Group assignment 2: Social network analysis and modelling

Date: 09-04-2021

Authors: Eva Steenhoven (9969632), Rosanne Vreugdenhil (4163869), Robin Reijers (5069769), and Mirthe Hendriks (6866999)

EXERCISE 1 Exercise 1 Build and analyse a small network from Facebook

Check out the Summary and Plot, how many friends do Douglas have on Facebook? Is this a directed or undirected graph and why? What is the meaning of the link between nodes in the plot? **[Question 1, 2 points].**

```
Network attributes:
  vertices = 93
  directed = FALSE
  hyper = FALSE
  loops = FALSE
  multiple = FALSE
  bipartite = FALSE
 total edges = 322
   missing edges = 0
   non-missing edges = 322
 density = 0.07526882

Vertex attributes:

 friend_count:
   integer valued attribute
   93 values

 group:
   character valued attribute
   attribute summary:
      BookClub        College
             5              7
        Family GraduateSchool
            23             10
    HighSchool          Music
             5             16
         Spiel           Work
             6             21

 mutual_friend_count:
   integer valued attribute
   93 values

 relationship_status:
   character valued attribute
   attribute summary:
                 In a Relationship
              42                 9
         Married             Single
              38                 4
```

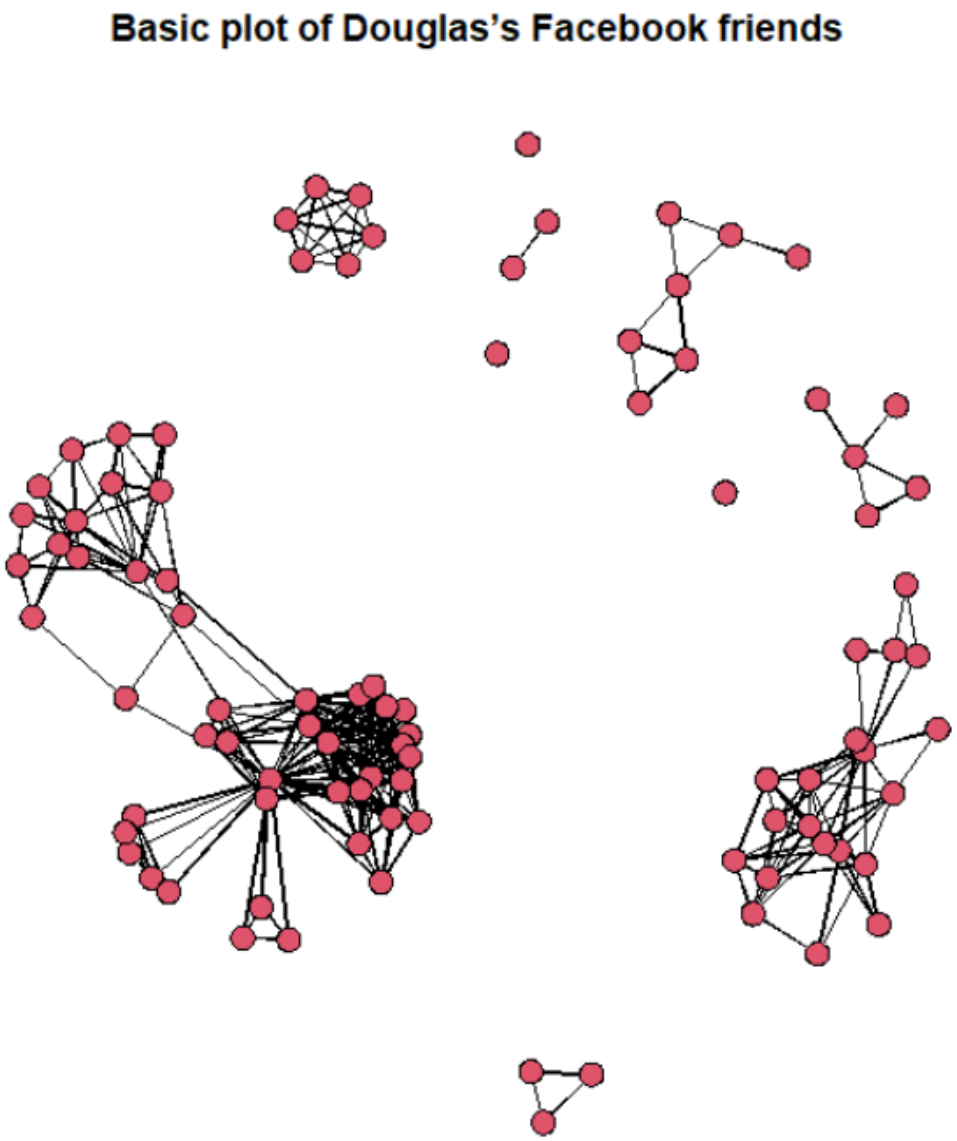
```
 sex:
   character valued attribute
   attribute summary:
female   male
    54     39
  vertex.names:
   character valued attribute
   93 valid vertex names

No edge attributes

Network edgelist matrix:
      [,1] [,2]
  [1,]   64   90
```

Friend count: Douglas has 93 friends

directed = FALSE, so it is an undirected graph. Besides, we do not see arrows in the plot. This makes sense because on facebook you can only be friends if you both agree (you have to accept someone's friend request to become friends).
The nodes in the plot represent the people - i.e. facebook friends of Douglas - and the links are the facebook connections between Douglas' friends.



Compare the degree, closeness and betweenness of Vertex 1 to the values of other nodes in the network. How will you evaluate the role of Vertex 1 in this network? **[Question 2, 3 points].**

```
> degree(facebook, v = 1, mode = "total") #friend nr 1
 1
32
> closeness(facebook, v = 1, normalized = TRUE)
          1
0.02120793
Warning message:
In closeness(facebook, v = 1, normalized = TRUE) :
  At centrality.c:2784 :closeness centrality is not well-def
ined for disconnected graphs
> betweenness(facebook, v = 1, directed = FALSE, normalized
 = TRUE)
         1
0.1411131
> eigen_centrality(facebook)$vector[1]
1
1
> transitivity(facebook, type = "localundirected", vids = 1)
[1] 0.3024194
```

The mean of degree is 6.92. Vertex 1 has a degree of 32 so this is much higher than the mean of the degree for other nodes in the network.
The mean of closeness is 0.0166. Vertex 1 has a closeness of 0.0212 so this is higher (about 1 standard deviation higher) than the value of the mean of the closeness for other nodes in the network.
The mean of betweenness is 0.0036. Vertex 1 has a betweenness of 0.0212 so this is much higher than the value as the mean of the betweenness for other nodes in the network.

Vertex 1 is a central node/ individual in Douglas facebook network. The score for degree centrality shows us that vertex 1 is connected to 32 other nodes, which is relatively high in comparison to the number of nodes adjacent to other nodes in the network (the mean of node connections is about 7). The score for closeness centrality implies that vertex 1 is less close - in geodesic distance - to other nodes than the average node in the network; it takes relatively more steps to reach vertex 1 from other nodes in the network.
The betweenness of vertex 1 is relatively high with a value of 0.0212. This means that compared to the other nodes in the network vertex 1 is relatively often used as a ‘bridge’ between nodes.

We expect that this node is located at the left cluster of the network, as this would explain the high degree centrality and betweenness while it has a high closeness centrality, implying that the node is less close to the other nodes.

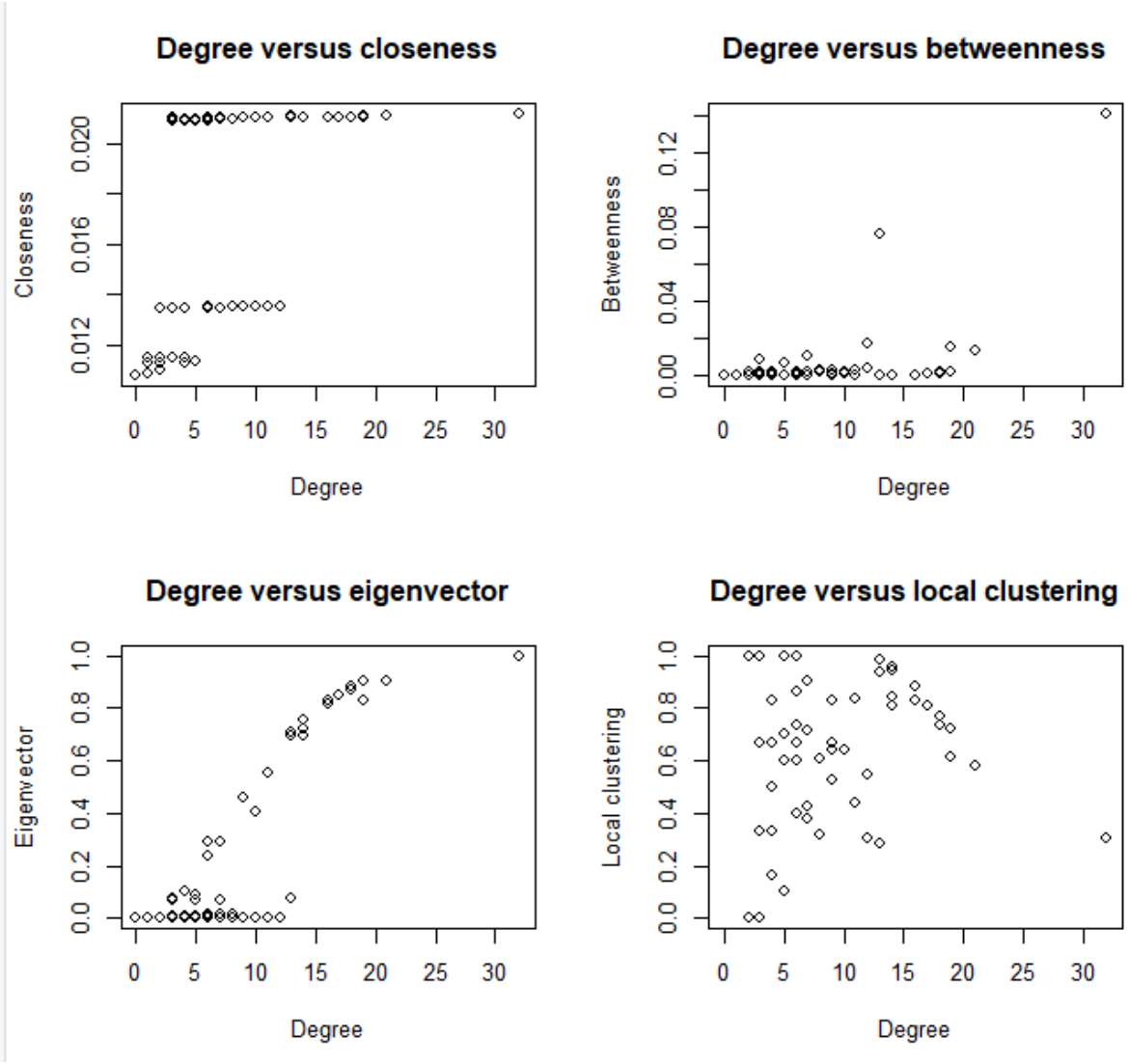
> mean(deg)
[1] 6.924731
> mean(cls)
[1] 0.01665854
> mean(btw)
[1] 0.00369126

> deg1: 32
> cls1: 0.02120793
> btw1: 0.1411131

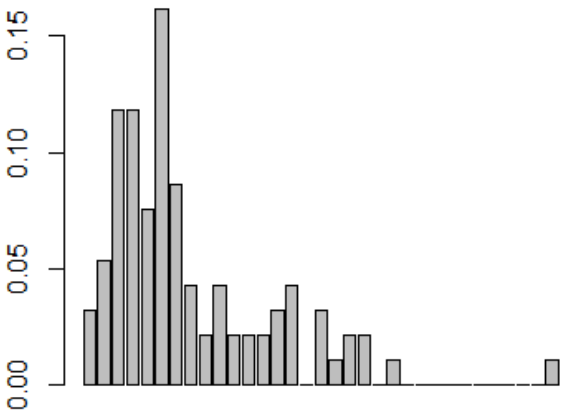
In the lecture, we introduced a few measures of centrality: degree, betweenness, eigenvector. Try to find top 5 nodes according to a) degree, b) betweenness, c) closeness and d) eigenvector. And develop the scatter plot between different metrics, you can refer to the code below. Discuss with your group and describe to your teacher, 1) how well does the top 5 nodes by different metrics overlap with each other? 2) why we need more than one metric to define centrality? **[Question 3, 5 points]**

Top 5 according to degree: 1 (32), 12(21), 23(19), 31(19), 26(18), 30(18)
Top 5 according to betweenness: 1 (0.141), 16 (0.076), 76 (0.017), 23 (0.016), 12 (0.014)
Top 5 according to closeness: 1 (0.02120), 16 (0.02111), 12 (0.02110), 23 (0.02110), 31 (0.02108)
Top 5 according to eigenvector: 1 (1.0000), 31 (0.9016), 12 (0.9013), 30 (0.8839), 26 (0.8678)

For the top 5 nodes according to these different factors, vertex 1 is consistently the most central node. However, we see differences in what other nodes are deemed central according to the various metrics of centrality. 16, 12, 23, and 31 appear multiple times in the different top 5s, but not always at the same place. We also see the relations between the degree and other metics of centrality in the scatter plots. We need more than one metric to define centrality, because the most central node in a network is dependent on how you define ‘centrality’. Therefore we need more than one metric, to get a more wholesome idea on what the central point in the network is.



