

Statistics and Machine Learning I

Week 11: Social Networks

Coursework 11: Grey's Anatomy sexual interactions

Luis Da Silva

December 16, 2018

1 Introduction

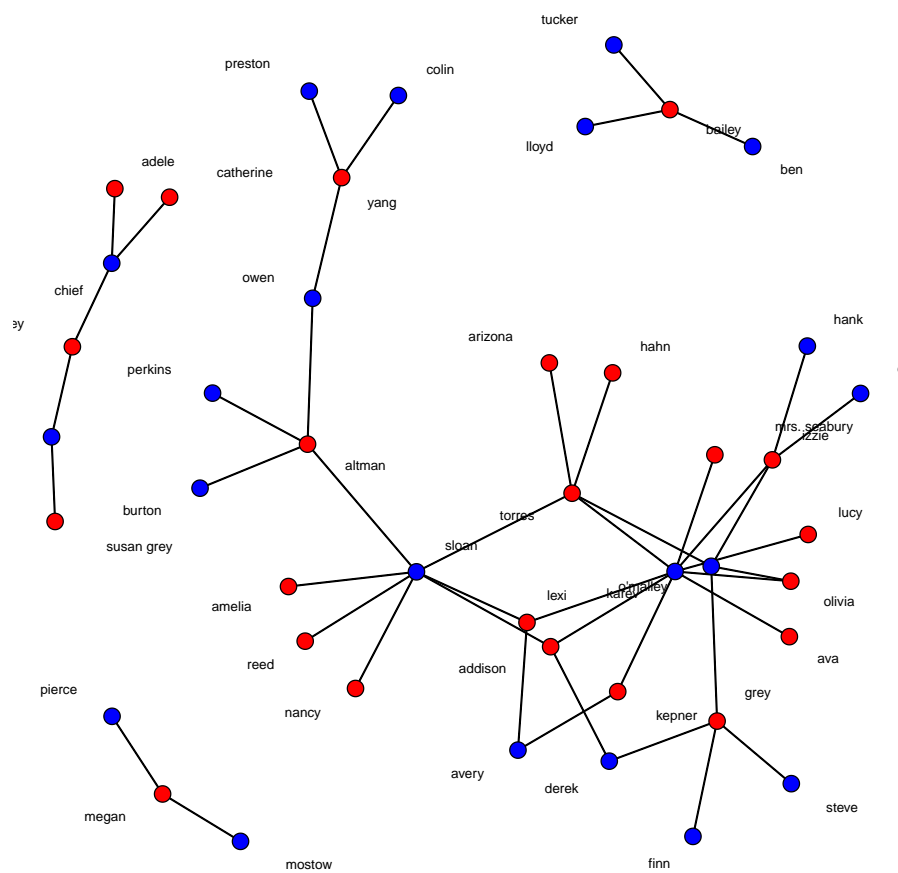
Presented Grey's Anatomy sexual data allows us to see, as a graph model, how people interrelate to each other in terms of sexual interactions in the TV show. We have data for 44 characters and 7 attributes: name, sex, race, birth year, position, season and sign. Real network is presented in figure 1, it shows a relatively sparse network, with no triangles and mostly heterosexual and multiple interactions.

As these networks are quite volatile, in order to generalise conclusions into this network we will use Exponential Random Graph Models (ERGM) to simulate some random graphs based on observed characteristics.

2 Exponential Random Graph Models (ERGM)

ERGM allows us to use Maximum Likelihood Estimation to estimate how selected parameters affect the evolution of our graph and then use Markov Chain Monte Carlo simulation to produce random networks based on this model.

Figure 1: Original network



2.1 Model

To implement them to our data, we first select which terms we want to add to our model. In this model, I selected these 4 terms:

- **Edges:** the number of edges (i.e. connections between vertices¹).
- **nodematch("sex"):** accounts for homophily² in gender, I include this because it is noticeable that heterosexual relations occur more often.
- **nodematch("position"):** accounts for homophily in position. I am presuming that people who get in contact more often (similar position) are more likely to form relations.
- **degree(1):** accounts for monogamy in relationships, making them more likely to appear.

All statistics used are dyad independent (i.e. the probability of a edge to form is independent on each other.)

