

## Анализ персональной сети Facebook в среде Gephi

---

1. Запустите Gephi
2. Загрузите сохранённый файл с данным персональной сети Facebook в среду.  
File -> Open, выберите файл, нажмите Open  
В диалоге параметров загрузки заметьте количество вершин и рёбер, оставьте параметры по умолчанию (undirected) и запустите загрузку.
3. Выберите метод визуализации сети.  
Сверху выберите раздел Overview.  
В Layout выберите метод ForceAtlas 2 и нажмите Run для наилучшего отображения крупных сообществ визуально.
4. Проанализируйте сеть на сообщества.  
Справа в Statistics запустите (Run) алгоритм Modularity (разбиение графа на сообщества по модульности).  
При этом с вершинами связались данные о принадлежности к некоторому классу/сообществу. Выберите сверху раздел Data Library. Последним столбцом в таблице будет Modularity Class, сформированный в результате работы алгоритма. Вернитесь в раздел Overview.  
Раскрасьте сеть по сообществам. Выберите слева сверху Partition, Nodes, Color, выберите параметр раскраски Modularity Class, нажмите Apply. Сообщества раскрашены. Заметьте основные сообщества сети.
5. Проанализируйте важных участников сети (хабы).  
В Statistics запустите алгоритм Avg. Path Length. При этом рассчитываются несколько характеристик центральности вершин и данные о них добавляются к вершинам (три новых столбца в Data Library).
6. Ранжируйте размер вершин на основе центральности по посредничеству. Выберите слева сверху Layout, Nodes, Size/Weight, выберите параметр ранжирования Betweenness Centrality, нажмите Apply. Важные вершины выделены. Заметьте, какие вершины важны в Вашей сети.
7. Для понимания состава сообществ важно видеть, кто им принадлежит. В столбце инструментов функции мыши справа от графа выберите Edit – Edit node attributes. В правом верхнем фрейме выберите Edit. Теперь, выбирая вершины графа мышью, можно отображать в правом верхнем фрейме их атрибуты, включая фамилии и имена людей.

## Анализ сети на языке R

---

1. Запустите среду R

2. Установите библиотеку igraph

```
install.packages("igraph")
```

3. Включите необходимые библиотеки

```
library(igraph)
```

4. Загрузите граф сети, ранее загруженной из Facebook.

```
setwd("C:/Users/nskv/Dropbox/edu/social/sna")  
G = read.graph(file="nskvModel.gml", format="gml")
```

5. Рассчитайте среднюю степень сети

```
d = degree(G)  
sum(d)/length(d)
```

6. Рассчитайте количество кратчайших путей, проходящих через вершины сети

```
b = betweenness(G)  
b
```

```
closeness(G)  
evcent(G)
```

7. Рассчитайте среднюю длину кратчайших путей сети

```
average.path.length(G)
```

8. Обнаружение сообществ

```
comm = cluster_fast_greedy(G)  
sizes(comm)  
membership(comm)
```

9. Отобразите граф

```
layout <- layout_with_graphopt(G)  
layout <- layout_fruchterman_reingold(G)
```

```
plot(G, layout=layout, vertex.size=log(b + 1), vertex.label=d)
```

## Моделирование сетей на языке R

---

1. Установите библиотеку igraph

```
install.packages("igraph")
```

2. Включите необходимые библиотеки

```
library(igraph)
library('ggplot2')
```

3. Смоделируйте случайную сеть

```
n=30
p=2.0/n
m=(n*(n-1)/2)*p
gnp <- erdos.renyi.game(n, p, type="gnp")
plot(gnp, layout=layout.fruchterman.reingold(gnp), main="Erdos-
Renyi Network")
```

4. Варьируя p, посмотрите на формирование гигантского компонента.

```
gcc=c()
cc=c()
probability=c()
for (i in 1:70){
  prob=0.0002*i
  probability=c(probability, prob)
  er <- erdos.renyi.game(500, prob, type="gnp")
  comp <- clusters(er)
  largest <- max(comp$ccsize)
  gcc=c(gcc, largest)
  clust <- transitivity(er, type="global")
  cc=c(cc, clust)
}
plot(probability*500, gcc, pch=19, col=26, xlab="np",
main="Gigantic component")
```

5. Рассмотрите распределение степеней

```
d <- degree(gnp)
summary(d)
hist(d, col=rgb(1,0,0,.5), xlim=c(0,20), xlab="degree",
ylab="freq", main="p=0.2")
```

6. Смоделируйте сеть с предпочтительным присоединением

```
b=barabasi.game(30,power=1,m=3,directed=F)
plot(b,layout=layout.fruchterman.reingold(b),main="Preferencial")
```

7. Рассмотрите распределение степеней

```
d <- degree(b)
```

```
hist(d, breaks=10, col=rgb(1,0,0,.5), xlab="degree", ylab="freq",  
main="n=30, m=3")
```

8. Исследуйте распределение степеней реальной эгоцентрической сети

```
setwd("C:/Users/nskv/Dropbox/edu/social/sna")  
G = read.graph(file="nsvModel.gml", format="gml")  
d <- degree.distribution(G)  
plot(d, log = 'xy', xlab = 'Degree d', ylab = 'p(d)', main = 'Ego  
network degree distribution')
```

## Модели NetLogo

---

1. <http://ccl.northwestern.edu/netlogo/>
2. <http://synthesis.ipi.ac.ru/synthesis/student/BigData/master-social-environments/SocialEnvironmentsModels.zip>