lab1_zh2448

Zhengze Hou

2/20/2022

0. prepocess the lazega dadta

```
library(igraph)
```

```
##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

```
#  Import Lazega friend adjacency matrix
lazega_friends_matrix <- as.matrix(read.csv(file.choose(),header=TRUE,row.names=NULL,check.names=FALSE))

lazega_friends_graph <- graph.adjacency(lazega_friends_matrix,mode="undirected",weighted=NULL)
```

```
lazega_attributes <- read.csv(file.choose(), header=TRUE)
vertex_attr(lazega_friends_graph, index=lazega_attributes$ID) <- lazega_attributes

one_direction_edges <- data.frame(get.edgelist(lazega_friends_graph))
#  With names of columns ID1 and ID2
colnames(one_direction_edges)  <- c('ID1', 'ID2')

#  The above data.frame only has one direction for each edge, we need the data fram to contain both
#  e.g. Louis -> Niall and Niall -> Louis, so:
all_edges <- rbind(  # Add a bunch of rows
           one_direction_edges, # to our list of edges
           setNames( # which are contained in a new data frame created by
                one_direction_edges, # copying one_direction_edges
                rev(names(one_direction_edges)) # and changing the column names to be in the reverse or
                # i.e. call ID1 ID2
            )
)

all_edges_with_attributes <- merge(all_edges, #  Merging our exisitng list of edges
      setNames(lazega_attributes,  # With a new data.frame created from `lazega_attributes`
```

```
        paste0(names(lazega_attributes) ,'1')), # by changing the names of its
                                 # columns to append a '1'
    by='ID1') # and we want the elemments of all_edges which have a given ID1 to
    #  be matched with the elements of the new data.frame which have the same value for ID1


#  Pretty much the same thing to add the attributes for the second vertex in each edge
all_edges_with_attributes <- merge(all_edges_with_attributes,  # except that
                   # we're merging with our new edge list, so that we end up
                   # with attributes for the first and second vertices
      setNames(lazega_attributes,
        paste0(names(lazega_attributes) ,'2')),
       by='ID2')

#  Add columns to all_edges_with_attributes which inidcate if the gender of
#  the first vertex in each edge is the same as that of the second vertex
all_edges_with_attributes$homo_gender = ifelse(
       all_edges_with_attributes$gender1==all_edges_with_attributes$gender2,  # if the genders for this
       1, # the value of homo_gender is 1
       0) # otherwise it's 0

# And the same for status
all_edges_with_attributes$homo_status = ifelse(all_edges_with_attributes$status1==all_edges_with_attribu

# Get the proportion of each ego's connections who have the same gender
ego_homophily_stats <- aggregate(all_edges_with_attributes[,c('homo_gender', 'homo_status', 'gender1')]
# remember: the mean of a binary variable is equal to the proportion of cases
# where it's equal to 1

# Merge these data back with the original attributes
lazega_attributes <- merge(lazega_attributes,ego_homophily_stats,  by.x="ID", by.y="ID1")


# Add the degree of each vertex to the attributes
lazega_attributes <- merge(lazega_attributes,  #  Merge lazega_attributes
               data.frame(  # With a new data.frame
                     ID=V(lazega_friends_graph)$ID,  # Where the ID is the ID of each vertex
                     degree= degree(lazega_friends_graph) # and the degee is its degree
                  ),
       by='ID')
```

##Create brokerage data in bz dataframe.

```
library(statnet)
```

```
## Loading required package: tergm

## Loading required package: ergm

## Loading required package: network

##
```

```
## 'network' 1.17.1 (2021-06-12), part of the Statnet Project
## * 'news(package="network")' for changes since last version
## * 'citation("network")' for citation information
## * 'https://statnet.org' for help, support, and other information

##
## Attaching package: 'network'

## The following objects are masked from 'package:igraph':
##
##     %c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices,
##     get.edge.attribute, get.edges, get.vertex.attribute, is.bipartite,
##     is.directed, list.edge.attributes, list.vertex.attributes,
##     set.edge.attribute, set.vertex.attribute

##
## 'ergm' 4.1.2 (2021-07-26), part of the Statnet Project
## * 'news(package="ergm")' for changes since last version
## * 'citation("ergm")' for citation information
## * 'https://statnet.org' for help, support, and other information

## 'ergm' 4 is a major update that introduces some backwards-incompatible
## changes. Please type 'news(package="ergm")' for a list of major
## changes.

