'stress')
# we color the edges and make the width proportional to the weight in order to make better
edges <- geom_edge_link2(alpha = .45,
                         arrow = arrow(length = unit(1.5, 'mm'),
                                       type = "closed"),
                         end_cap = circle(2, 'mm'),aes(width = weight,color=factor(weight)))

#nodes are colored according to gender and size proportional to the degree
nodes <- geom_node_point(aes(shape=factor(level),size = degree(g,mode="in"),color = age))
plotlabs <- labs(shape = "Hierarchy level ",size = "In-degree",color ="Age")
# we draw the plot
degr.plot <-
  ggraph(layout.graph)+
  labs(edge_color = "Friendship") +
  edges+
  nodes+
  geom_node_text(aes(label = dept,size =
                       15),nudge_x=-0.05, nudge_y=0.05)+
  plotlabs+
  scale_edge_color_manual(values =c("yellow",'green',"blue"),label = c("Only asks for
advice","Consider as friend but not ask for advice","Asks for advice and friendship")) +
  scale_color_gradient( low = "blue",high = "red")+
  scale_edge_width(guide="none",range = c(0.5, 0.8))+
  theme_graph()+theme(legend.key.size = unit(1, 'cm'),legend.title = element_text(size=15),
legend.text = element_text(size=13))+
  scale_shape_discrete(label = c("CEO","Vice President","Manager"))+
```

