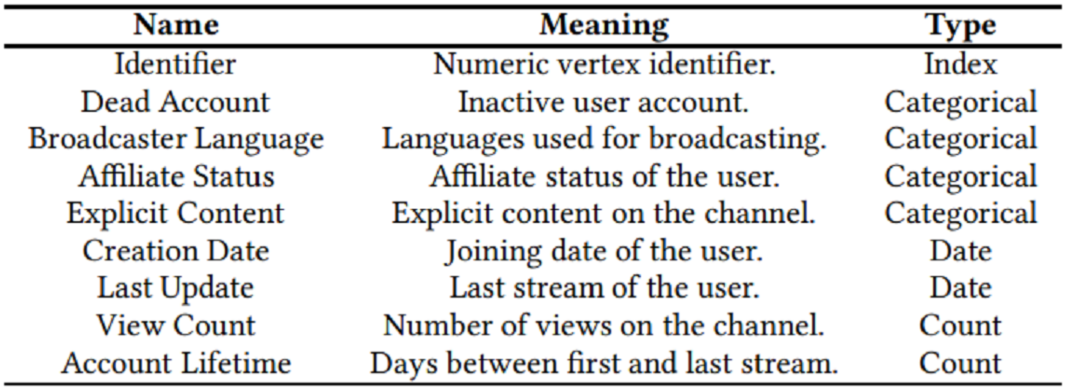
Background and Data Description

In the project proposal, we analyzed the features of this dataset, understood the meaning behind numbers, and came up with a series of questions that we wanted to explore with a thorough plan on how to implement them. There are mainly three parts. First, grab a general idea of the configuration of this data through visualization. Second, find some relevance between data attributes and the formation of ties, and propose to use the ERGM model to explore their relationships. Also, evaluate how the model fits the data. Lastly, we want to provide some suggestions for the Twitch company based on the model results.

We downloaded this data from ‘https://snap.stanford.edu/data/twitch_gamers.html’.The data was collected from a popular live streaming website called Twitch. Like youtube, it shows various items such as gaming, music, movies, and so on. The original data is composed of 2 ‘CSV’ files, one is ‘twitch_feature.csv’ which has more than 100,000 rows and 9 columns. Each row represents a streamer(node) and each column represents a feature of this streamer, as shown in the bellowing figure. The other one is ‘edge.csv’ which has more than 6000,000 rows and 2 columns, each row represents an undirected tie between two nodes, in our case, it means if two streamers have followed each other.



Name	Meaning	Type
Identifier	Numeric vertex identifier.	Index
Dead Account	Inactive user account.	Categorical
Broadcaster Language	Languages used for broadcasting.	Categorical
Affiliate Status	Affiliate status of the user.	Categorical
Explicit Content	Explicit content on the channel.	Categorical
Creation Date	Joining date of the user.	Date
Last Update	Last stream of the user.	Date
View Count	Number of views on the channel.	Count
Account Lifetime	Days between first and last stream.	Count

In our project, concerning the questions we want to answer, only 4 attributes were used. The first one is ‘affiliate status’, which is a partnership run by Twitch between streamers and the platform. Joining affiliate status will boost streams’ profits and maximize their influence on the platform. A streamer being an affiliate means the close connection and loyalty of the platform. The second variable is ‘explicit content’, which means the account shares mature information or video on the website ever since the account has been created. The ‘view count’ indicates the number of their fans. The last attribute, ‘account lifetime’, means the time period from the date of registration to the latest date of login.

Pros and cons of Twitch data set:

Pros:

1. This data set has collected information of steamers who created accounts between 2007 and 2018, so it contains substantial data for analysis.

Cons:

1. The feature information of each streamer is quite limited, except for the identifier only 8 meaningful features. For instance, there is only one variable ‘explicit content’ (referred as ‘mature’ in the rest of this report) to describe the content, it’s either explicit or not, which can be a coarse division and possibly leads to a bad model.
2. Some descriptions of attributes are not precise, which can be misleading for later analysis. For instance, the description of ‘view count’ is the number of views on the channel, we can’t tell if this is a cumulated view count, or only count the number of views when it’s live.

The in-depth questions you are going to answer

The questions we were attempting to answer:

1. Streamers who have a higher ‘lifetime’ are more likely to connect with other streamers.
2. Streamers with higher ‘views’ are more likely to connect with other streamers.
3. Streamers who spread mature content are more likely to connect with other streamers who also broadcast mature content.
4. Streamers with affiliates are more likely to connect with other streamers who also have affiliates.