Measure group-level metrics



Discuss within your group on how you understand each of these measures. And describe to your teacher, 1) why diameter should be larger than 1, and other metrics such as edge density and transitivity are smaller than 1? 2) is this a tightly knitted network? **[Question 4, 3 point].**

```
> edge_density(facebook)
[1] 0.07526882
> diameter(facebook, directed = FALSE)
[1] 4
> transitivity(facebook, type = "global")
[1] 0.6649123
> centr_degree(facebook)$centralization
[1] 0.2725573
```

The barplot shows the degree distribution. It gives the relative frequencies of the node-level degrees; how often each degree-value occurs in the data. The degree-values clearly are not normally distributed. The lower and left-centre degree values occur more frequently in the network. Most individuals in this network have a low to moderate number of connections, i.e. friends within Douglas' network. There is a right-skewed long tail which shows one relatively very high degree-value for a single node; relatively this person has an abnormally high degree, i.e. number of friends within Douglas' facebook network.

The diameter of the network is the longest of the shortest path between two nodes. The diameter should be larger than 1 because if you want to draw a path between two nodes, this path can never be smaller than 1. If you want to go from one node to another node, the diameter will by definition be 1.

Edge density is calculated by the number of connections a network has, divided by the total possible connections a network could have. Dividing a number by a bigger nummer will always lead to a number between 0 and 1, therefore it is always 1 or smaller.

The transitivity (i.e. the clustering coefficient) of the network would be the probability that your friends would be connected to each other. A value of 1 would mean that all the friends in the network are directly connected to each other, this would be the maximum value. A value of 0 would be that there is no direct connection between any of the friends, the minimum value possible. The network of Douglas has a transitivity of approximately 0.66, this would mean that about ⅔ of Douglas his network is also friends with each other.

Douglas' facebook network is not a super tightly knitted network. First of all, the edge density is quite low. Besides, see a few clusters in which the nodes do have many connections, but there remain a lot of unconnected nodes or nodes with only a few connections. We see this in the barplot with the degree distribution, where lower numbers of connections occur more frequently. This in combination with the relatively low diameter and low centrality of the network would suggest that there are a few friends that have a high degree, and are pretty central. But that most of the friends are not that well connected to others

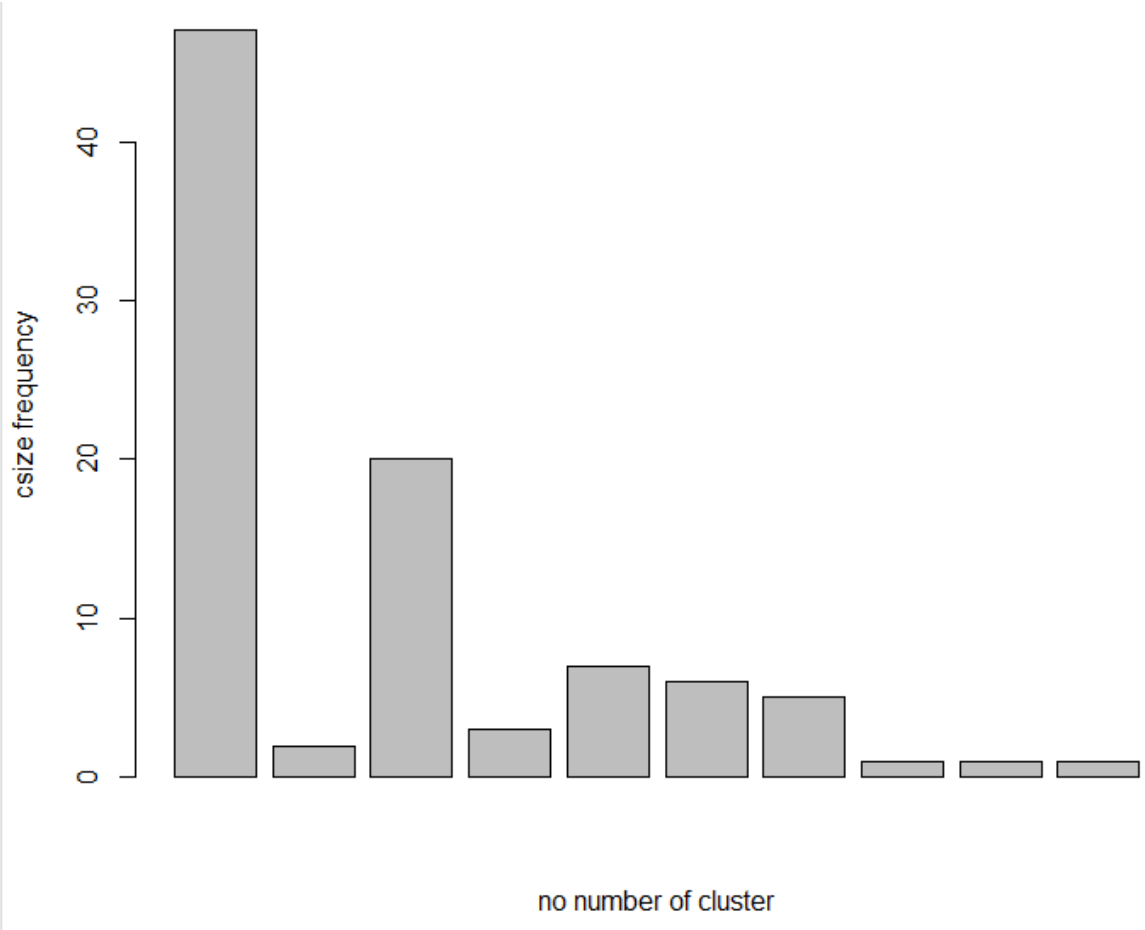
---

Detect the component and community

Reflect on what we have discussed about the Facebook network on Slide 47 of the lecture. Do you think this small network of Douglas resonates some general patterns of the entire Facebook network in terms of the components size and number? How can you explain such an observation **[Question 5, 3 point].at**

In the Facebook network on slide 47 we see a lot of connected components. Most components are relatively small, these are basically small groups in which nodes are connected. In the facebook network 99.91% of the nodes - i.e. individuals - belong to a single large connected component. This is different in Douglas' network; there are less nodes - i.e. individuals that belong to one single large component. We can see this in the visualization of Douglas' network; there is one large group on the left hand side, this is the largest connected component in the network. The network of Douglas is comparable to that of Facebook in that it has one bigger component, and all other components are much smaller than that one. This can be explained by specific groups like high school friends that are well connected. We see a smaller connected component in Douglas' network on the right hand side and many very small connected components.

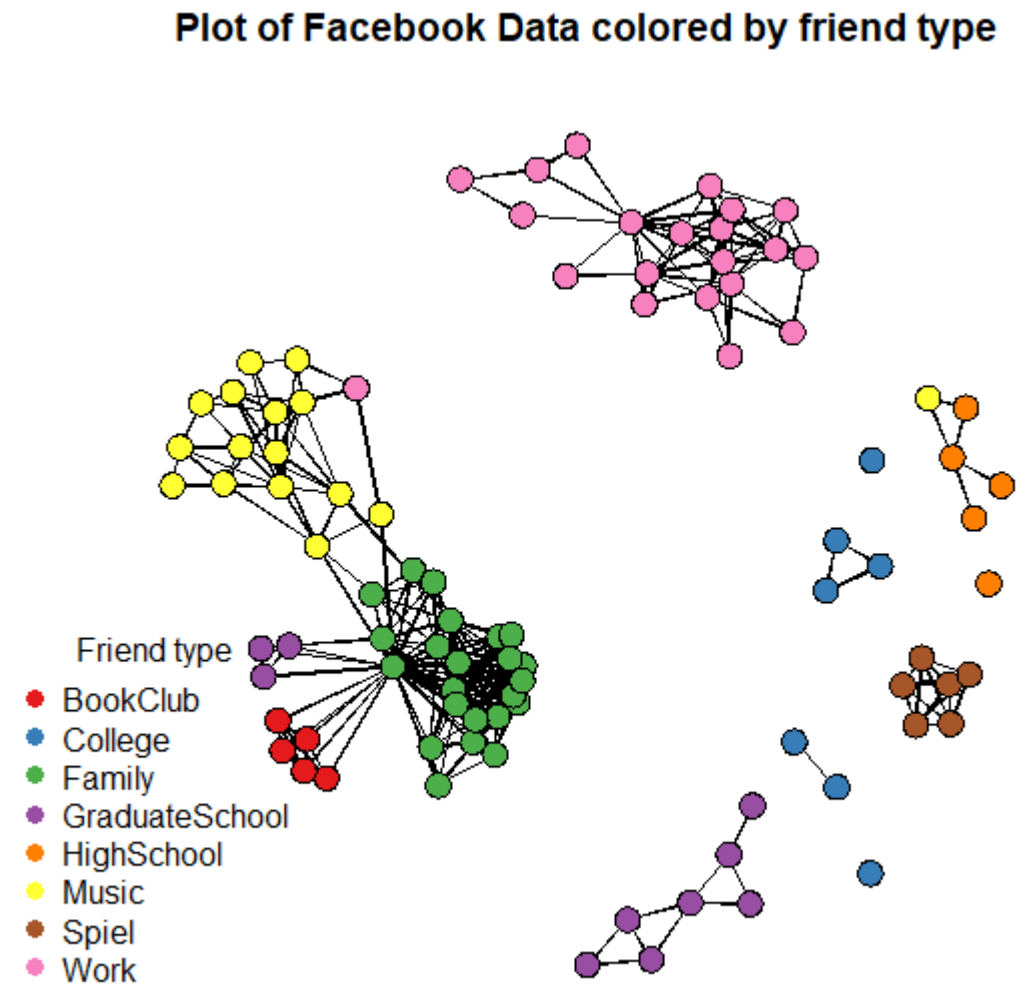
This can be explained by considering the amount of people in the network. In Douglas' network there are a less people than in the network of facebook, therefore Douglas' network is less extensive. If the amount of people is large enough there is almost always an indirect connection between two people in some way. This is in line with the six degrees of separation by Stanley Milgram, with a larger network this is more realistic. With a smaller network there is a smaller chance of being connected.



Based upon the plot you produced, discuss with your group and describe to your teacher the distinctions of the components. Do you find instances of intermingling of Douglas friends (i.e., belong to different groups but end up in the same component)? Do you find any isolated groups here? What can you conclude about the mixing of Douglas' Facebook friends? **[Question 6, 3 point].**

The network shows us instances of intermingling of Douglas' friends. His friends from music intermingle with family, his graduate school and his bookclub in the largest component. In other words, individuals from these groups have a facebook friend connection with individuals in another group. However, we do see only a few direct lines from the family group to the graduate school group and the bookclub, there seem to be only two individuals in the family group who are facebook friends with these other individuals. Besides, while the family group and music group are path of the same component, there are only few direct friend connections. Besides, there are some isolated groups in the network: spiel, college and work. Within this network, individuals from these group only form connections to other individuals from this same group, i.e. they don't intermingle with Douglas' other friends.