## Loading required package: networkDynamic

##
## 'networkDynamic' 0.11.0 (2021-06-12), part of the Statnet Project
## * 'news(package="networkDynamic")' for changes since last version
## * 'citation("networkDynamic")' for citation information
## * 'https://statnet.org' for help, support, and other information

## Registered S3 method overwritten by 'tergm':
##   method                   from
##   simulate_formula.network ergm

##
## 'tergm' 4.0.2 (2021-07-28), part of the Statnet Project
## * 'news(package="tergm")' for changes since last version
## * 'citation("tergm")' for citation information
## * 'https://statnet.org' for help, support, and other information

##
## Attaching package: 'tergm'

## The following object is masked from 'package:ergm':
##
##     snctrl

## Loading required package: ergm.count
```

```
##
## 'ergm.count' 4.0.2 (2021-06-18), part of the Statnet Project
## * 'news(package="ergm.count")' for changes since last version
## * 'citation("ergm.count")' for citation information
## * 'https://statnet.org' for help, support, and other information

## Loading required package: sna

## Loading required package: statnet.common

##
## Attaching package: 'statnet.common'

## The following object is masked from 'package:ergm':
##
##     snctrl

## The following objects are masked from 'package:base':
##
##     attr, order

## sna: Tools for Social Network Analysis
## Version 2.6 created on 2020-10-5.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##  For citation information, type citation("sna").
##  Type help(package="sna") to get started.

##
## Attaching package: 'sna'

## The following objects are masked from 'package:igraph':
##
##     betweenness, bonpow, closeness, components, degree, dyad.census,
##     evcent, hierarchy, is.connected, neighborhood, triad.census

## Loading required package: tsna

##
## 'statnet' 2019.6 (2019-06-13), part of the Statnet Project
## * 'news(package="statnet")' for changes since last version
## * 'citation("statnet")' for citation information
## * 'https://statnet.org' for help, support, and other information

## unable to reach CRAN
```

```r
## make the statnet sociograph

nrelations<-network(lazega_friends_matrix,directed=TRUE)

## connect attributes too
```

```r
nrelations %v% "ID" <- lazega_attributes$ID
nrelations %v% "status" <- lazega_attributes$status

## run the brokerage command

b=brokerage(nrelations, lazega_attributes$status)

bz=cbind(lazega_attributes, b$z.nli)
```

1. I am interested in which factors may affect a person to be a representative role in his social circle. In particular, I want to suggest that if I'm a person with higher status, I will be more likely to be a representative of my social circle.

2. People with higher status usually are seen as a justification of better quality or ability, whether it is true or not. And people may hope this kind of person to decorate themselves as a symbole of this social circle.

3.(a) My dependent variable is Representative role variable in bz datafame, produced from the lazega file. The variable means the broker mediates an outgoing contact from an in-group member to an out-group member. Two-path structure: A -> A -> B

(b) My independent variable are status and seniority variable in the bz dataframe. status (1=partner; 2=associate) , the partner is higher than associate. And seniority is the member's years with the firm, which is also a symbol of status.