Hence, we made four questions corresponding to hypotheses.

1. Are streamers who have a higher ‘lifetime’ more likely to have connections with other streamers?
2. Are streamers with higher ‘views’ more likely to have connections with other streamers?
3. Are streamers who post mature content more likely to follow each other?
4. Are streamers who have affiliate status more likely to have connections with each other?

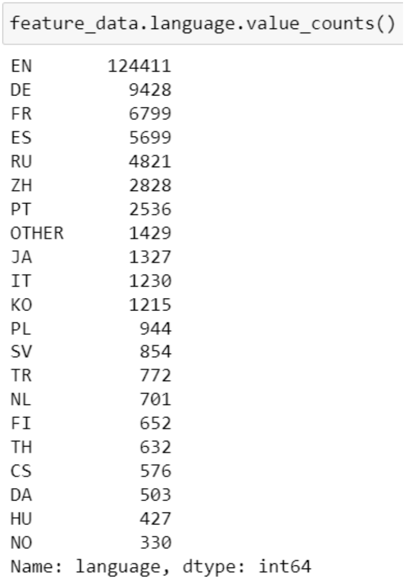
We are trying to learn which attributes of a streamer can make him/her more likely to establish connections with other streamers. And the answers to these questions would be very beneficial for the organization because they could provide Twitch with significant insights into its future development. The four questions point to the four critical features of the platform. Taking advantage of current data collected from Twitch users helps the Twitch executives make decisions on how to improve the platform for the sake of profitability and the sustainable development of the organization.

Preprocessing and Analysis

Initially, we imported our twitch data, both edge list and features files, into “Rstudio” and tried to visualize the whole network and grab a sense of its structure. The first step read in the data as a network matrix was successful but when we tried to convert the edge list network into an “igraph” object, the “Rstudio” instantly gave an error message. After some online research, we realized that everything that goes into the converter needs to have indices starting with 1. Hence, we preprocessed the edge.csv file to have indexes starting from 1 and their corresponding feature sets synchronically so that each pair of nodes can be matched in the feature lists.

However, this was insufficient, and it was still unable to execute properly, due to the exceeding size of the given datasets, which consist of over 6 million edges and 100 thousand nodes, so it was still impossible to convert a network matrix into a graph object for visualization. We later decided to randomly sample the data to a reasonable size. More specifically, we ended up using 0.5% of the original data. We preprocess the edge lists and. Yet, this approach has a huge drawback of deconstructing structures and relations of the original graph, and the manner of using random sampling will also give unpredicted and unrelated results during the examinations.

Therefore, we filtered out the datasets by only looking at a specific language that users speak in the twitch profile. After checking all the value counts for different languages(as can be seen in the below figure) “NO " (Norwegian) is selected as our study object, which has about 300 nodes and 1000 relationships.



```
feature_data.language.value_counts()
```

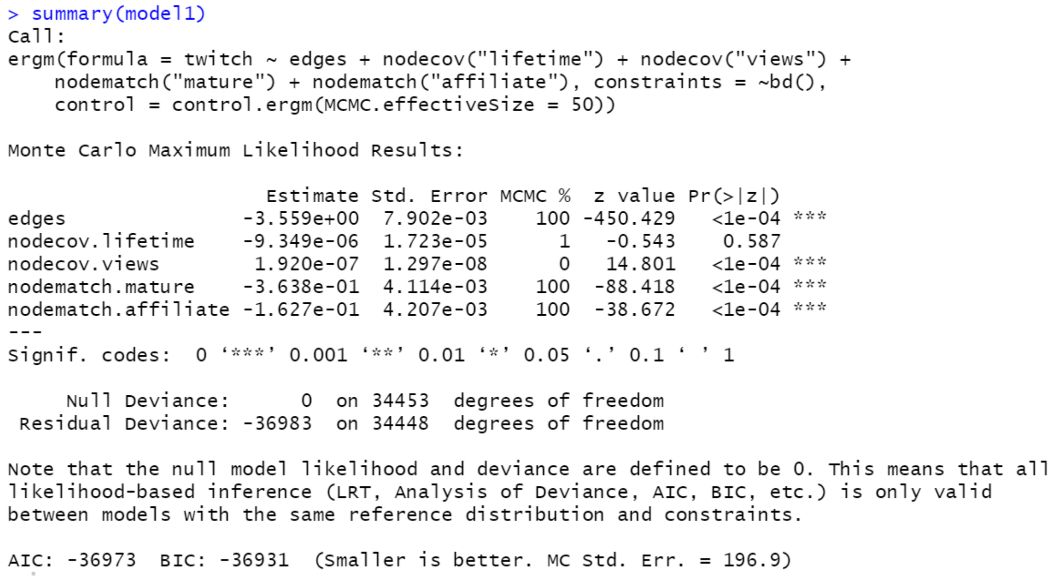
```
EN       124411
DE         9428
FR         6799
ES         5699
RU         4821
ZH         2828
PT         2536
OTHER      1429
JA         1327
IT         1230
KO         1215
PL          944
SV          854
TR          772
NL          701
FI          652
TH          632
CS          576
DA          503
HU          427
NO          330
Name: language, dtype: int64
```