It seems like there is not that much mixing between Douglas' his friends. In the largest community there are some different groups, however these are not very well connected to each other. Most often the groups are connected with just one or two persons of the other group. Overall there is not that much mixing between the groups.



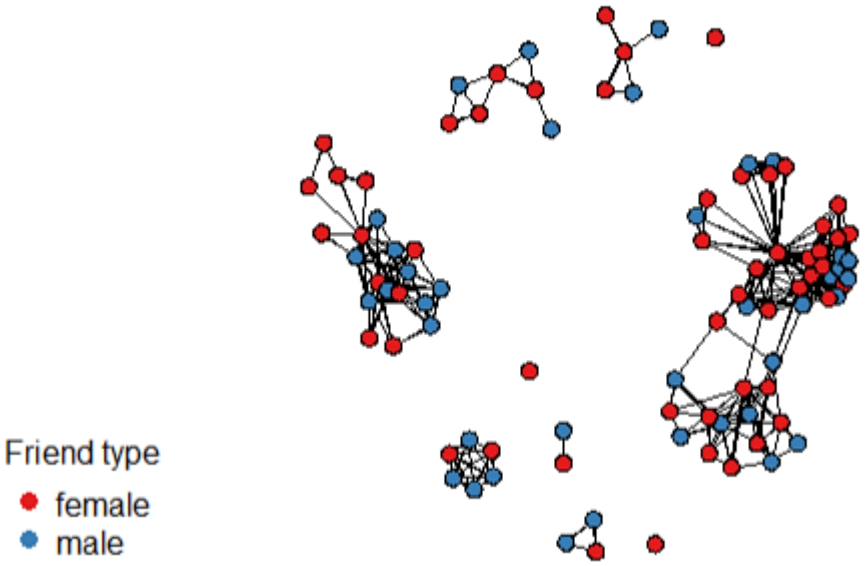
Using the above code as a reference, check out the attributes of other factors (e.g., sex, relationship_status) in terms of people in the same components. Note that you can specify the 'attrname' parameter within the function 'get.vertex.attribute'. Discuss with your group and describe to your teacher whether or not these factors are the keys in determining the formation of components **[Question 7, 3 point].**

The network shows us that these factors are not the keys in determining the formation of components. The factors - being male or female, or the relationship status - do not not define any of the groups. The network shows that the female - male division and the relationship status is unrelated to the component, because of the diverse distributions in the groups. We do not see one component in which all individuals are married for example. In each component (with more there one individual) there seems to be a mix of females and males, and most of the time there seems to be a mix of married individuals, individuals in a relationship or individuals of who the relationship status is unknown.

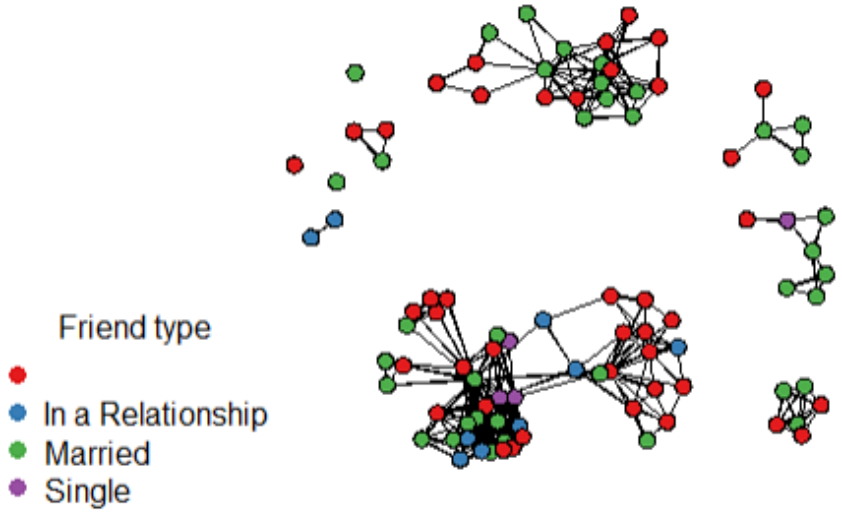
group <- as.factor(get.vertex.attribute(facebook_net, attrname = 'group'))

group <- as.factor(get.vertex.attribute(facebook_net, attrname = 'sex'))

Plot of Facebook Data colored by sex



Plot of Facebook Data colored by relationship status



From this analysis, what do you observe from the density values? Are they similar across different groups? What is the minimum and maximum value you observed here and how do you explain that? **[Question 8, 3 point].**

The groups differ in density. The density of a group tells the number of connections within the group itself, divided by the total possible connections within the group. The maximum density value observed is 1 in BookClub friends and Spiel friends, this is also the maximum possible value for edge density. The minimum value is 0.19 in the component college friends.

```
                                                    BookClub
                                  "Density for BookClub friends is 1"
                                                     College
                         "Density for College friends is 0.190476190476190"
                                                      Family
                         "Density for Family friends is 0.624505928853755"
                                                  GraduateSchool
"Density for GraduateSchool friends is 0.266666666666667"
                                                  HighSchool
                                "Density for HighSchool friends is 0.3"
                                                       Music
                          "Density for Music friends is 0.316666666666667"
                                                       Spiel
                                    "Density for Spiel friends is 1"
                                                        Work
                                   "Density for Work friends is 0.3"
```

The density for the bookclub and spiel is 1, which implies that all of the individuals in this network have a connection, in other words they are all facebook friends. In the family group we see that the majority (62 percent) of the individuals have a connection. The lower values for the college friends group, the graduate school group, the music friends, and work group indicate that most of the individuals in the group are not connected to all other individuals in the specific group.

To answer this question, search and discuss within your group on the theory of 'community detection'. Describe to your teacher what community detection is, and why it is useful to understand complex networks **[Question 9, 3 point].**

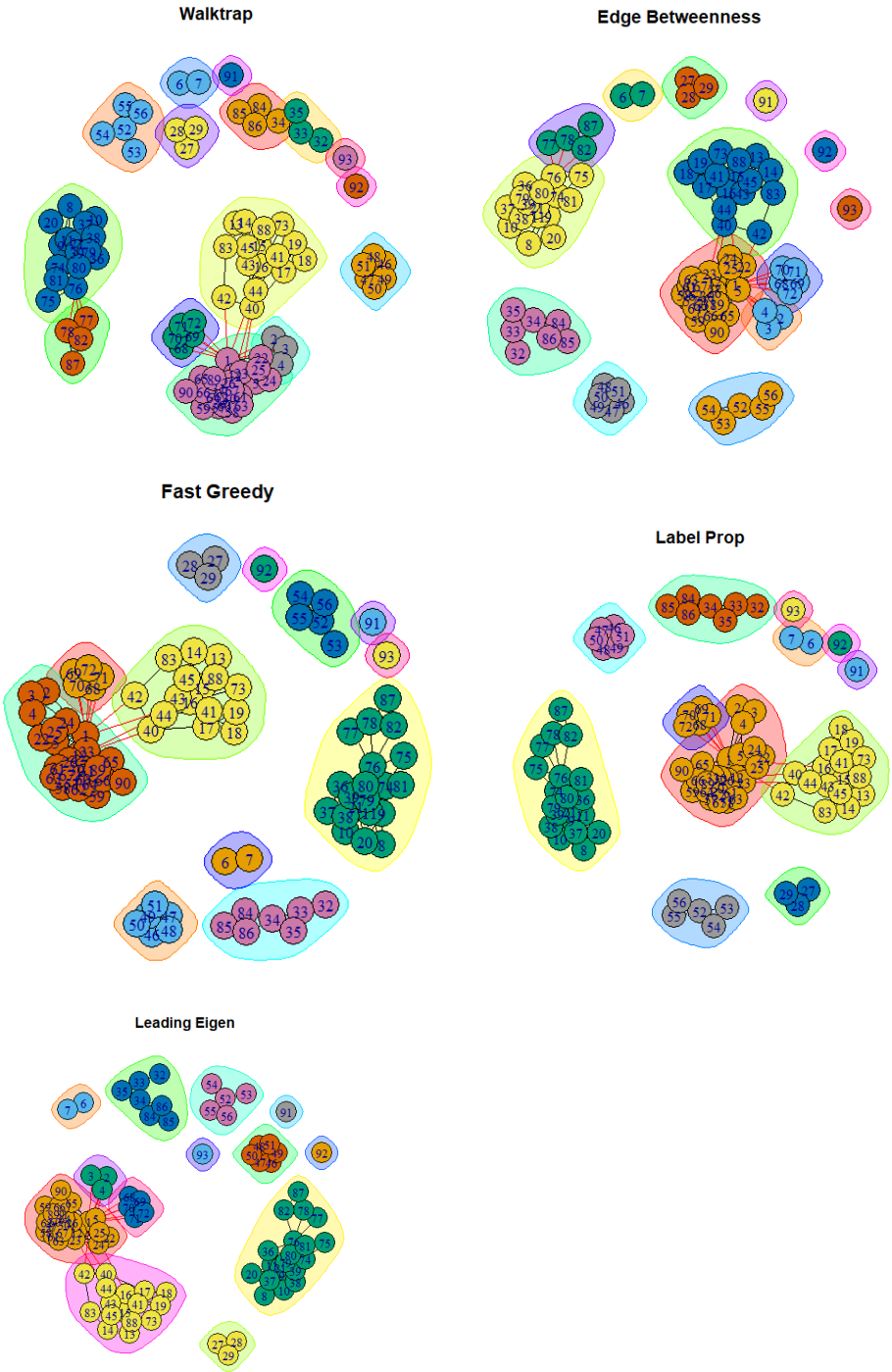
Community detection is the process of discovering groups or clusters in a network. It is often used to discover the structure of a network. Such a social network is made up of nodes and connections between the nodes, i.e. the edges. In other words, the individuals and the interaction between individuals. Communities represent the groups of people that tend to cluster together. For example, in a Facebook network or another social media platform, people with similar tastes, interests or opinions tend to cluster together. We can identify these groups using cluster detection.

It is useful to understand complex networks as they can allow us to create a large scale map of the network, as communities act like meta-nodes. This gives a better insight in complex networks. Communities also give us a better insight in how network functions and topology affect each other. In a network, communities also play a role in, for instance, rumour spreading. Studying them can give a better insight in the process of the spread of information, and the influence of communities in this process. Finally, another reason for studying communities is that they often have different properties than the average properties of the network. Studying communities can give a more detailed insight in the network.

Compare the plots that you generate from the different algorithms; do you find them similar and why or why not? **[Question 10, 2 point]**

The different algorithms are quite similar in the communities they generate. However there are some differences. In the graphs you can see that some of the communities are connected, but presented as two different communities while other algorithms present these communities as one.
An example of this are the communities on the left in the walktrap algorithm graph (blue/green & orange/green), you can see that this algorithm presents them as two separated clusters that are connected. In the Fast Greedy algorithm however these two communities are taken together and presented as one big cluster, which can be found on the right of the graph (green/yellow).

components and communities are similar



Discuss with your group and describe to your teacher: What is the societal problem that the authors are studying here? How can this problem be addressed using social network analysis? **[Question 11, 3 point].**

The societal problem that the authors are studying here is the polarization on the topic of vaccination due to social media platforms. The authors focus on the problem of echo chambers. Echo chambers are a phenomenon where people surround themselves with people that have similar opinions as they have, resulting in polarisation of the society. In this study, the authors conclude that due to the existence of echo chambers social-media

campaigns providing accurate information have limited reach in anti vaccination groups and can even have a further polarizing effect. This research gave an insight in the influence of how polarization can be caused by social media and how the problem can be addressed. They conclude that dissenting information in anti vaccination groups is not a good way to address the problem and can even create a backfire effect. It would be good to understand the contents of the echo chambers by getting passively involved in such groups, to find a better way to address the problem.