```r
table(bz$status)
```

```
##
##  1  2
## 36 35
```

```r
summary(bz$seniority)
```

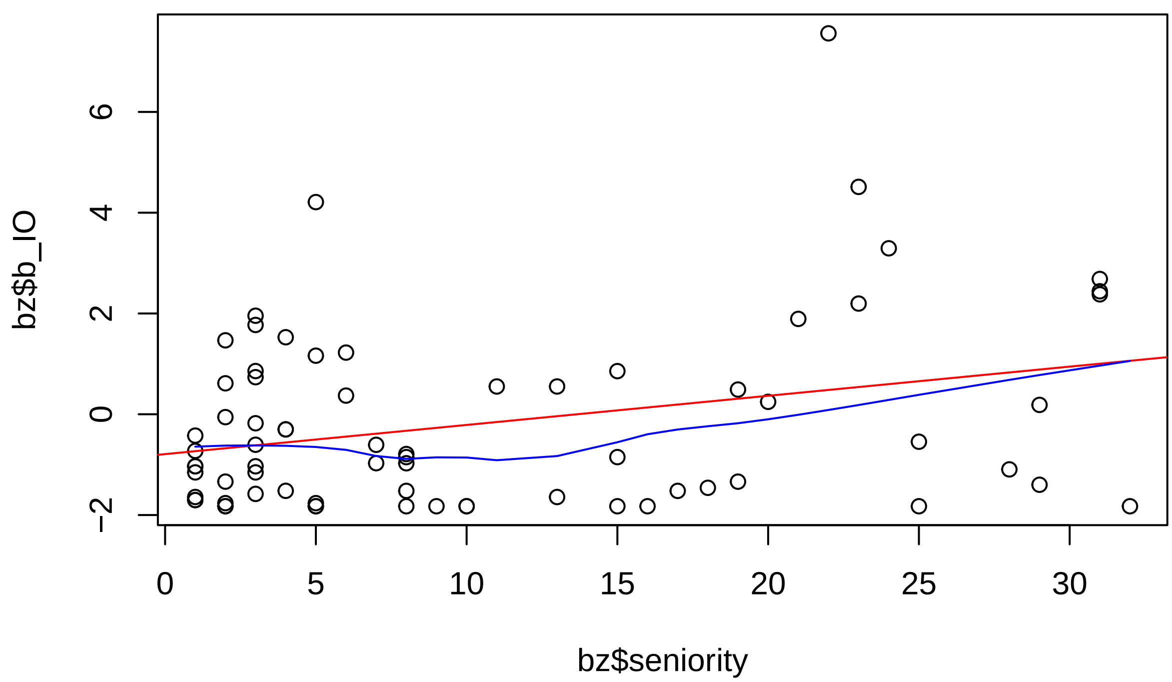
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    3.00    7.00   10.56   17.50   32.00
```

4. Since the status is a binary variable, we don't plot it anymore. In the plot of senioarity, we can dsicover a weak lift of representative role score with the more years with the film, which is also justified in the model 1. However, the status outcomes shows that associates(2) has higher reporesentative role score in average than which of partners(1) in model 1.

A possible explanation is that the partners are more seperated and more care about work. Thus ,they cannot represent for others.

The degree is positive correlated with the reporesentative role score.

```
plot(bz$seniority, bz$b_IO)
abline(lm(bz$b_IO~bz$seniority), col="red")
lines(lowess(bz$seniority,bz$b_IO), col="blue")
```




```
## model 1
summary(lm(b_IO ~ degree + as.factor(status) + seniority, bz))
```

```
##
## Call:
## lm(formula = b_IO ~ degree + as.factor(status) + seniority, data = bz)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -3.7172 -0.7297  0.0282  0.6483  4.9096
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -5.30713    0.72833  -7.287 4.64e-10 ***
## degree               0.13902    0.02055   6.766 3.97e-09 ***
## as.factor(status)2  1.89459    0.50610   3.743 0.000379 ***
## seniority            0.12787    0.02607   4.904 6.28e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.351 on 67 degrees of freedom
## Multiple R-squared:  0.4826, Adjusted R-squared:  0.4594
## F-statistic: 20.83 on 3 and 67 DF,  p-value: 1.208e-09
```