According to our questions, which mostly focus on exploring the formation of ties among nodes, we employed the ERGM model as the main method to analyze this data set, not only because ERGM can be used to conduct regression-like analyses but also because it can allow us to include interactions between individual attributes at the dyadic level, as well as edge attributes and predictor networks. Here, we chose nodecov("lifetime"), nodecov("views"), nodematch("mature"), nodematch("affiliate") as the model input variables. Each of them corresponds to a hypothesis we proposed before. 'Nodecov' measures the main effect of a covariate, and this term adds a single network statistic for each quantitative attribute or matrix column to the model equaling the sum of attr(i) and attr(j) for all edges (i,j) in the network, so the first two hypotheses can be validated by inputting nodecov("lifetime"), nodecov("views") into the ERGM model and analyzing the output model statistics. The last two hypotheses, which are exploring homophily among attributes of nodes, so 'nodematch' was used to measure this effect since this term adds one network statistic to the model, which counts the number of edges (i,j) for which attr(i)==attr(j). For the 3rd and 4th hypotheses, we can validate them by using nodematch('mature') and nodematch('affiliate') as model-independent variables and then follow the rule of hypothesis testing to evaluate the statistical significance of each variable.

Also, we employed three methods to evaluate the model performance from different angles. First, Markov chain Monte Carlo diagnostics(MCMC diagnostics) was used to create simple diagnostic plots for MCMC sampled statistics produced from the above model. Second, the ERGM model was used to simulate a series of null models to check how well the estimated model captures certain features of the observed network. Here we chose to extract the triangle data from the simulated networks, plotted the triangle distribution of these simulated networks as a histogram, and added an arrow to the histogram to show the position of the observed network. Third, the Goodness of fit test was used to see how well each model statistics fit.

However, after defining the framework of our experiment, the first attempt was a failure, which produced a model that can't converge. After doing a lot of online research and consulting with TA, we realized there is still a problem with the input data. Even though the size and number of ties among streamers who speak Norwegian are reasonable, there are too many isolates, which may derogate the performance of ERGM seriously. So we processed the data again to delete isolates and only keep the largest component of the network, then finally got the expected results.