Discuss with your group and describe to your teacher: How many communities did the authors detect and how did they do this? Additionally, reflect on which of the communities you expect has the highest density, and which of the communities do you think has the lowest density. **[Question 12, 3 point].**

They detected two communities, the anti-vaccination and pro-vaccination community. The common users of the pages can be clustered together, into two clusters connected by the common use of pages. First, they considered the narrative of pages and the contents of the posts and manually classified the two main groups with two raters (145 pro-vaccine with 1,388,677 users and 98 anti-vaccine with 1,277,170 users). In turn they applied four well known community detection algorithms: FastGreedy, WalkTrap, MultiLevel, LabelPropagation. These algorithms allow for unsupervised clustering to find these clusters. They compared all the community classifications with the FastGreedy classification. They validated the community partition as well and this showed high agreement.

The density of a group is defined by the number of connections within a group divided by the total possible connections within a group. This study shows that the anti-vaccination community overtime grows more cohesively as a whole then the pro-vaccination community. The anti-vaccination community shows more cohessenives in new pages and in the community as a whole. Due to all the interconnection between pages and people, the anti-vaccination network is probably more dense. While on the other hand the pro-vaccination community is more fragmented on different pages, meaning that the interconnection between the users is limited and thus has a lower connection density.

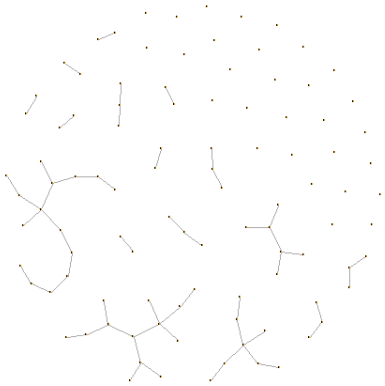
Exercise 2: Formulate a social network for certain architectures

Discuss with your group, then describe to your teacher, under which circumstances, we might need to work on a synthetic network. **[Question 13, 2 point].**

Synthetic networks can be used to test new algorithms or solutions to study the effect of network topology using for example controlled experiments. Classing potential 'real' networks without actually having the full data of one, either out of consideration of privacy or not having the (full) data available.

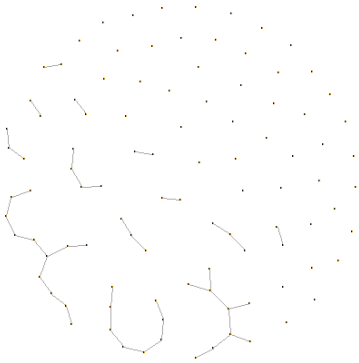
Plot your network (with n=100, and p=1/100) and compare with those with your group members. Are they identical? Explain why they are/aren't **[Question 14, 1 point].**

The networks are not identical, as they are randomly generated. A random model means that every time the script is run, it will randomly generate a network. Therefore the chances that there would be identical networks are extremely small and highly unlikely. The only constant is the probability of an edge for a certain note - p, which is 1/100.



ER Random Network: G(N,p) model

ER1 Random Network: G(N,p) model



ER Random Network: G(N,p) model

ER Random Network: G(N,p) model

Develop three networks with the same number of vertices (n), but different probability (p); Name them as ER1, ER2, and ER3. Develop the plots of ER1, ER2 and ER3, describe how these three graphs look differently as p increases and explain why. **[Question 15, 2 point].**

p value used

ER1: 1/100

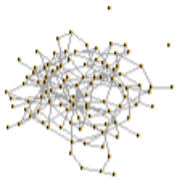
ER2: 1/50

ER3: 1/30

n is the number of nodes in the network. p is the value used to determine the probability of each edge in the graph with a probability closer to 1 it means that more edge connections will be made and the density of the plot becomes higher. This is what we see in or random network plots; the higher the probability of a random edge, the more edges appear.

ER1 Random Network: G(N,p) model

ER2 Random Network: G(N,p) model



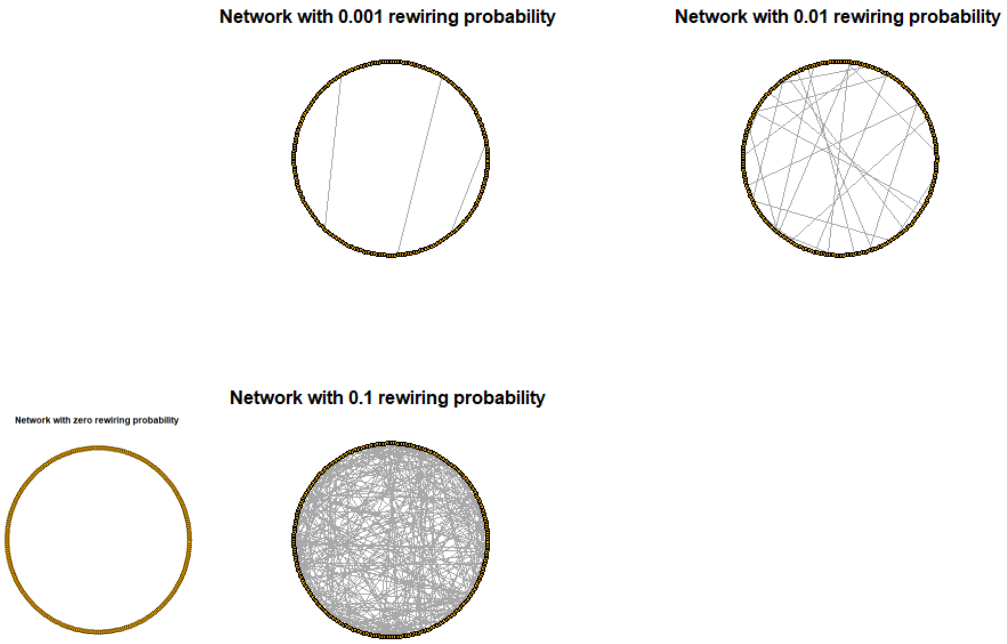
ER3 Random Network: G(N,p) model

If p<1, for n great enough, what happens to the clustering coefficient of an ER random graph and why? (You can use the 'transitivity' function to test your guess). Discuss with your group and describe the answer to your teacher. **[Question 16, 2 point].**

```
> transitivity(ER_100)
[1] 0.09713855
> transitivity(ER_200)
[1] 0.09994547
> transitivity(ER_500)
[1] 0.1018276
> transitivity(ER_1000)
[1] 0.09981423
> transitivity(ER_2000)
[1] 0.09986765
>
```

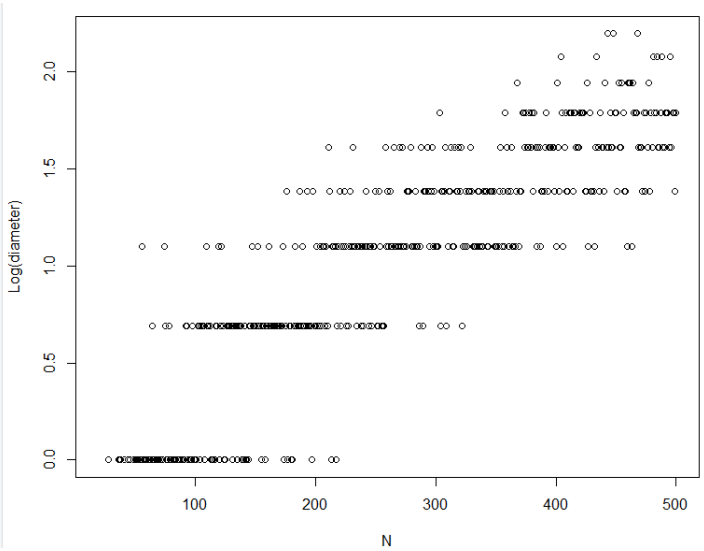
The clustering coefficient is consistent with your p value, the probability of an edge. This makes sense because the transitivity (i.e. the clustering coefficient) of the network is the probability that your friends would be connected to each other. We noticed that with a constant probability value, and with increasing number of nodes (n from 100 to 2000), the transitivity score simply fluctuates around the 0.1.

Small world model (Watts and Strotgatz model). It assumes that you know a certain number of persons (k) and that you are more likely to know your closest neighbors



For rewiring probability p=0.001, develop the networks for n from 20 to 500 and record the diameter of each network; Plot N~log (diameter); what do you find and how will you explain that? **[Question 17, 3 point].**

We see that with the increase of n, the diameter increases as well. The diameter of the network is the longest of the shortest path between two nodes. The small-world model introduces shortcuts between the clusters to reduce the diameter. The model is known for high clustering and low diameter. The plots show us that if there are less nodes in the network, the path between two nodes is overall shorter. If there are many nodes in the network (between 400 and 500), the shortest path between two nodes becomes relatively larger.



Check the clustering coefficient and average path length of the Regular, SW1, SW2 and SW3. Describe the trend of clustering coefficient and average path length as p increase. Does any of these graphs show the desirable attributes that you are looking for a small world network? **[Question 18, 3 point].**

transitivity (i.e. the clustering coefficient) of the network would be the probability that your friends would be connected to each other. The trend we see is that as p increases both the clustering coefficient and average path length decrease.

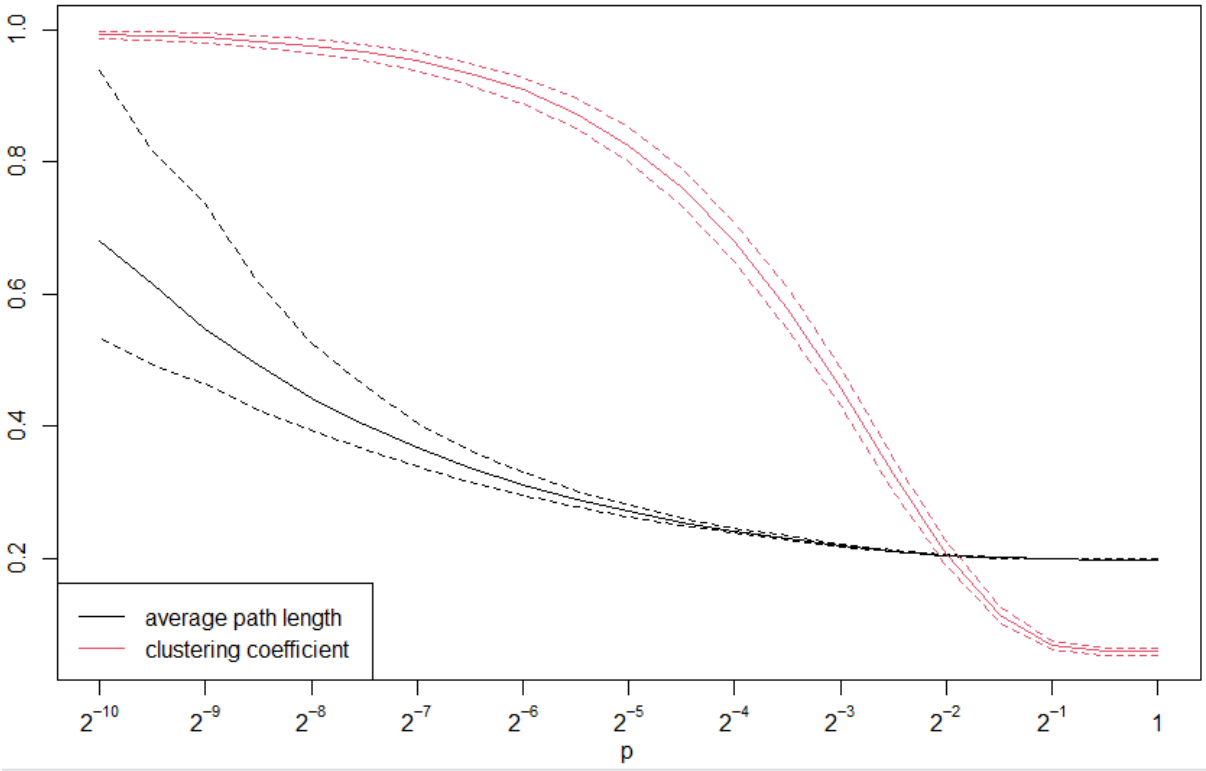
A small world network is characterised by a high clustering coefficient and a low average path length. If we consider the transitivity scores, the average path length scores and the graphs, we think that SW2 would be the most accurate small world network. The SW2 network has a transitivity score of 0.6414 and an average path length of 4.4334, compared to the other network this is a relatively high clustering coefficient and low average path length. This shows us that not every value of p can return a small-world network.

```
> transitivity(Regular)
[1] 0.6818182
> transitivity(SW1)
[1] 0.6777419
> transitivity(SW2)
[1] 0.6414096
> transitivity(SW3)
[1] 0.379216
```

```
> mean_distance(Regular)
[1] 12.95987
> mean_distance(SW1)
[1] 10.4899
> mean_distance(SW2)
[1] 4.737837
> mean_distance(SW3)
[1] 2.907603
```

For the same setting, i.e., size =300, nei=6, what are the range of *p* you will suggest to build a small network and **why**? One solution you can consider is to refer to the Figure 2 in Watts and Strogatz (1998), which explains the properties of small-world network for the family of randomly rewired graphs. Reproduce Figure 2 in the current context (i.e., size=300, nei=6).