5. Homo_gender is the proportion of each ego's connections who have the same gender. Homo_status is the proportion of each ego's connections who have the same status. In the model 2, homo_status is strongly negatively correlated with the b_IO(the reporesentative role score). IIn the model 3, the added homo_sex doesn't show significant with the b_IO.

```r
## model 2
summary(lm(b_IO ~ degree + as.factor(status) + seniority + homo_status, bz))
```

```
##
## Call:
## lm(formula = b_IO ~ degree + as.factor(status) + seniority +
##     homo_status, data = bz)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9931 -0.7129  0.0064  0.7418  4.1885
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -3.55626    0.85538  -4.158 9.47e-05 ***
## degree               0.15833    0.01998   7.925 3.60e-11 ***
## as.factor(status)2  1.32588    0.50074   2.648   0.0101 *
## seniority            0.11012    0.02485   4.432 3.61e-05 ***
## homo_status         -3.33922    0.99424  -3.359   0.0013 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.258 on 66 degrees of freedom
## Multiple R-squared:  0.5581, Adjusted R-squared:  0.5313
## F-statistic: 20.84 on 4 and 66 DF,  p-value: 3.837e-11
```

```r
## model 3
summary(lm(b_IO ~ degree + as.factor(status) + seniority + homo_status + homo_gender, bz))
```

```
##
## Call:
## lm(formula = b_IO ~ degree + as.factor(status) + seniority +
##     homo_status + homo_gender, data = bz)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0185 -0.6523  0.0894  0.6716  4.2301
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -4.10560    0.95763  -4.287 6.12e-05 ***
## degree              0.15611    0.01997   7.817 6.17e-11 ***
## as.factor(status)2  1.45932    0.50979   2.863  0.00565 **
## seniority           0.11214    0.02479   4.523 2.65e-05 ***
## homo_status        -3.29779    0.99050  -3.329  0.00144 **
## homo_gender         0.75247    0.59973   1.255  0.21409
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.252 on 65 degrees of freedom
## Multiple R-squared:  0.5685, Adjusted R-squared:  0.5354
## F-statistic: 17.13 on 5 and 65 DF,  p-value: 8.931e-11
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::as_data_frame() masks tibble::as_data_frame(), igraph::as_data_frame()
## x purrr::compose()       masks igraph::compose()
## x tidyr::crossing()      masks igraph::crossing()
## x dplyr::filter()        masks stats::filter()
## x dplyr::groups()        masks igraph::groups()
## x dplyr::lag()           masks stats::lag()
## x purrr::simplify()      masks igraph::simplify()
```

```
## model 4
homostatus <- bz %>%
  select(status, homo_status) %>%
  group_by(status) %>%
  summarise(mean = mean(homo_status))
homostatus
```

```
## # A tibble: 2 x 2
##   status  mean
##    <int> <dbl>
## 1      1 0.557
## 2      2 0.446
```

```r
## model 5
homostatus2 <- bz %>%
  select(b_IO, status, homo_status) %>%
  filter(status == 1)
summary(lm(b_IO ~ homo_status, homostatus2))
```

```
##
## Call:
## lm(formula = b_IO ~ homo_status, data = homostatus2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6981 -1.5514 -0.1226  1.4289  6.2718
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.710      1.371   3.437  0.00157 **
## homo_status   -8.442      2.395  -3.524  0.00124 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.869 on 34 degrees of freedom
## Multiple R-squared:  0.2676, Adjusted R-squared:  0.246
## F-statistic: 12.42 on 1 and 34 DF,  p-value: 0.001235
```

```r
homostatus3 <- bz %>%
  select(b_IO, status, homo_status) %>%
  filter(status == 2)
summary(lm(b_IO ~ homo_status, homostatus3))
```

```
##
## Call:
## lm(formula = b_IO ~ homo_status, data = homostatus3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6796 -1.1114 -0.4972  1.1610  4.5341
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.9007     0.6305  -1.429    0.163
## homo_status   1.1888     1.3010   0.914    0.367
##
## Residual standard error: 1.454 on 33 degrees of freedom
## Multiple R-squared:  0.02467,    Adjusted R-squared:  -0.00488
## F-statistic: 0.8349 on 1 and 33 DF,  p-value: 0.3675
```

The explanation above mentioned is supplemented that the partners are more likely to have friends with each other, and they don't tend to be convinced of or compelled to. As a result, the partners in average has fewer b_IO. The model 4 and 5 has justified this indication.

Firstly, the mean of homo_status score is higher in the partners group. Secondly, when we just filter the partners group, the homo_status score is highly negatively correlated with b_IO and is statistically significant. By comparision. in the associates group, the homo_status score is positively correlated with b_IO and is statistically insignificant

6. In summary, the seniority is obviously related with the representative role score. And after processing the data, we can indicate that when there is a social circle with more associates and fewer but not 0 partners, the partners are more likely to be the representative role.

The initial hypothesis holds, which means I'm a person with higher status, I will be more likely to be a representative of my social circle.