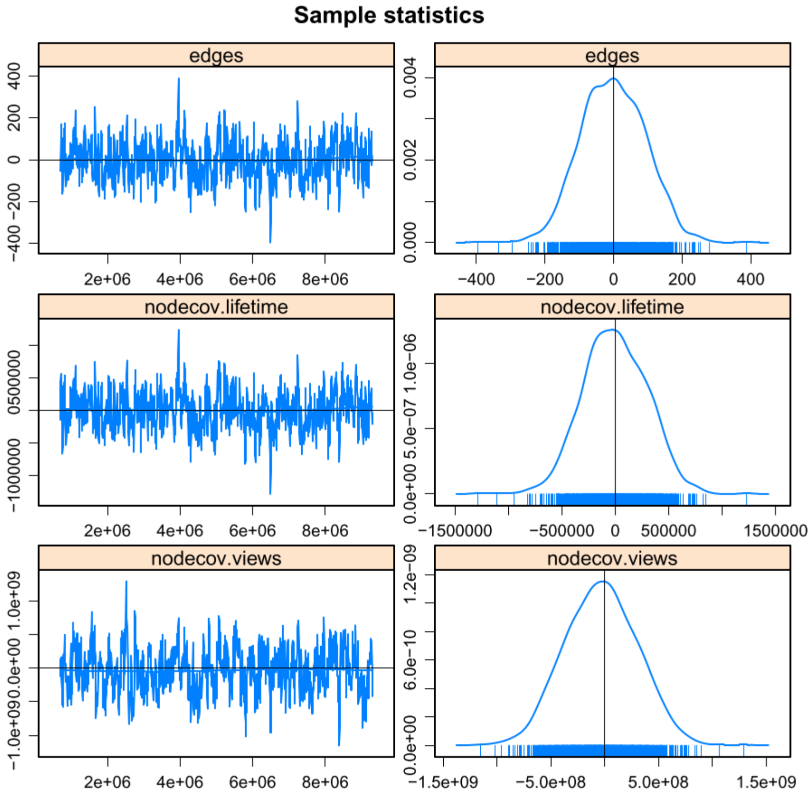
Findings

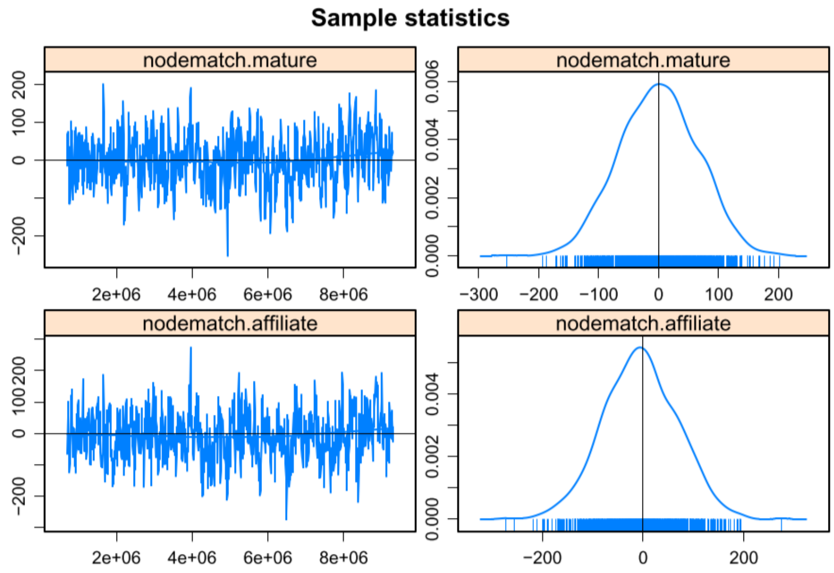


```
> summary(model1)
Call:
ergm(formula = twitch ~ edges + nodecov("lifetime") + nodecov("views") + 
    nodematch("mature") + nodematch("affiliate"), constraints = ~bd(), 
    control = control.ergm(MCMC.effectiveSize = 50))

Monte Carlo Maximum Likelihood Results:

                      Estimate Std. Error MCMC %  z value Pr(>|z|)    
edges               -3.559e+00  7.902e-03    100 -450.429   <1e-04 ***
nodecov.lifetime    -9.349e-06  1.723e-05      1   -0.543    0.587    
nodecov.views        1.920e-07  1.297e-08      0   14.801   <1e-04 ***
nodematch.mature    -3.638e-01  4.114e-03    100  -88.418   <1e-04 ***
nodematch.affiliate -1.627e-01  4.207e-03    100  -38.672   <1e-04 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

     Null Deviance:      0  on 34453  degrees of freedom
 Residual Deviance: -36983  on 34448  degrees of freedom
 
Note that the null model likelihood and deviance are defined to be 0. This means that all
likelihood-based inference (LRT, Analysis of Deviance, AIC, BIC, etc.) is only valid
between models with the same reference distribution and constraints.

AIC: -36973  BIC: -36931  (Smaller is better. MC Std. Err. = 196.9)
```