Because we saw last week that many of you are working with R for the first time, we provide an example code segment of how you would generate a graph similar to that of Watts & Strogatz (1998). Discuss what the following code segment does in your group and interpret the resulting graph. **[Question 19, 5 point].**



The range of p we would suggest to build a small-world network is from approximately p = 0.0078 to p = 0.031. A small world network is characterised by high clustering coefficient and a low diameter, as it introduces shortcuts to reduce the diameter. The clustering coefficient is hard to reduce, but path length is not, so the idea behind the small-world model is to find a range of p that allows for a shorter path length. For the suggested range the clustering coefficient is relatively high & the average shortest path relatively low.

As a follow-up question of Q19, do you find the p values of very large or relatively small? What are its implications? **[Question 20, 3 point].**

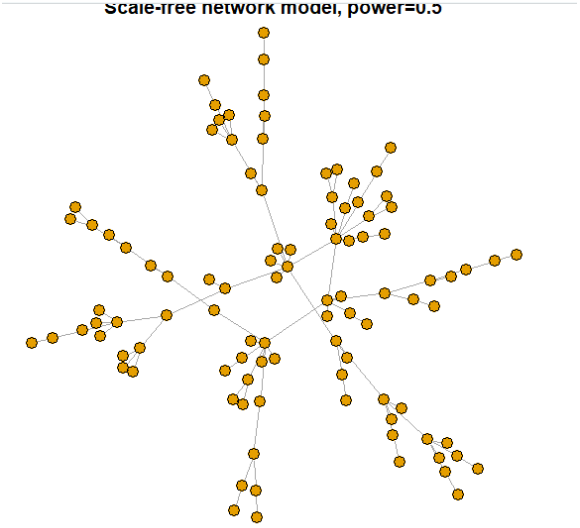
Comparing the p values to the range selected by Watts and Strogatz, we find a similar range of p. This was of course to be expected because we generated the graph similar to that of Watts and Strogatz. For each edge with a rewire probability, the network moves the end of the edge to a node on the other side of the network. This implies that in the small-world network, these shortcuts are created to join remote parts and this reduces the diameter. From the range we deduce that even with a relatively low rewire probability, enough shortcuts are created to significantly reduce the diameter.

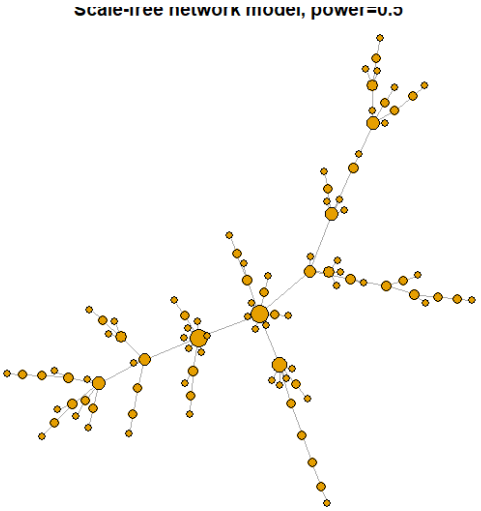
Scale-free graphs according to the Barabasi-Albert model

What does the power in the above function mean? How can it govern the structure of the network (e.g., the formulation of hubs)? (Hint: Change the value of power from 0.05, 0.5, 1,

1.5; See how the plot change; if you still fail to see the difference, visualize the vertex size according to the edge number, you can consider the code below.) **[Question 21, 3 point].**

The Barabasi-Albert model uses a probabilistic mechanism: A new node is free to connect to any node in the network. However, if a new node has a choice between a degree-two and a degree-four node, it is twice as likely that it connects to the degree-four node. This is the "rich get richer"-idea; the more friends one node has, the more likely the new nodes will be friends with him. The power in the function above is determining the centrality of the hub. The higher the power the more nodes are connected to the central hub.





Discuss with your groups, if you are maintaining a network with a power of 0.5 and 1.5, respectively, what will be your plans to build up resilience for random and targeted attack? **[Question 22, 3 point].**

Random attack is that nodes are randomly removed. In a targeted attack the most connected nodes are removed. For the network with a power of 1.5, the targeted attack is a bigger problem as it has one strongly connected node, therefore removing that node will destroy the network.
In case of a random attack, it would be wise to protect the nodes/people by using the 'acquaintance immunization' strategy. This says that a random acquaintance of a random node will be protected. Protecting the network this way will take a lot of time and effort.
In case of a targeted attack it would be wise to protect the hubs. By protecting the hubs a large proportion of the network will be protected. This is a relatively fast way of protecting the network, the only downside of this strategy is that locating the hubs can be somewhat difficult.

Download the rds data of these network from the BB, import the data to R and build the network. Check out their network attributes. Do you find these real networks show some attributes of the synthetic architectures we studied above (e.g., random ER graph, small-world, and scale-free network)? Show your teachers some numbers, plots and how you interpret the results. **[Question 23, 6 point].+**

A small world network is characterised by a high clustering coefficient and a low diameter. A random ER graph is characterised by a low clustering coefficient and a low diameter. A scale-free network is

We think that the Brightkite network resembles the random ER graph as it shares the attributes of a low transitivity and a small diameter. We think Amazon resembles a random ER graph as well, due to the *relatively* small diameter, low transitivity and low centralization. The collaboration network resembles the small-world network as it shares the attributes of high transitivity and a small diameter.

Amazon

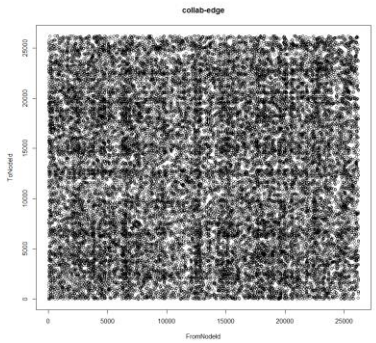
- Diameter: 44
- Transitivity: 0.2052244
- Edge density: 1.651383e-05
- Centralization: 0.001622968

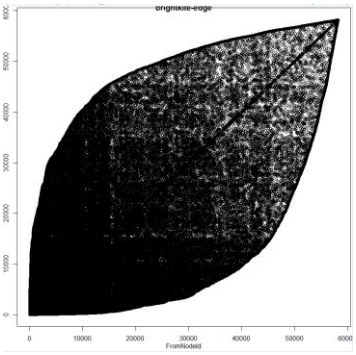
Brightkite

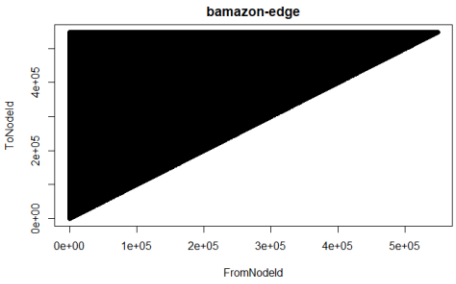
- Diameter: 16
- Transitivity: 0.1105669
- Edge density: 2.525665e-04
- Centralization: 0.03869844

Collaboration

- Diameter: 17
- Transitivity: 0.6298425
- Edge density: 2.109685e-03
- Centralization: 0.02880045

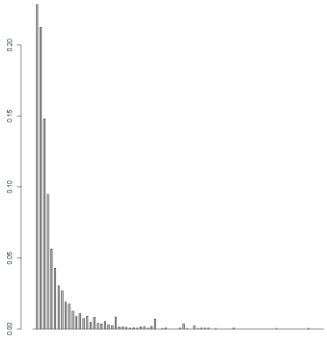






colab

amazone degree distribution



Exercise 3: Build the social network of this class and simulate the contagion process

The fourth and fifth questions are designed to collect the parameters that we will need to build 1) an independent cascade (IC) model and 2) a threshold model. Can you see which question is for the IC model and which one is for the threshold model? **[Question 24, 1 point].**

The Independent Cascade Model is an information diffusion model where the information flows over the network through cascade. Nodes can have two states, active: it means the node is already influenced by the information in diffusion, inactive: node is unaware of the information or not influenced. Cascade size is the fraction of adopters. This is visualized with a diffusion curve, which shows a more smooth pattern. Besides, the threshold model holds that to adopt a novel behavior, an individual needs to be convinced by an absolute number or a fraction of his/her social contacts.

We would say that question four is for the IC model, because it would allow you to model diffusion as a fraction of the group of students. The more people you talk to frequently after class, the faster diffusion would work. Question five is better suited for a threshold model, because it asks for likelihood with an absolute number. With what threshold would you share something with other students in the class.

Download the file “Class network survey.xlsx” from BB, check out the response of Q4 and Q5. Among the three types of behaviors, which one is the least contagious? Which one is the most contagious? And why? **[Question 25, 2 point].**

In question 4 the most contagious would be the paper related to the lecture and the least contagious would be the vegetarian recipe. We decided on this by looking at the sum/mean of the likelihood that something would be shared. The higher the sum/mean the more likely it is that people would share something, so that would mean that it is more contagious. For question 5, we looked at the lowest sum as this indicates that a low number of people is needed to reach the threshold. In this case ‘try a vegetarian recipe’ is the most contagious.

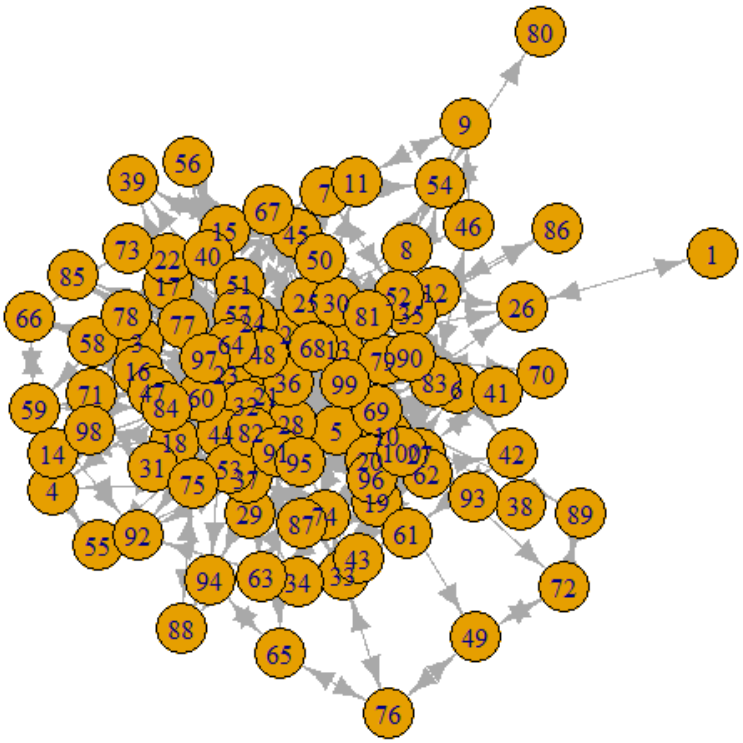
For simple illustration here, you can just check the degree of this network, i.e., the number of friends and close friends and their response to Q4 and Q5 in “Class network survey.xlsx”. Do you think network size can explain the differences in their likelihood to share and their thresholds to adopt? And why their answers are dependent/ independent on network size here? [Question 26, 2 point].