Described model may be generated in R with the following code:

```
model11 <- ergm(ga.net~edges+nodematch("sex")
               +nodematch("position")+degree(1))
```

2.2 Results

Former model throws the following results:

Monte Carlo MLE Results:

	Estimate	Std. Error	MCMC %	z value	Pr(> z)	
edges	-1.6894	0.2685	0	-6.292	< 1e-04	***
nodematch.sex	-3.2114	0.7153	0	-4.490	< 1e-04	***
nodematch.position	0.9275	0.3211	0	2.888	0.00387	**
degree1	2.0701	0.4472	0	4.629	< 1e-04	***

¹In this example, each vertex is a character

²Tendency to form social ties with similar groups

As these coefficients indicate change in the logarithmic odds, they don't have a linear explanation in terms of probability, so let's illustrate three examples:

- **Same sex on different positions and more than 1 existent interaction:** a tie will have a probability of forming of $\frac{1}{1 + \exp(1.6894 + 3.2114)} = 0.7386\%$
- **Different sex on different positions and more than 1 existent interaction:** a tie will have a probability of forming of $\frac{1}{1 + \exp(1.6894)} = 15.59\%$
- **Different sex on same positions and just 1 existent interaction:** a tie will have a probability of forming of $\frac{1}{1 + \exp(1.6894 - 0.9275 - 2.0701)} = 78.72\%$

2.3 Goodness-of-fit

Before evaluating goodness of fit, we need to simulate some networks, in R, this is:

```
model1.sim <- simulate(model1, nsim=10)
```

Figure 2 shows a sample simulated network. I would say that this actually looks very similar to the original network, but let's perform a goodness-of-fit test to decide if this is an appropriate model for this network. Figure 3 shows these diagnostics for the degrees of the model and statistics used; used model seems to generalize well presented data, and thus could be an appropriate choice to model this network.

3 Full code

```
library(ergm)
```

Figure 2: Sample simulated network

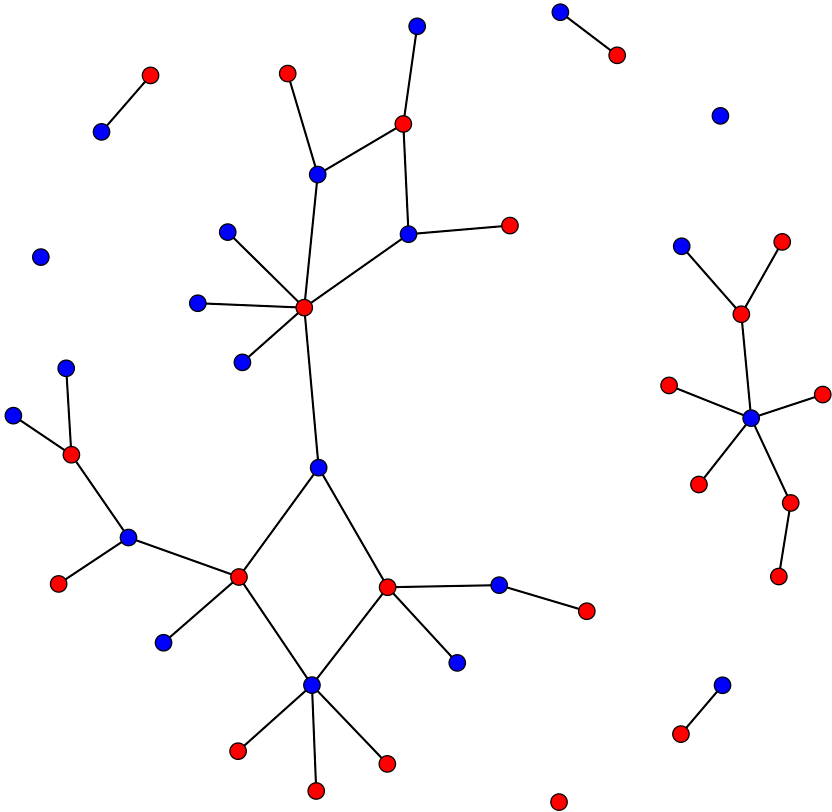
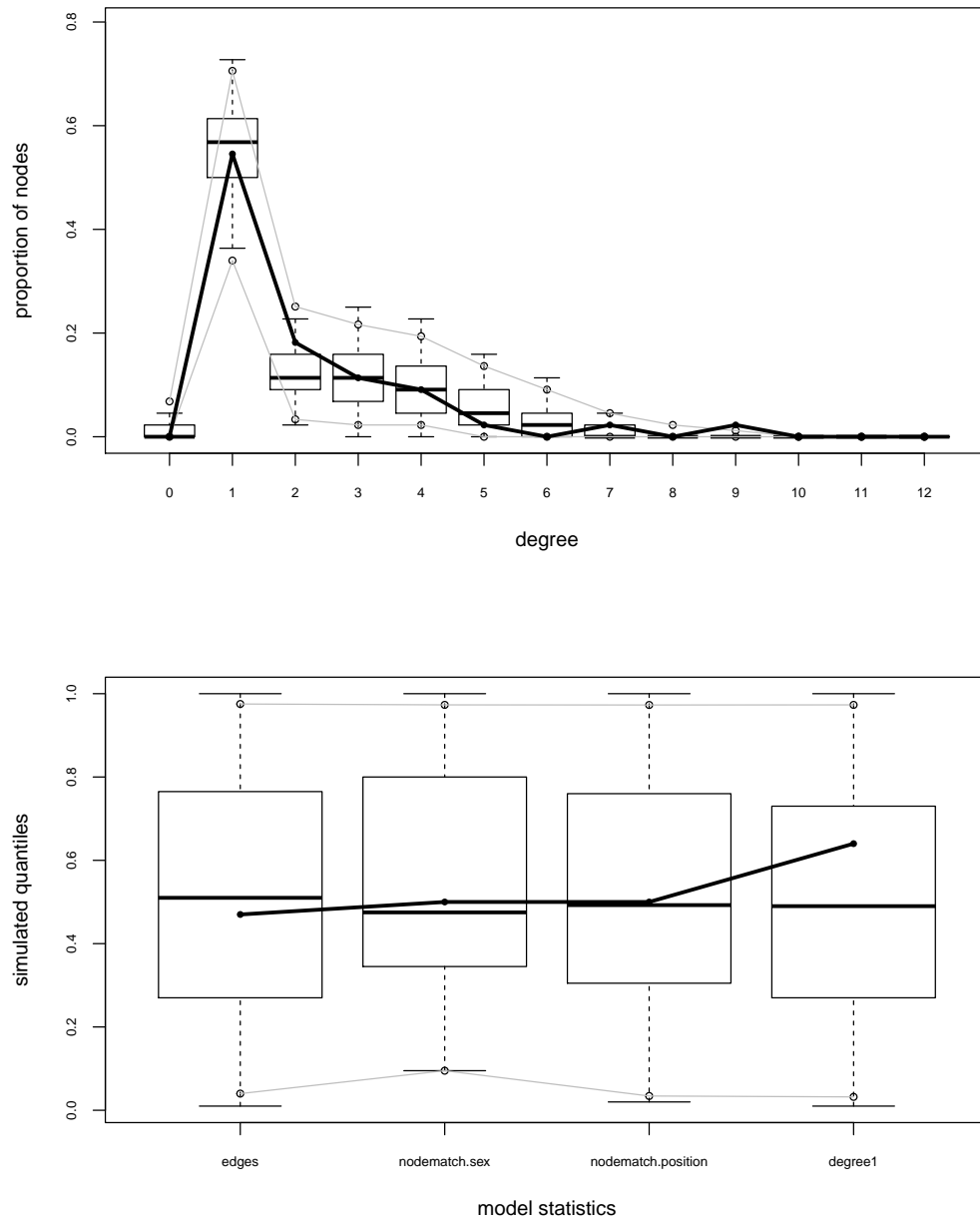