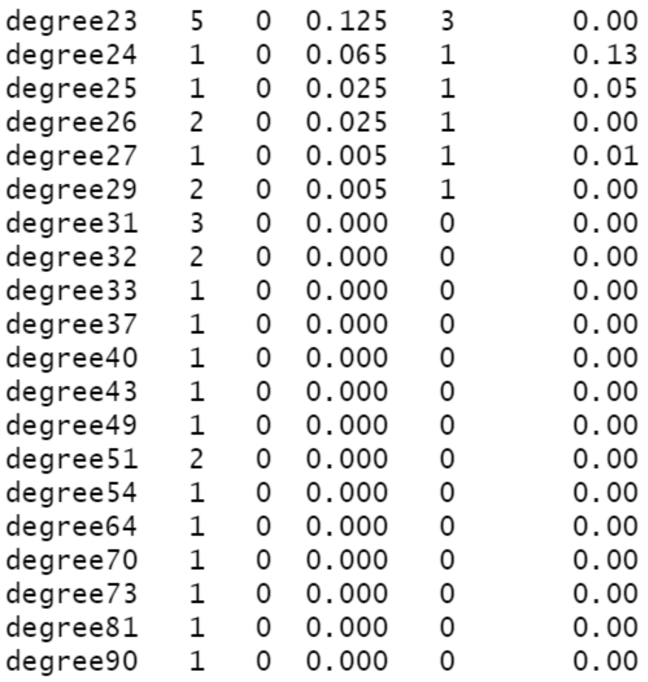
The picture above is a summary of the results generated from our model. The estimate of Nodecov.lifetime is negative, indicating a negative relationship between the streamers' “lifetime” and the connections they make. However, the p-value of this variable is 0.587, which is too large to prove its statistical significance. The p-values of the other three variables are small enough to show that the results are statistically significant. Nodecov.views examines the number of links created by streamers with different views. The odd ratio is about 1, meaning that the probability of streamers with higher views making connections with other streamers is about the same as the probability of streamers with relatively low views making connections with other streamers. Nodematch.mature measures the links between streamers who generate and do not generate explicit content. The negative estimate suggests that explicit content streamers tend to not make connections with each other. Specifically, the odds ratio of 0.69503 means that Streamers who post mature content are 0.7 times more likely to make connections with other streamers who generate mature content than those who do not make mature content. Finally, the estimate of nodematch.affiliate also points to a negative relationship between the streamers' affiliate status and the connections they make with each other. The odd ratio of 0.849846 suggests that the probability of steamers who have affiliate status making connections with each other is about 0.85 times the probability of streamers who do not have affiliate status making connections with each other.

To evaluate how good our model fits, we have utilized three different methods, two of which suggest that our model is a good one. The MCMC process converges to the desired state and displays a stable trend. The distributions also resemble normal distributions. In the test for the goodness-of-fit, the P-values are also close to 1, which is a good sign for our model. Nevertheless, the model might not be a very precise one in terms of triangle measure. As we can see from the histogram, there is a substantial distance between the observed network and the simulated network. Therefore, there is still huge space for our model to improve.