```
  ggtitle("High-tech Managers (Advice)")
degr.plot
```

(b) Choose two network theories you have gotten to know (e.g., strength of weak ties, structural holes, structural balance theory, etc.) and come up with two hypotheses that could potentially be tested using the data (or, if applicable, if any additional data need to be collected for testing the hypotheses and how you would do it). Provide theoretical arguments for why there might be evidence for your hypotheses. Note that you don't need to provide the analysis for hypothesis testing. (max. 400 words)

Hypothesis 1: Individuals with a high number of weak ties are more likely to occupy central positions in the network.

The strength of weak ties theory proposes that weak ties (i.e., infrequent or superficial connections between individuals) are important for information flow and innovation. Specifically, individuals with a high number of weak ties tend to have more diverse sources of information and are more likely to be exposed to new ideas and opportunities. In the ht_advice network, we could test whether individuals with a high number of weak ties (those only have yellow or green relationships, i.e. those who are either only being asked for advice, or only being considered as friends, but not simultaneously) are more likely to occupy central positions in the network (e.g., higher betweenness centrality). The theoretical argument for this hypothesis is that weak ties connect different subgroups of the network and thus provide individuals with access to diverse sources of information and opportunities. To test the hypothesis, we can determine whether there is a correlation between the strength of ties and centrality in the network. If individuals with a high number of weak ties are found to occupy more central positions in the network, this would support the hypothesis. However, if no correlation is found, we may need to revise the hypothesis.

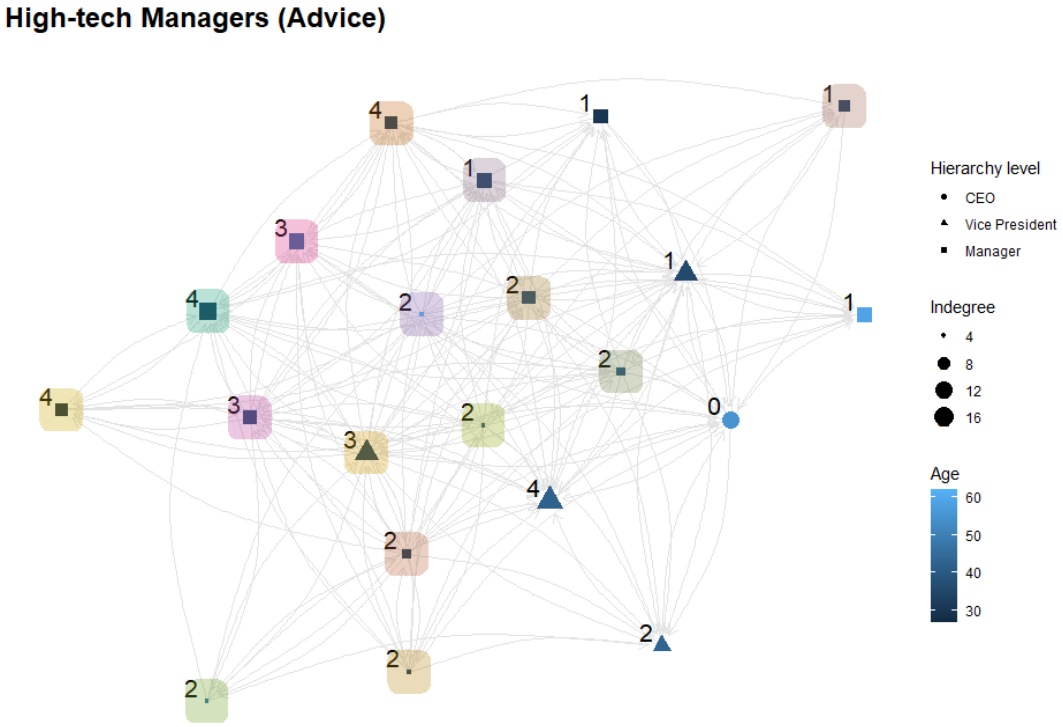
Hypothesis 2: Triadic closure

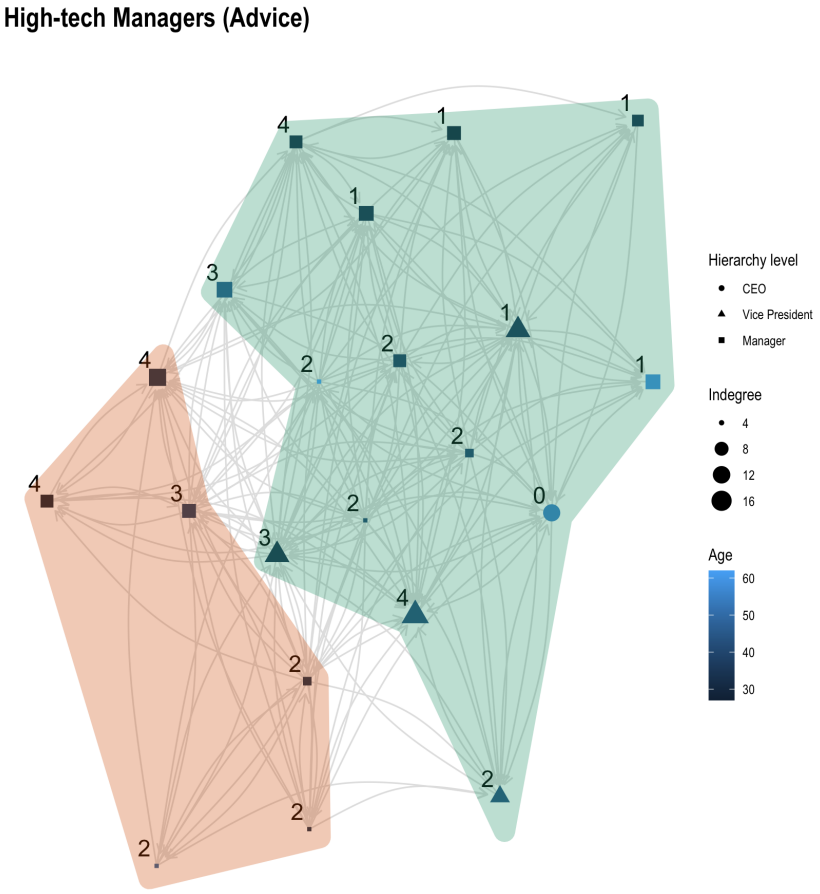
(Strong) Triadic closure makes reference to the increased likelihood of development of a (at least a weak) connection between nodes B and C if there is a (strong) tie between nodes A and B and nodes A and C. In the context of our study, we anticipate observing a higher number of triads than expected by chance due to the transitivity principle and the relatively small size of the company network. For instance, if two individuals, A and B, work in the same department and seek advice from a third person in a different department, Person A's recommendation of this contact to Person B could result in the creation of a new edge on our graph. Due to the frequent interactions between co-workers and the small scale of the organization, diffusion may occur across different community borders. To verify this hypothesis, we may compare the number of triads in a stylized model with the same number of edges to those in our networks. Additionally, we may employ other metrics such as the clustering or transitivity coefficients.

Task 2: Data Analysis (10 points)

(a) Apply one community detection algorithm to the dataset you have chosen, provide justifications for why you choose this algorithm, and interpret your findings. What are patterns you find in this network? What are the possible explanations for those patterns? (max. 200 words)

The application of community detection algorithms, specifically the edge betweenness and walktrap algorithms, to the dataset ht_advice yielded varying results. The edge betweenness algorithm (Figure 1) produced mostly individual unary clusters, which aligns with the notion that coworkers with different knowledge and perspectives are sought for advice, depending on the problem. Also, we see that one non-unary community including 3 vice-presidents, a senior manager and the CEO constitute a group of people that are set apart, maybe the executive group of the company which is frequently asked for advice. This algorithm's output is reasonable for a work environment in our belief. However, the walktrap algorithm (Figure 2) only separated the nodes into two clusters, potentially due to the influence of friendships between coworkers. Although this finding is interesting, it does not make sense in the context of advice-seeking within a company, as it is not necessarily limited to a preferred group of coworkers. Additionally, the small size of the company and high number of ties suggest that the identified communities are not related to specific departments. Therefore, the conclusion is that community detection algorithms may not be useful for this dataset.





Code for walktrap algorithm:

```
library("concaveman")
ht.wt <- cluster_walktrap(ht_advice)

length(ht.wt) #How many communities were identified by the algorithm?
sizes(ht.wt) #What are their sizes?
member.wt <- membership(ht.wt) #Who belongs to which community?
modularity(ht.wt)

set.seed(3)
g <- ggraph(ht_advice, layout = 'fr') +

geom_edge_arc(color="gray89",strength=0.15,arrow=arrow(length=unit(2.5,'mm')),end_cap=circle(3, 'mm')) +
  labs(shape="Hierarchy level",color ="Age", size="Indegree") +
```

```
                                geom_node_point(aes(shape=factor(level)                ,size                =
igraph::degree(ht_advice,mode="in"),color=age)) +
  theme_graph() +
  #scale_color_gradient2( low = "blue",high = "red")+
  geom_node_text(aes(label = dept,size = 15),nudge_x=-0.05, nudge_y=0.05)+
  scale_shape_discrete(label = c("CEO","Vice President","Manager"))+
  ggtitle("High-tech Managers (Advice)")
g +geom_mark_hull(aes(x,y,fill = factor(member.wt)),
                  colour = NA,show.legend = FALSE)+scale_fill_brewer(palette = "Dark2")
```

Code for edge-betweenness algorithm:

```
library("concaveman")
ht.wt <- cluster_edge_betweenness(ht_advice)

length(ht.wt) #How many communities were identified by the algorithm?
sizes(ht.wt) #What are their sizes?
member.wt <- membership(ht.wt) #Who belongs to which community?
modularity(ht.wt)

nb.cols <- 21
mycolors <- colorRampPalette(brewer.pal(8, "Dark2"))(nb.cols)