Figure 3: Goodness of fit

Goodness-of-fit diagnostics



```

# Start by reading in the adjacency matrix showing relationships between
ga.mat<-as.matrix(read.table("Grey's_Anatomy_-_sociomat.tsv", sep="\t",
                             header=T, row.names=1, quote="\\"))

#" check it:
ga.mat

# Next import the attribute file
ga.attrs <- read.table("Grey's_Anatomy_-_attributes.tsv", sep="\t",
                      header=T, quote="\\"",
                      stringsAsFactors=F, strip.white=T, as.is=T)
#" check it and familiarise yourself with the attributes available:
ga.attrs
ga.attrs$sex = as.logical(model.matrix(~ sex, data=ga.attrs)[,-1])

# create the network object to use for the coursework tasks
ga.net <- network(ga.mat, vertex.attr=ga.attrs,
                  vertex.attrnames=colnames(ga.attrs),
                  directed=F, hyper=F, loops=F,
                  multiple=F, bipartite=F)

# check it:
ga.net

# Visualise the network, colour nodes based gender and include labels (na
plot(ga.net, vertex.col=c("blue","red")[1+(get.vertex.attribute(ga.net, "
    label=get.vertex.attribute(ga.net, "name"), label.cex=.7)
# label.cex determines the label size

# ERGM
model1 <- ergm(ga.net~edges+nodematch("sex")+
               nodematch("position")+degree(c(1)))
summary(model1)

# simulation----
model1.sim <- simulate(model1,nsim=10)
class(model1.sim)
summary(model1.sim)
plot(model1.sim[[3]],
vertex.col=c("blue","red")[1+
              (get.vertex.attribute(ga.net, "sex")==0)])

```

```
# Goodness of Fit----  
model1.gof <- gof(model1~degree)  
model1.gof  
par(mfrow=c(2,1))    # Separate the plot window into a 2 by 1 orientation  
plot(model1.gof)  
dev.off()
```