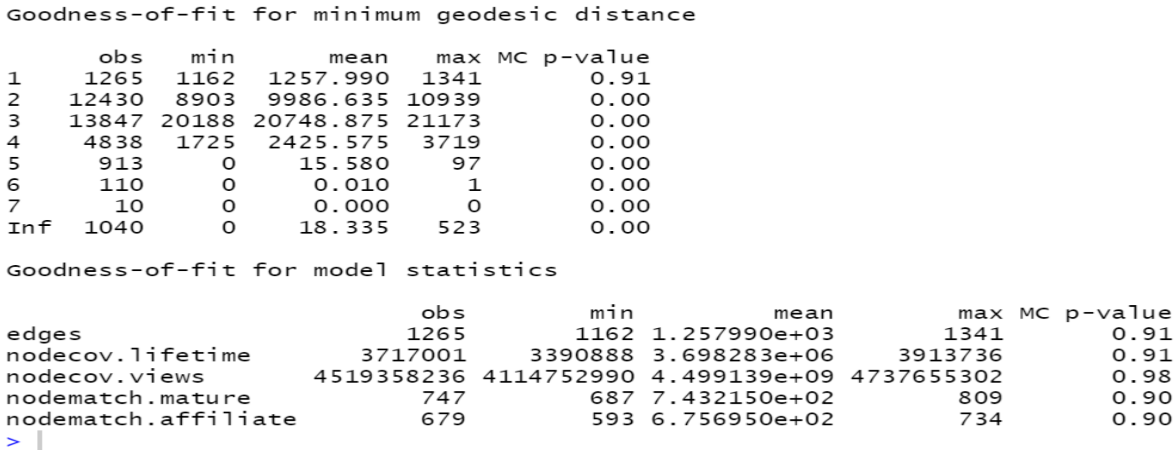




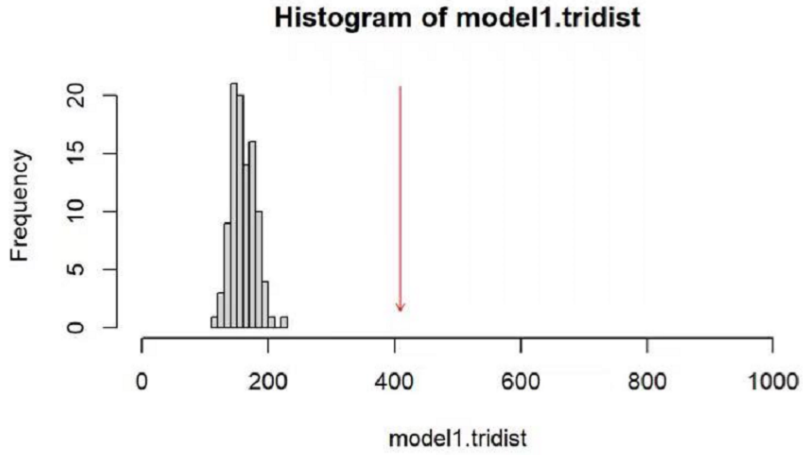
```
Goodness-of-fit for degree

          obs min   mean max MC p-value
degree0     0   0  0.070   2       1.00
degree1    54   0  0.605   3       0.00
degree2    31   0  2.040   7       0.00
degree3    28   1  5.075  11       0.00
degree4    19   2 10.220  20       0.01
degree5    14   7 16.010  28       0.79
degree6    17  12 22.365  31       0.19
degree7     5  13 26.035  42       0.00
degree8     5  18 28.885  44       0.00
degree9    10  17 27.900  39       0.00
degree10    3  15 25.925  46       0.00
degree11    8  13 23.400  36       0.00
degree12   10  10 19.375  30       0.01
degree13    5   7 15.360  26       0.00
degree14    4   3 12.445  23       0.02
degree15    3   2  9.805  20       0.04
degree16    5   1  6.235  13       0.78
degree17    2   0  4.235   9       0.43
degree18    2   0  3.010   8       0.84
degree19    1   0  1.840   6       0.90
degree20    0   0  0.990   5       0.79
degree21    4   0  0.645   3       0.00
degree22    3   0  0.280   2       0.00
```



```
degree23   5   0  0.125   3        0.00
degree24   1   0  0.065   1        0.13
degree25   1   0  0.025   1        0.05
degree26   2   0  0.025   1        0.00
degree27   1   0  0.005   1        0.01
degree29   2   0  0.005   1        0.00
degree31   3   0  0.000   0        0.00
degree32   2   0  0.000   0        0.00
degree33   1   0  0.000   0        0.00
degree37   1   0  0.000   0        0.00
degree40   1   0  0.000   0        0.00
degree43   1   0  0.000   0        0.00
degree49   1   0  0.000   0        0.00
degree51   2   0  0.000   0        0.00
degree54   1   0  0.000   0        0.00
degree64   1   0  0.000   0        0.00
degree70   1   0  0.000   0        0.00
degree73   1   0  0.000   0        0.00
degree81   1   0  0.000   0        0.00
degree90   1   0  0.000   0        0.00
```



```
Goodness-of-fit for minimum geodesic distance

      obs   min      mean   max MC p-value
1    1265  1162  1257.990  1341       0.91
2   12430  8903  9986.635 10939       0.00
3   13847 20188 20748.875 21173       0.00
4    4838  1725  2425.575  3719       0.00
5     913     0    15.580    97       0.00
6     110     0     0.010     1       0.00
7      10     0     0.000     0       0.00
Inf  1040     0    18.335   523       0.00

Goodness-of-fit for model statistics

                             obs        min         mean        max MC p-value
edges                       1265       1162 1.257990e+03       1341       0.91
nodecov.lifetime         3717001    3390888 3.698283e+06    3913736       0.91
nodecov.views         4519358236 4114752990 4.499139e+09 4737655302       0.98
nodematch.mature             747        687 7.432150e+02        809       0.90
nodematch.affiliate          679        593 6.756950e+02        734       0.90
> 
```



Implications

Inferring from the findings, we've further drawn some insights about Twitch's current and future situations in terms of its target consumers, profitability, and community building. We propose three recommendations under these three aspects.