set.seed(3)
g <- ggraph(ht_advice, layout = 'fr') +

geom_edge_arc(color="gray89",strength=0.15,arrow=arrow(length=unit(2.5,'mm')),end_cap=cir
cle(3, 'mm')) +
  labs(shape="Hierarchy level",color ="Age", size="Indegree") +
                                geom_node_point(aes(shape=factor(level)                ,size                =
igraph::degree(ht_advice,mode="in"),color=age)) +
  theme_graph() +
  #scale_color_gradient2( low = "blue",high = "red")+
  geom_node_text(aes(label = dept,size = 15),nudge_x=-0.05, nudge_y=0.05)+
  scale_shape_discrete(label = c("CEO","Vice President","Manager"))+
  ggtitle("High-tech Managers (Advice)")
g +geom_mark_hull(aes(x,y,fill = factor(member.wt)),
                  colour = NA,show.legend = FALSE)+scale_fill_manual(values = mycolors)
```

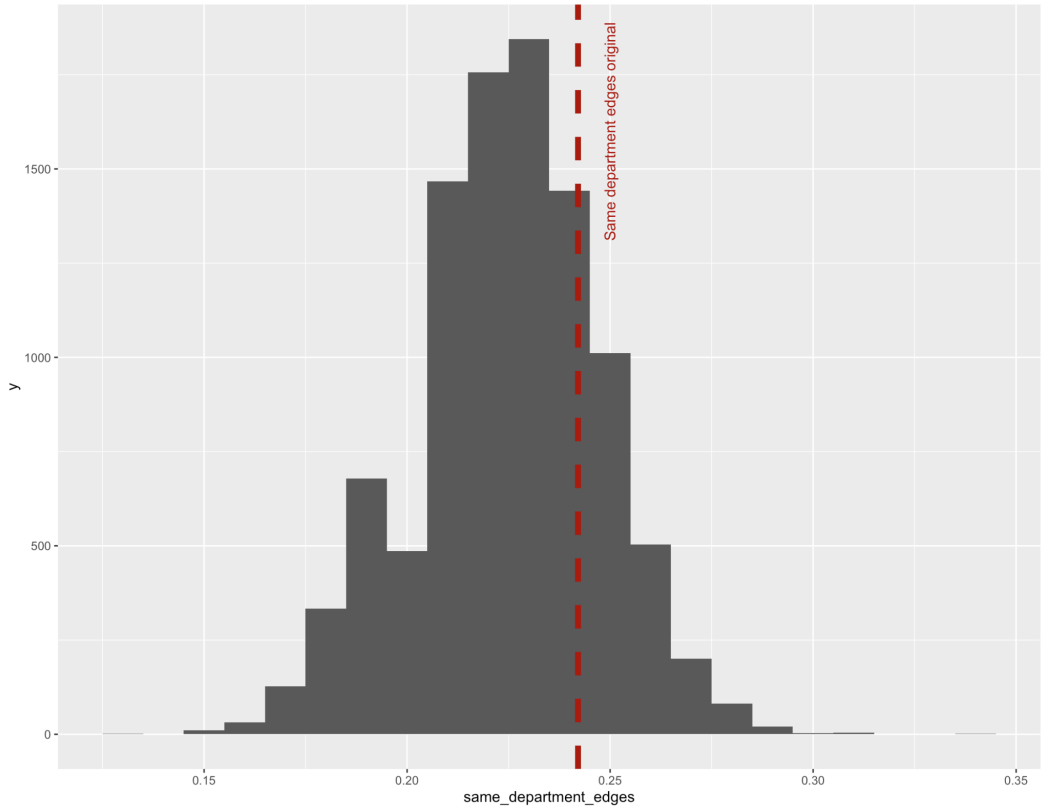
(b) Choose two of the descriptive features discussed in your answers to questions 1a and 2a and conduct two conditional uniform graph tests using an Erdos-Renyi model with the same (expected) density. Interpret the results. (max. 300 words)

In the first exercise, we noticed that there was a high density of edges without an evident pattern, it is hard to visually locate well-separated communities. Therefore, in order to find statistical evidence for any kind of possible homophily in our network, we implement in this section two hypotheses:

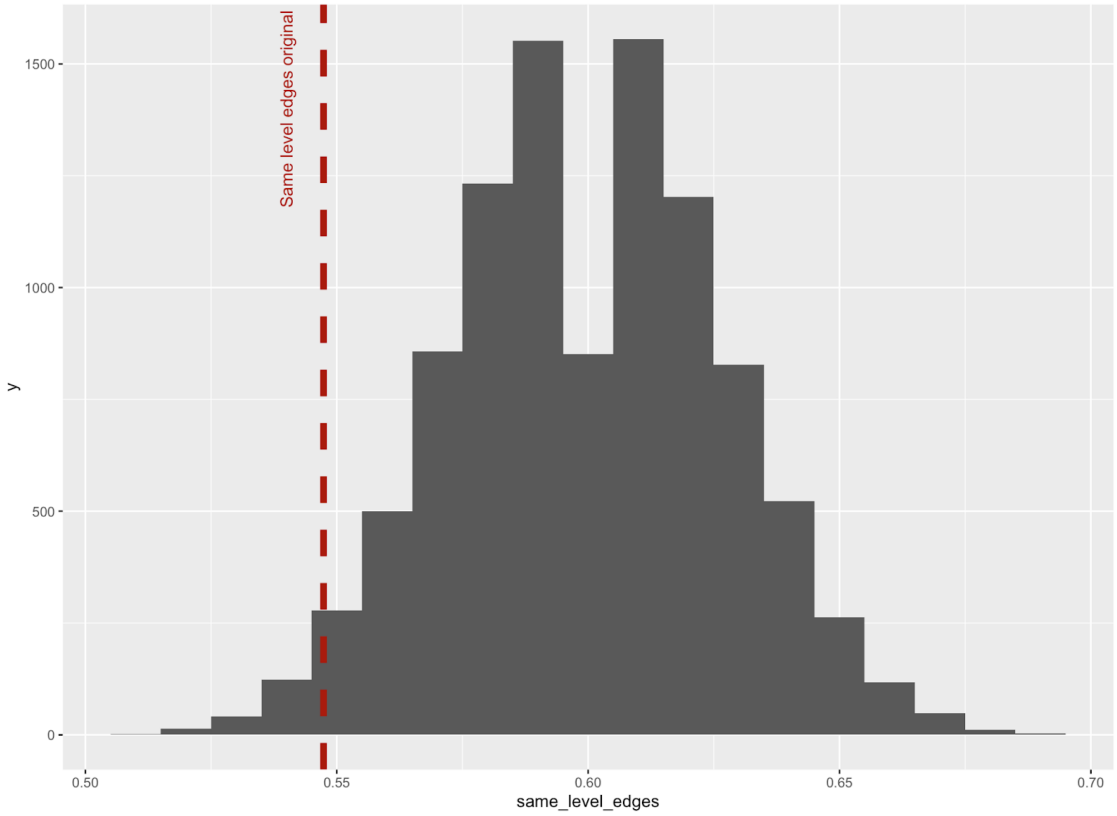
1) Is there a preference to ask for advice from colleagues in the same department?
2) Is there a preference to ask for advice from colleagues of a different hierarchical level?

Test 1)
Initially, our null hypothesis posits that there is no department homophily, meaning that individuals tend to seek advice from anyone regardless of their department. In contrast, our alternative hypothesis assumes that people prefer to consult colleagues from their own department. We preserved the density and attributes of our original graph and employed the Erdos-Renyi model to generate 10,000 graphs with random connections. Next, we compared the proportion of same-department ties in the original graph with the distribution observed from the random simulations.

When considering ties among people from the same department we obtain a p-value of 0.1824, which indicates that we have no significant evidence to reject the null hypothesis. As such, we conclude that the number of same-department edges is comparable to those obtained randomly. This similarity can be attributed to the high density of the graph.



Test 2)
We hypothesize that individuals prefer to seek advice from colleagues of different hierarchical levels (H1), as people tend to consult their superiors for guidance. Our analysis yielded a p-value of 0.029, indicating that we have significant evidence to suggest a preference for consulting people from different ranks in our network. This preference is reflected in the high in-degree of the Vice-presidents, which exceeds what we would expect from a random simulation.



Code for Test 1:

```
rm(list = ls()) #clear the environment

#rename graph for convenience
ht <- ht_advice

#get params for Erdos-Renyi simulation & print
nodes <- length(ht) # Nodes on the original graph
edges <- gsize(ht) # Edges on the original graph
nodes
```

```
edges


# First we compute the proportion of same-department edges on the original graph:
same_department_edges_original=sum(V(ht)$dept[ends(ht, E(ht))[,1]] == V(ht)$dept[ends(ht, E(ht))[,2]])/edges

# Secondly, we generate 10000 Erdos-Renyi graphs and compute the same quantity of interest
num_iterations <- 10000
same_department_edges_sim=numeric(num_iterations)

set.seed(1)

#main loop
for (i in 1:num_iterations) {
  #create erdos-renyi stylised graph with 21 nodes and 190 edges in G(N, M) mode, directed and without loops
  e <- erdos.renyi.game(
    nodes,
    edges,
    "gnm",
    directed = TRUE,
    loops = FALSE
  )
# Fix the original attributes of the nodes:
 e <- set_vertex_attr(e, "dept", value = vertex_attr(ht, "dept"))
# And compute the proportion of same-department edges on the simulated graphs:
same_department_edges_sim[i] = sum(V(e)$dept[ends(e, E(e))[,1]] == V(e)$dept[ends(e, E(e))[,2]])/edges
}

# Create a histogram and plot the results, add in dashed line the position of the original value of
# the quantity of interest
df = data.frame(same_department_edges=same_department_edges_sim)
g = ggplot(df,aes(x=same_department_edges)) + geom_histogram(binwidth = 0.01)
g + geom_vline(xintercept=same_department_edges_original, linetype="dashed", color = "#a91b0d",size=2)+
  annotate("text", x=0.25, y=1600, label="Same department edges original", angle=90,color = "#a91b0d")


# Compute p-value
cdf <- ecdf(sort(same_department_edges_sim))
cdf_value <- cdf(same_department_edges_original)
```

```
(p_value <- 1 - cdf_value)
# 0.1824
```

Code for Test 2:

```
rm(list = ls()) #clear the environment

#rename graph for convenience
ht <- ht_advice

#get params for Erdos-Renyi simulation & print
nodes <- length(ht) # Nodes on the original graph
edges <- gsize(ht) # Edges on the original graph
nodes
edges


# First we compute the proportion of same-level edges on the original graph:
same_level_edges_original=sum(V(ht)$level[ends(ht, E(ht))[,1]] == V(ht)$level[ends(ht, E(ht))[,2]])/edges

# Secondly, we generate 10000 Erdos-Renyi graphs and compute the same quantity of interest
num_iterations <- 10000
same_level_edges_sim=numeric(num_iterations)


set.seed(1)

#main loop
for (i in 1:num_iterations) {
  #create erdos-renyi stylised graph with 21 nodes and 190 edges in G(N, M) mode, directed and without loops
  e <- erdos.renyi.game(
    nodes,
    edges,
    "gnm",
    directed = TRUE,
    loops = FALSE
  )
  e <- set_vertex_attr(e, "level", value = vertex_attr(ht, "level"))

    same_level_edges_sim[i] = sum(V(e)$level[ends(e, E(e))[,1]] == V(e)$level[ends(e, E(e))[,2]])/edges
}
```

```
# Create histogram and compute p-value
df = data.frame(same_level_edges=same_level_edges_sim)
g = ggplot(df,aes(x=same_level_edges)) + geom_histogram(binwidth = 0.01)
g + geom_vline(xintercept=same_level_edges_original, linetype="dashed", color = "#a91b0d",size=2)+
  annotate("text", x=0.54, y=1400, label="Same level edges original", angle=90,color = "#a91b0d")

# Compute p-value
cdf <- ecdf(sort(same_level_edges_sim))
cdf_value <- cdf(same_level_edges_original)
(cdf_value)   # Probability on the left side -> p-value
# 0.029
```