We calculated the correlation between the degree and the responses to Q4 and Q5. We see that correlation between the degree and the likelihoods is very low. This indicates that there is no strong relation between the size of the network and the likelihood of the person to share or adopt something.

```
     cShare_Paper      cs_veg cShare_Yout   cTh_paper    cTh_veg   cTh_Yout
[1,]   0.04471796 0.008543078  0.02376913 -0.02052584 0.01087854 0.03243235
```

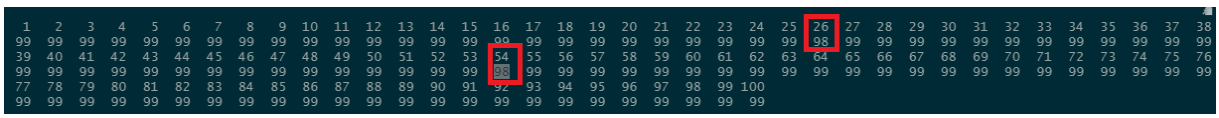
Find a single node (n=1) that after removal, will lead to the greatest decrease in the size of the largest component. Show your teacher the ID of node and describe your observations. **[Question 27, 3 point].**

By performing a brute force technique we discovered that the removal of node 26 & 54 led to the greatest decrease in the size of the largest component. In the results there can be seen that the removal of these nodes resulted in a greatest component size of 98, while the removal of all other nodes resulted in a greatest component size of 99.



```
> component_reduction(g)
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23
 99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99
 24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46
 99  99  98  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99
 47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69
 99  99  99  99  99  99  99  98  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99
 70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92
 99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99  99
 93  94  95  96  97  98  99 100
 99  99  99  99  99  99  99  99
```

To answer the above question, you might search explicitly to remove the 100 nodes one by one from the network. But can you apply such explicit search if you try to find out a set of nodes (i.e., n>1) that will lead to the greatest decrease? **[Question 28, 2 point].**



In the case of using the above method (using code from mister K. Wittenberg) node 26 and 54 have a decrease in component size (from 99 to 98) when compared to the rest of the network.

This method cannot be applied to find a set of notes which leads to the greatest decrease in component size. For example it is likely that two people with the biggest component sizes are in the same grouping of people removing one of those would still

mean the group is connected with each other. While it could be that someone with lesser component size actually has a more valuable connection between groups.

In the lecture, we mentioned two board categories of approximation algorithms. One is by heuristics such as degree, closeness, betweenness and eigenvector. The other is by greedy algorithm. Using 1) degree heuristics 2) betweenness heuristics and 3) greedy algorithms, find out a set of 5 nodes that should be removed to produce the greatest decline in component size. Compare the solutions provided by different algorithms. Do you find significant differences between the efficiencies of these algorithms and why? **[Question 29, 5 point].**

Degree heuristics means that the node with the most degrees will be removed from the network. There is assumed that the removal of the node with the highest degree will result in the greatest decline in component size. Based on the degree heuristics we would remove the following 5 nodes: 68, 21, 69, 64, 36. These are the nodes with the highest degrees in the network.
Betweenness heuristics mean that the node with the highest betweenness value will be removed. As nodes with a high betweenness are often the 'bridge' between two components, they assume that the removal of nodes with a high betweenness can disrupt the network. According to the betweenness heuristics the following 5 nodes should be removed to produce the greatest decline in component size: 68, 69, 44, 21, 64.
For the greedy algorithm, the following 5 nodes should be removed to produce the greatest decline in component size. The results show that removing the following 5 nodes would have the greatest impact on the component size: 56, 54, 2, 3, 4.

There can be seen that these three methods all propose different nodes that would disrupt the network most by removal. The nodes proposed by the two heuristic algorithms are quite similar, however the greedy algorithm proposes very different nodes. An important difference between these algorithms is that the greedy algorithm considers the influence of removing a certain node on the network when the previous node is removed from the network. The heuristics methods base their results on the full network and do not remove the influential nodes before computing the next node that would disrupt the components the most.

```
> deg <- degree(g)
> sort(deg, decreasing=TRUE)
 68  21  69  64  36  44  57  20  37  82  91  28  32  97  25   5  24  48  90  35  47  52  95
 62  40  40  38  34  30  28  24  24  24  24  22  22  22  20  18  18  18  18  16  16  16  16
 10  13  16  17  22  27  30  60  78  83  96   2  12  19  23  29  40  51  58  61  75  99 100
 14  14  14  14  14  14  14  14  14  14  14  12  12  12  12  12  12  12  12  12  12  12  12
  6   7  15  59  62  77  79  81  84   3   4  18  26  33  34  43  45  53  54  66  73  74  87
 10  10  10  10  10  10  10  10  10   8   8   8   8   8   8   8   8   8   8   8   8   8   8
 92  94  98   8   9  11  14  31  39  41  42  46  49  50  55  56  63  65  67  70  71  72  76
  8   8   8   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6
 85  88  93  38  86  89   1  80
  6   6   6   4   4   4   2   2
>
```

```
> bet <- betweenness(g)
> sort(bet, decreasing=TRUE)
        68         69         44         21         64         91         36         57         20         82         37
2142.5243607 1084.5084623 816.4433401 775.5951393 714.2344614 668.4417602 559.5957535 443.7631332 412.8515358 374.7435581 364.7780292
        25         28         48         32         24         90         54         12         27         26          5
 315.6934060 281.9668576 272.9884418 266.7710286 235.3174493 224.7267230 218.4470296 210.4310217 208.9918429 208.5936497 194.3916058
        75         35         96         61         52         83         16         95         97         33         47
 178.1153595 170.1067063 168.8536548 164.6312309 163.0234866 159.2814015 156.3485741 156.2120553 145.5866060 132.2342269 130.3698737
        60         84         78         11         23         10         22         40         17         51          6
 129.5749032 125.3196973 121.7776634 118.2883922 114.4944248 107.0161032 99.2583816 98.8576406 98.4336938 91.3538107 88.6996039
        58        100         87         94         81         15         30         49         73          2         53
  86.0700316 83.2782735 82.0039080 81.5170885 80.7828064 79.5966815 79.1742363 77.7320508 75.5163781 72.6172759 72.0801564
        19         79         29         89         92          7         13         65         62         98         59
  68.9643559 68.9629034 65.7811232 65.1415094 63.8225626 63.1223377 62.5571803 59.9745356 57.3491207 54.3576177 52.0698179
        46         99          8         41         74         45         77         43         72         55         56
  45.2105754 44.5471641 43.3007021 39.8180365 37.9159744 35.8506691 34.5203730 29.0532098 28.7168452 27.4924825 27.2004847
        66          3         70          4         34         39         71         76         42         50         14
  25.5831613 24.9728991 24.4608567 23.4075448 23.0724450 20.0479197 19.9233729 18.2250874 17.6261957 16.7745457 16.4924332
        18         88          9         63         31         86         67         38         85         93          1
  16.2004306 15.8511905 13.0450400 12.9484704 12.4689539  8.3839262  5.8561272  4.4298330  1.5766234  0.9183945  0.0000000
        80
   0.0000000
```

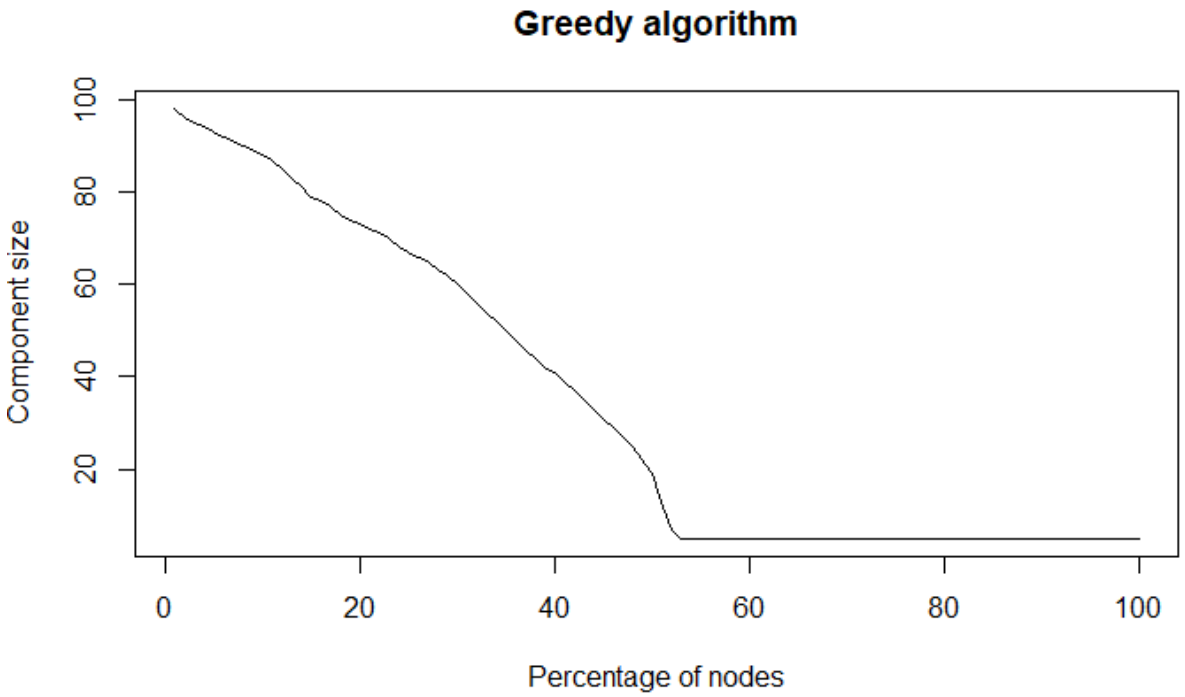
```
> greedy_component(g, k=5)
$node
[1] 26 54  2  3  4

$component_size
[1] 98 96 95 94 93
```

(component_size after deleting the nodes)

Try to find out the percentage of nodes that you should take out from the network that, by doing so, the size of the largest component will decrease dramatically. You should compare the percentages you find according to the greedy algorithms, degree heuristic and betweenness heuristics. Describe the answer to your teacher with the supported plots. What's the potential implication if you consider vaccination strategy for this small community? **[Question 30, 5 point].**

According to the greedy algorithm about 50% of the nodes in a community should be vaccinated to protect the whole community.



(Hints: you should develop a plot where the x-axis is the percentage of nodes that you take out (q), and y-axis is the size of the largest component after you take out q (G(q)).)

Build an IC model according to these probabilities. In this model, everyone has two states: S=0, unadopted, and S=1, adopted. Everyone starts from S=0. By seeding, you change the initial states of some nodes in the network from 0 to 1. These seeds pass the signal to their neighbors according to the probabilities we assigned them, which you will find in the node attributes. If you seed node 1, for example, it has a 45% probability to share a Youtube video. A random number will be generated between 0 and 1, and compare with 45%. If the random number is smaller than 45%, node 1 will pass the signals to its neighbours. For every seed and activated nodes, they only send out signal to their neighbours at the timestep right after their own activation. At other time steps, they remained activated but cannot send out signal again.

If you can seed only one person, who will you choose? Will you choose the same person to promote these three different things? **[Question 31, 5 point].**

For influence maximization, logically you would seed the most popular person in the class; the one with the most friends. To get a substantiated conclusion on which person i.e. node, to seed for influence maximization use a greedy algorithm with the IC (Independent Cascade) model.

For youtube

```
  |======================================| 100%
nodes1  nodes2  nodes3 spread1 spread2 spread3
   6.0    21.0    68.0    71.0    60.2    58.2
```

for vegetarian recepties

```
  |======================================| 100%
nodes1  nodes2  nodes3 spread1 spread2 spread3
   6.0    48.0    68.0    83.6    83.2    60.6
```

For paper

```
  nodes1  nodes2  nodes3 spread1 spread2 spread3
    6.0    21.0    68.0    46.0    43.2    46.0
```

For the diffusion of a youtube video you would seed node 6, for the diffusion of a vegetarian recepty you would seed node 6, and for the diffusion of the paper you would seed node 6. Evidently, you will choose the same person to promote these three different things. We see that some nodes would also be influential in the promotion of all three things, such as node 68. The output also shows the scores of the spread if you would seed the node in question. But as this is a greedy algorithm, another run might provide us with different nodes.