First of all, as we can interpret the links generated by actors as the users' interactions, the finding of our model shows that there is no significant discrepancy between the amount of interaction created by old and new customers of Twitch. In other words, old and new users are equally important. If the organization aims to encourage the interactions between Twitch users, it does not have to deliberately retain its old customers. One reasonable speculation is that old customers may hope the platform to keep its old features and traditions to a very large extent. Old users are used to the old interface and functions of the website and may even have attached their affections and sentiments to the old versions. Unfortunately, trying so hard to retain old users could potentially limit the platform's future development. Our practical advice is that Twitch still needs to mainly focus on innovation and attracting more new customers for the sake of not being left behind in this rapidly changing market.

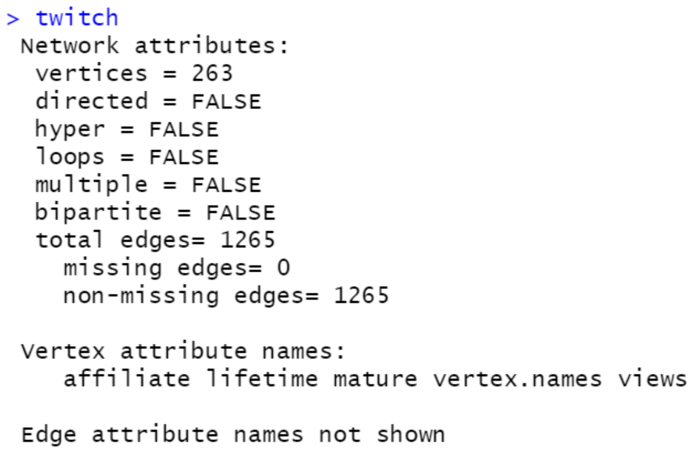
Secondly, we examine the important issue of an organization's profitability. Twitch's profitability is closely related to the number of views that the platform receives. However, we do not observe a correlation between users' interaction and the views of their content. That is to say, the number of views that the platform receives has nothing to do with how active the users are in terms of their interactions with each other. To be more specific, for the goal of pursuing higher views for the website in general, Twitch should put more emphasis on encouraging streamers to generate content with higher quality instead of putting much effort into stimulating users' interaction.

Our third piece of advice regards Twitch's community building. Currently, streamers who generate explicit content do not make connections with each other, according to our findings from the model. We believe that it would be a better option if Twitch could build a community solely for the display and communication of explicit content. Dividing explicit and non-explicit content on the platform ensures a regular and healthy operation of the website.

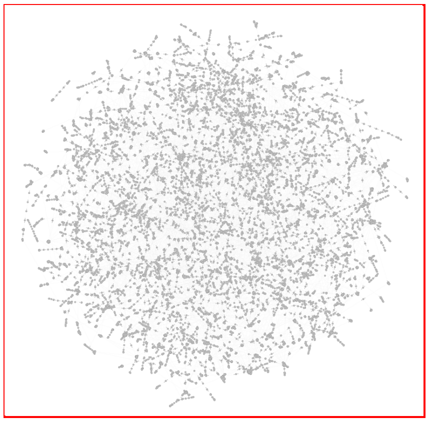
Reflection

The ERGM model we constructed can help us address three of the four questions we raised. We fail to test if the streamers who have a higher 'lifetime' are more likely to have connections with other streamers because the p-value is too small and thus we do not have sufficient evidence to support our hypothesis. The results for the other three questions are also out of our expectations. We hypothesize that streamers with higher views are more likely to make connections with other streamers. However, the actual finding is counterintuitive, which indicates streamers who have higher views make as many interactions as streamers with lower views. Moreover, while we conjecture that streamers who generate mature content are more likely to make connections with each other, and streamers who have affiliate status are more likely to make connections with each other, our findings prove the opposite of these two assumptions to be true. Although the results surprise us, it also brings us exceptionally fruitful information. We can thus give meaningful advice to the organization since the executives would be very likely to think otherwise without the support from our data.

Visualization



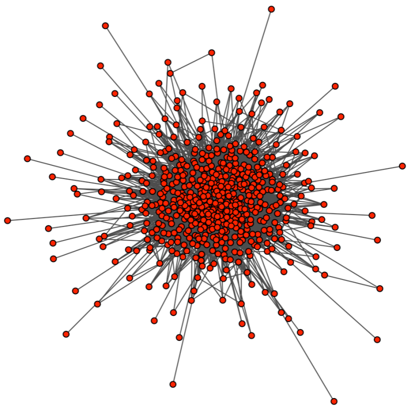
```
> twitch
 Network attributes:
  vertices = 263
  directed = FALSE
  hyper = FALSE
  loops = FALSE
  multiple = FALSE
  bipartite = FALSE
  total edges= 1265
    missing edges= 0
    non-missing edges= 1265

 Vertex attribute names:
    affiliate lifetime mature vertex.names views

 Edge attribute names not shown
```



The graph above is the minimized dataset(0.5%), which is hard to identify any feature or structure from it. Also, the graph is too big to apply to ‘igraph’ objects since deconstructing the relations of the original graph and the manner of using random sampling will give unpredicted and unrelated results during our examinations.

Finally, we decide to separate the datasets by only looking at a specific language that users speak in the twitch profile. After checking all the counts for different languages. “NO'' (Norwegian language) is selected as our study object, which specifically has about 300 nodes and 1000 ties in the network.

Therefore we have the graph below:



As we can see, the graph here has no isolated points since we have preprocessed the dataset to filter out the isolated points, which we will have a more accurate model to evaluate.

These are the attributes we have in the network:

- Affiliate: Affiliate status of the node
- Lifetime: Number of ties between views and life_time
- Mature: Whether have explicit content on the channel of the node
- Views: Number of views on the channel of the node
- Vertices: number of nodes we have in the network.

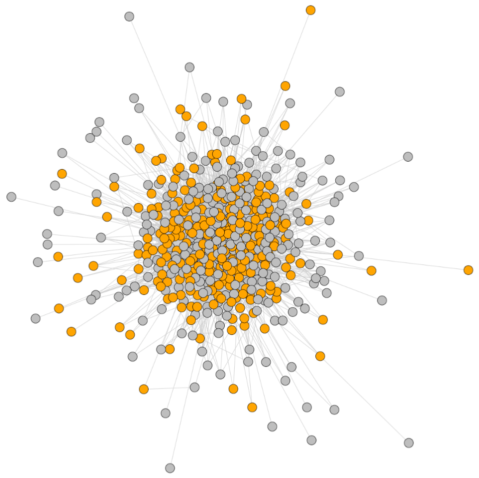
We plotted the separate network with node coloring based on mature attribute:

Plot setting:

Here we choose to set vertex size and arrow size to make them easier to see and observe the connections between nodes.

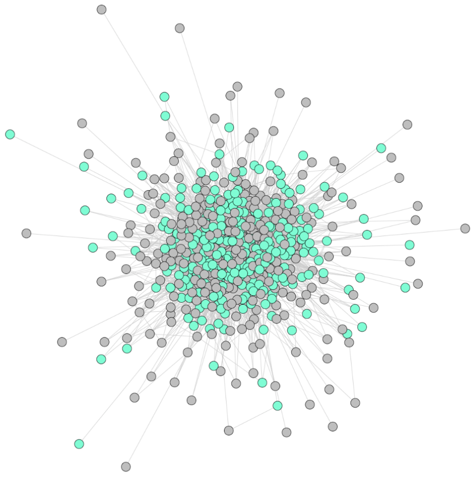
```
igraph_options(vertex.size = 4, vertex.color = 'black', edge.color='gray80', edge.arrow.size=.7,
vertex.label = NA)
```

```
twitch_igraph <- set_vertex_attr(twitch_igraph,"mature",value = read.csv("mature.csv")$mature)
V(twitch_igraph)$color = ifelse (V(twitch_igraph)$mature ==1, " orange ", "grey")
plot(twitch_igraph, layout=layout_with_drl(.), edge.color='black', vertex.color =
V(twitch_igraph)$color)
```



Observation with mature = 1

```
twitch_igraph <- set_vertex_attr(twitch_igraph,"affiliate",value = read.csv("affiliate.csv")$mature)
V(twitch_igraph)$color = ifelse (V(twitch_igraph)$mature ==1, "aquamarine", "grey")
plot(twitch_igraph, layout=layout_with_drl(.), edge.color='black', vertex.color =
V(twitch_igraph)$color)
```



Observation with affiliate = 1

In the above 2 charts, we have clearly observed the nodes that have affiliate = 1 and mature = 1, thus we could know the whole structure with one single attribute and try to come up with the hypothesis we want to know.