You now have a little more time or budget to seed 3 people. Using greedy algorithms and degree heuristics to 1) find out the seeds for Youtube, vegetarian recipe and paper, respectively. 2) the differences of the performance by degree and greedy algorithms. 3) Check out the network attributes (e.g., centrality measures) and probabilities of the seeds provided by the greedy algorithms. What do you find? **[Question 32, 5 point].**

Greedy

1) Youtube

```
  |======================================| 100%
nodes1  nodes2  nodes3 spread1 spread2 spread3
   6.0    21.0    68.0    71.0    60.2    58.2
```

2) Vegetarian recipe

```
  |======================================| 100%
nodes1  nodes2  nodes3 spread1 spread2 spread3
   6.0    48.0    68.0    83.6    83.2    60.6
```

3) Paper

```
  nodes1  nodes2  nodes3 spread1 spread2 spread3
    6.0    21.0    68.0    46.0    43.2    46.0
```

Degree

```
> deg <- degree(g)
> sort(deg, decreasing=TRUE)
 68  21  69  64  36  44  57  20  37  82  91  28  32  97  25   5  24  48  90  35  47  52  95
 62  40  40  38  34  30  28  24  24  24  24  22  22  22  20  18  18  18  18  16  16  16  16
 10  13  16  17  22  27  30  60  78  83  96   2  12  19  23  29  40  51  58  61  75  99 100
 14  14  14  14  14  14  14  14  14  14  14  12  12  12  12  12  12  12  12  12  12  12  12
  6   7  15  59  62  77  79  81  84   3   4  18  26  33  34  43  45  53  54  66  73  74  87
 10  10  10  10  10  10  10  10  10   8   8   8   8   8   8   8   8   8   8   8   8   8   8
 92  94  98   8   9  11  14  31  39  41  42  46  49  50  55  56  63  65  67  70  71  72  76
  8   8   8   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6   6
 85  88  93  38  86  89   1  80
  6   6   6   4   4   4   2   2
```

Degree: 68, 21, 69

```
> centr_degree(g)$centralization
[1] 0.251607
```

What we see is that the greedy algorithm suggests different seeds than the degree heuristic. For the IC_greedy node 6 was a popular node to seed. Nevertheless, this node does not have the highest degree, i.e. the highest number of connections. It could however be that this node 6 is very central in the network and connects different (groups) of people. When comparing the IC_greedy and the degree heuristics we also see that nodes 68 and 21 are output by both, apparently these nodes would have a large influence and allow something to diffuse quickly through the network.

node 68, youtube; 75, vegetarian; 81, for paper 36

Probability of sharing & audience of the nodes provided by the greedy algorithm:

Youtube : 6 - 0.85 & 8.5 , 21 - 0.85 & 34, 68 - 0.85 & 52.7

Vegetarian : 6 - 0.85 & 8.5, 48 - 0.85 & 15.3, 68 - 0.85 & 52.7

Paper : 6 - 0.65 & 6.5, 21 - 0.65 & 26, 68 - 0.45 & 27.9

The probabilities of sharing are overall very high, this is in line with the approach of the greedy algorithm. The greedy algorithm considers, besides the probability of sharing, the degree of the nodes as well. The nodes that will share the information with the biggest audience (AUDIENCE = Popularity of seeds * probability of sharing) will be chosen as seeds.

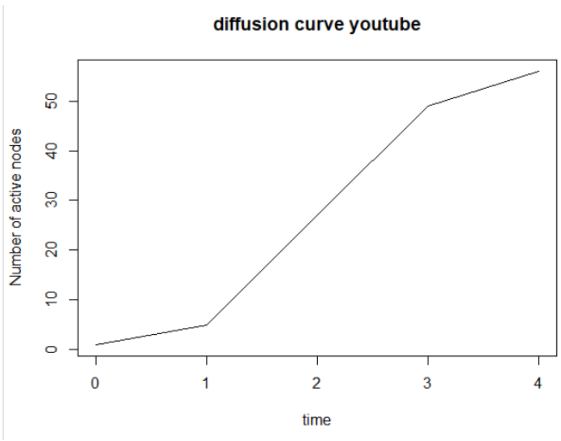
Recall the diffusion curve we mentioned in Lecture 3, slide 27. For an ideal case, the diffusion curve is S-shaped, which you can find the 'social tipping' point at the waist of the S. Produce the diffusion curves (x-axis as time, y-axis as the percentage of people being activiated) based on the 3 seeds that you selected according to the greedy algorithms, for Youtube, vegetarian recipe and paper, respectively. Do you find any

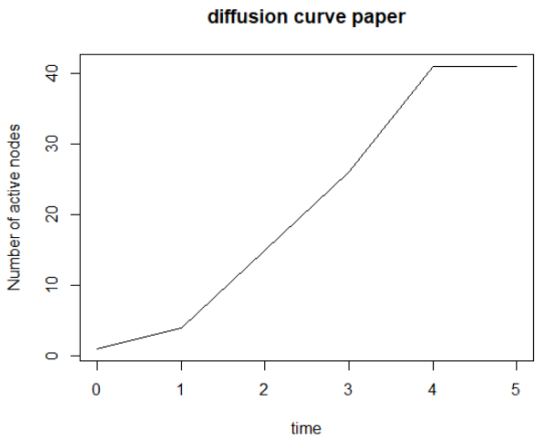
standardized S-shape curve? And explain how you interpret their shapes. **[Question 33, 3 point].**

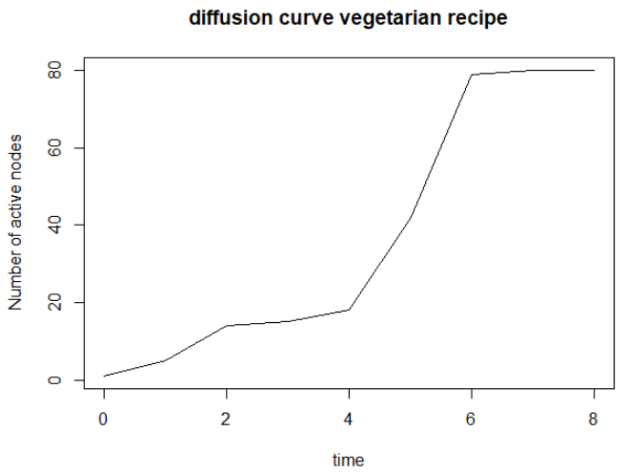
To create the diffusion curve for Youtube, vegetarian recipe and paper we initially seeded only one node, which was according to the greedy algorithm node 6 for all three cases. For each of them, this gave a somewhat crooked, S-shaped curve. The diffusion curve of Youtube has the most standardized S-shape. It has a steep rise in its active nodes through the timesteps and it reached the early majority stage of around 50% quite quickly. The diffusion curve of Paper however, does not reach the early majority stage as it stagnates at around 40%. The vegetarian recipe is slow in reaching its early adopters but then has a steep rise to the late majority stage of around 80% of the nodes reached.

When we implemented the three nodes given by the greedy algorithm for each case separately. none of the diffusion curves had a standardized S-shape curve. Each of the curves have a steep rise from the start. This makes sense as that 3 nodes on a total of 100 is quite a lot and can therefore have a great influence on the diffusion. The curve of Paper does stagnate again, this time at 50%. With 3 seeded nodes it therefore does not reach complete diffusion either. The Vegetarian recipe and the Youtube curve do not reach complete diffusion yet but it can be seen that the diffusion keeps on increasing with each timestep.

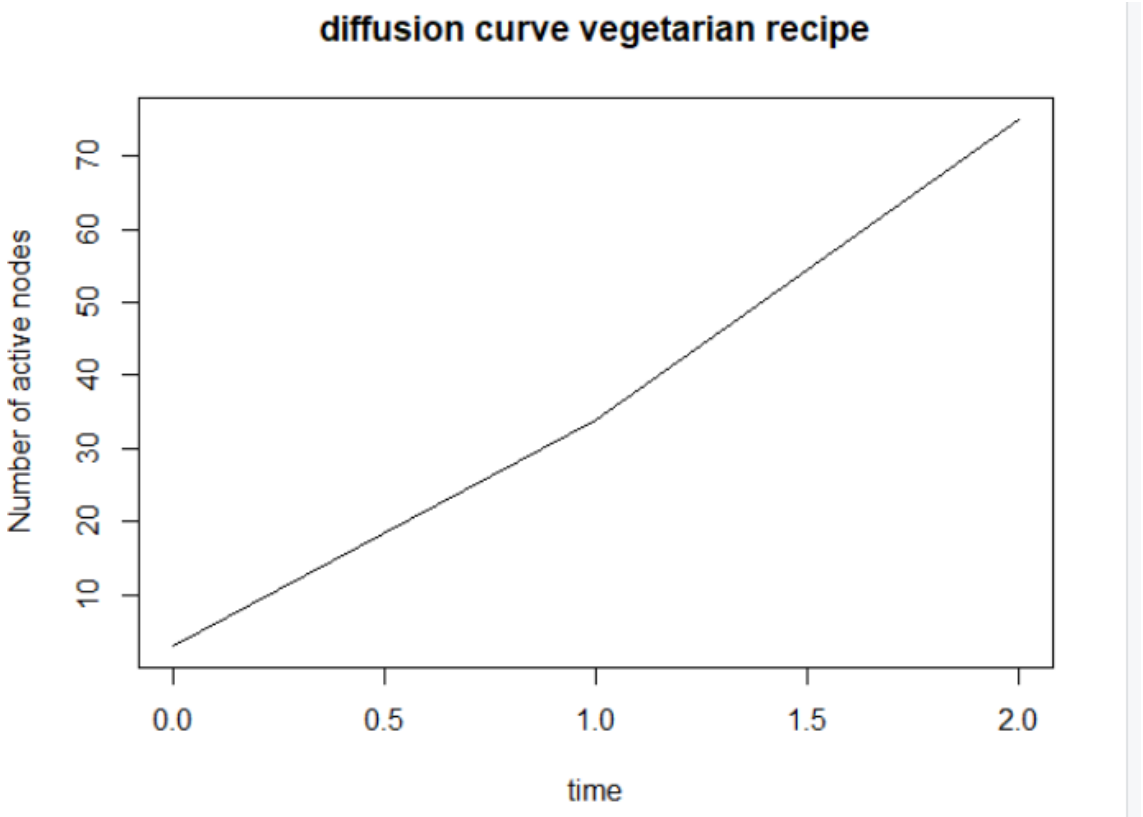
Seeding only node 6:

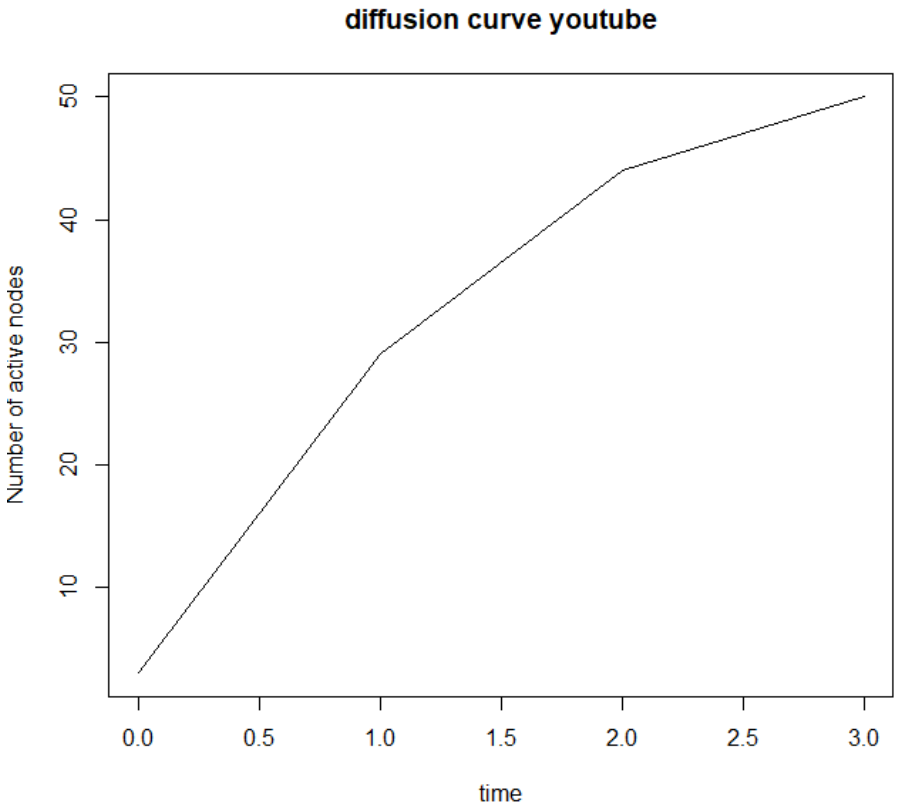


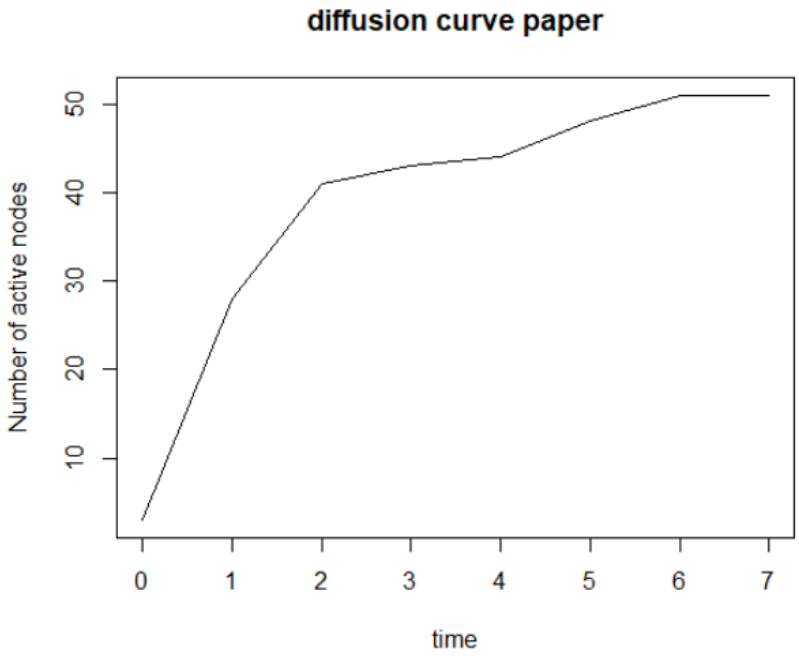




Seeding 3 nodes:







If you can seed only one person, who will you choose? Will you choose the same person to promote these three different things? **[Question 34, 5 point].** (Note that the

thresholds we assigned here are independent from the probability in the IC model, that is partly why you might find the seeds are very different from two models.)

Youtube

nodes	spread
68	100

Vegetarian recipe

nodes	spread
15	69

paper

nodes	spread
69	71

The linear threshold model greedy algorithm shows us that to spread a youtube video you would seed either node 68, 1 or 2 where the spread is 100. To spread a vegetarian recipe you would seed node 15, with a spread of 69. To spread a paper you would seed node 69. Evidently, you will not choose the same person to promote these three different things.

Again, you now have a little more time or budget to seed 3 people. Using greedy algorithms and degree heuristics to 1) find out the seeds for Youtube, vegetarian recipe and paper, respectively. 2) the differences of the performance by degree and greedy algorithms. 3) Check out the network attributes (e.g., centrality measures) and threshold of the seeds provided by the greedy algorithms. Do you find some common properties of the seeds? What difficulty will you foresee to implement the theoretical solutions suggested by the greedy algorithm in promoting the vegetarian receipe and the paper? **[Question 35, 5 point].**

Greedy

1) Youtube

nodes1	nodes2	nodes3	spread1	spread2	spread3
68	1	2	100	100	100

2) Vegetarian

nodes1	nodes2	nodes3	spread1	spread2	spread3
15	48	64	69	72	75

3) Paper

nodes1	nodes2	nodes3	spread1	spread2	spread3
69	22	50	71	84	87

node - spread

Youtube			Vegetarian recipe			Paper		
IC	LT	deg_H	IC	LT	deg_H	IC	LT	deg_H

6-71	68 -100	68 - 61	6-83.6	15 -69	68-82	6 - 46	69 - 71	68-35
21 - 60.2	1 - 100	21-72	48 - 83.2	48 - 72	21-71	21 - 43.2	22 - 84	21-44
68 - 58.2	2 - 100	69-69	68 - 60.2	64 - 75	69-71	68 - 46	50 - 87	69-35

Thresholds;

Youtube; 68 - 0.92, 1- 0.0925, 2- 0.0925

Vegetarian; 15- 0.92, 48-0.92, 64-0.92

Paper; 69-0.67, 22-0.92, 50-0.67

The difference between the nodes chosen as seeds by the IC and LT greedy algorithm is caused by the approach. The IC greedy algorithm chooses the seeds based on the popularity of the nodes, the nodes with the biggest the audience (degre*probability of sharing) will be chosen as seeds. The threshold greedy algorithm chooses the seeds based on the centrality & threshold. In case a node has a high centrality and also a high threshold (is harder to convince), the node is chosen as seed.
There is not a lot of overlap between the chosen seeds. Only node 68 & 48 is chosen by both algorithms as seed, but even these nodes only occur only once in the LT model.

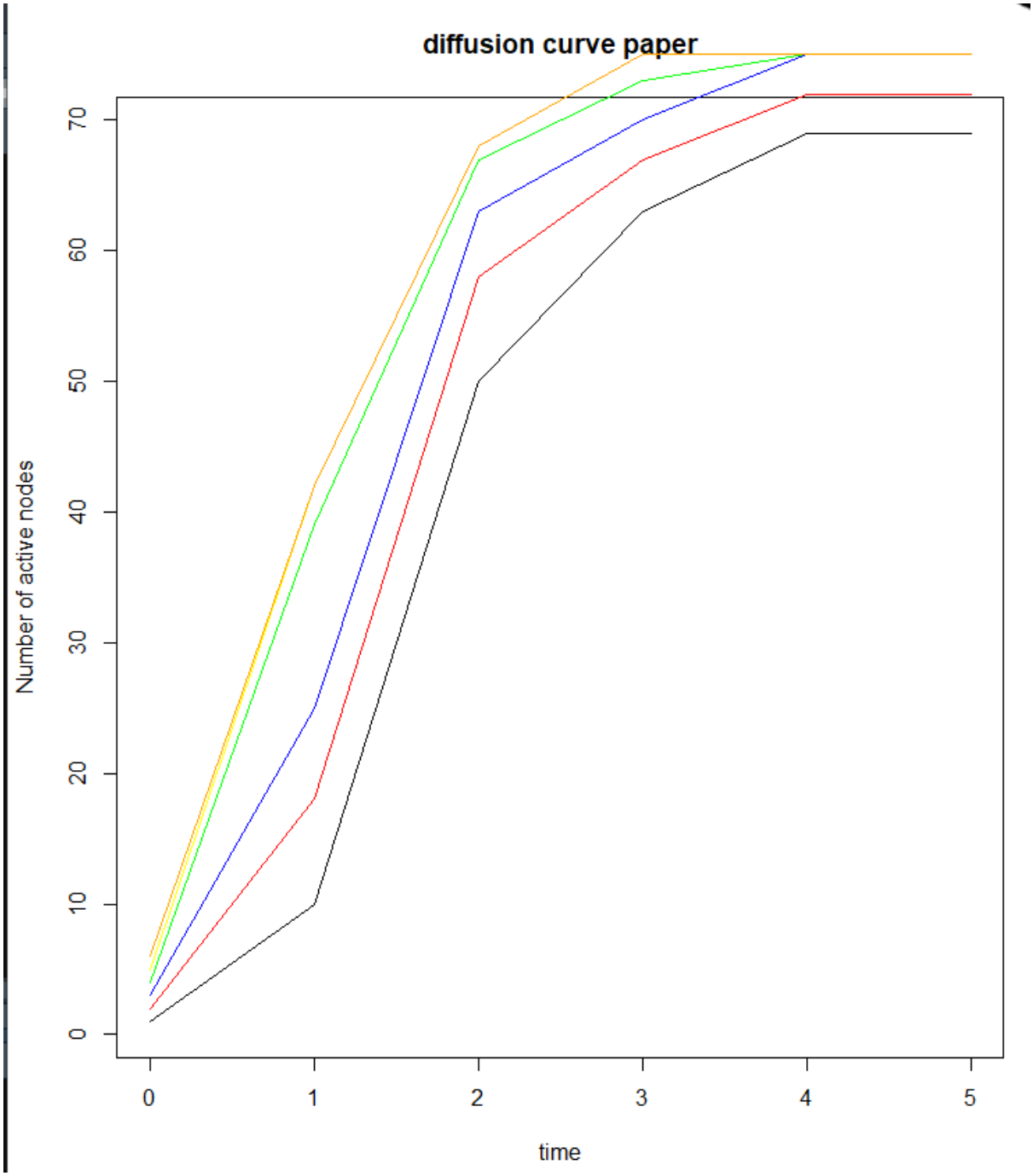
Difficulty you will foresee:

The greedy algorithms provide us with nodes that should ideally be seeded in order to reach the highest diffusion, in order words, the most successful promotion of a vegetarian recipe or a paper. The theoretical solutions for the promotion is aimed at influence maximization. The linear theoretical greedy algorithm suggests to seed the most central node with the highest threshold. In theory this solution works well; if you have already seeded someone with a high threshold (who is also central in the network), other persons in the network no longer need to convince this person about the vegetarian recipe or the paper. While theoretically it is an advantage that you have seeded a person with a high threshold, in practice this presents a problem. The high threshold implies that the person is more difficult to convince. If the threshold of this person is high because this person if not sensitive for group pressure, seeding this person can still be successfully done. However, if this person has a high threshold because of a certain dislike for the product or thing he/she has to be convinced about, this might make seeding this person infeasible. For example, if a person has a high threshold for the diffusion of a vegetarian recipe, because this person is a real meat-lover with strong feelings of dislike for vegetarian meal, it will be very difficult to seed

this person in practice. Similarly, if a person sincerely dislikes the topic/opinion written about in a paper, in real life you might not be able to convince this person to promote the paper.

Image you are now making a budget plan to show the cost-effectiveness of different seed sizes. You need to investigate the return (i.e., increase of activation size) to input (i.e., change of seed size). Try to produce a plot to show cost-effectiveness of increasing seed size. How will you interpret the shape of the curve? And if you target is to achieve 90% adoption rate, at least how many people you should seed? Demonstrate for the cases for vegetarian recipe **[Question 36, 3 point].**

With a seed size between 1 and 3, the curve is S-shaped. When increasing the seed size the curve rises much faster, so diffusion goes very fast. To achieve a 90% adoption rate at least 3 nodes are needed as, according to the diffusion curve, 75% is quite quickly reached while not many nodes are seeded.


diffusion curve paper
Number of active nodes
0
10
20
30
40
50
60
70
0
1
2
3
4
5
time

BONUS QUESTIONS:

In Q30, you studied how the largest component of this network was break down by the nodes suggested by greedy algorithm and the degree and betweenness heuristics. Can you propose an even more efficient algorithm (i.e., break down the largest component with even less percentage of nodes taken out)? **[Question 38, 5 point].**

The most "efficient" algorithm to have the last amount of node loss would be to use a brute force strategy to test out every possible combination of node deletion and then checking what combination the largest reduction in component size gives. For smaller networks this is computationally feasible but for larger networks this will not work.

A alternative which does not involve brute forcing our way through

Using weight of nodes you have 3 distinctions

1: the central nodes which receive many links

2: which links it receive, what the weight is behind those nodes

3: the centrality of the linkers

So using the approach to take out the person which reduces the biggest component size. At the same time the person which has a high weight attached to them when they spread information to other parts of the network and rate of influencing. Then following the nodes which have bigger weights attached to them (can be done through a pagerank centrality index algorithm) and then testing out nodes which have lower amount of connections but a high amount of betweenness (to reduce resources used).