INFX 576: Problem Set 5 - Core/Periphery Structure*

Harkar Talwar

Due: Friday, May 4, 2018

Collaborators: Prateek Tripathi, Aakash Agrawal

Instructions:

Before beginning this assignment, please ensure you have access to R and RStudio.

1. Download the `problemset5.Rmd` file from Canvas. You will also need the data from last week's Problem Set 4 in `problemset5_data.Rdata`.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text.
4. Collaboration on problem sets is acceptable, and even encouraged, but each student must turn in an individual write-up in his or her own words and his or her own work. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the R Markdown file to `YourLastName_YourFirstName_ps5.Rmd`, knit a PDF and submit the PDF file on Canvas.

Setup:

In this problem set you will need, at minimum, the following R packages.

```r
# Load standard libraries
library(statnet)
# Load the data set
load("problemset5_data.Rdata")
# Load the function to fit block models
load("block.fit.Rdata")
```

Problem 1: Core/Periphery Structure

In this problem we will use data from a famous series of studies by Bernard, Killworth, and Sailer[1] on the relationship between observed interaction and informants self-reports of interaction. The specific networks we will use here are from the "behavioral" side, meaning that the i, j cell corresponds to the number of times i and j were *observed* to interact during the data collection period. All interaction is these studies is interpersonal; the study contexts are: (1) communication among radio operators (`bfham`), (2) face-to-face interactions among members of a fraterity (`bkfrat`), (3) face-to-face interactions in a university research group (`bktec`), and (4) face-to-face interactions in a small business (`bkoff`). Here we investigate the possibility of latent two-class structure in these interation networks.

*Problems originally written by C.T. Butts (2009)

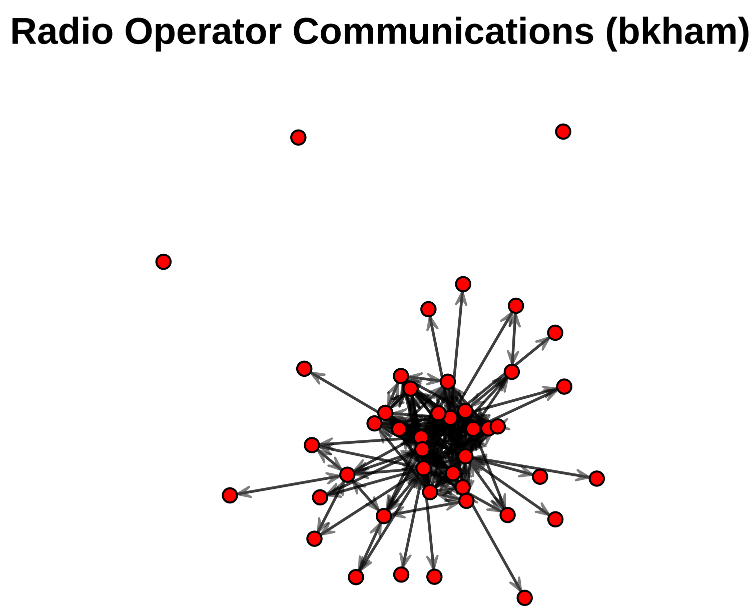
[1] Bernard H, Killworth P and Sailer L. (1982). Informant accuracy in social network data V. Social Science Research, 11, 30-66.

(a) Network Visualization

To begin visualize each network. You might find it helpful to use transparency when displaying edges using the `edge.col=rgb(0,0,0,0.5)` option of the `gplot` function. Based on each visualization, indicate whether there appears to be a two-class block structure present, and if so what it might be.

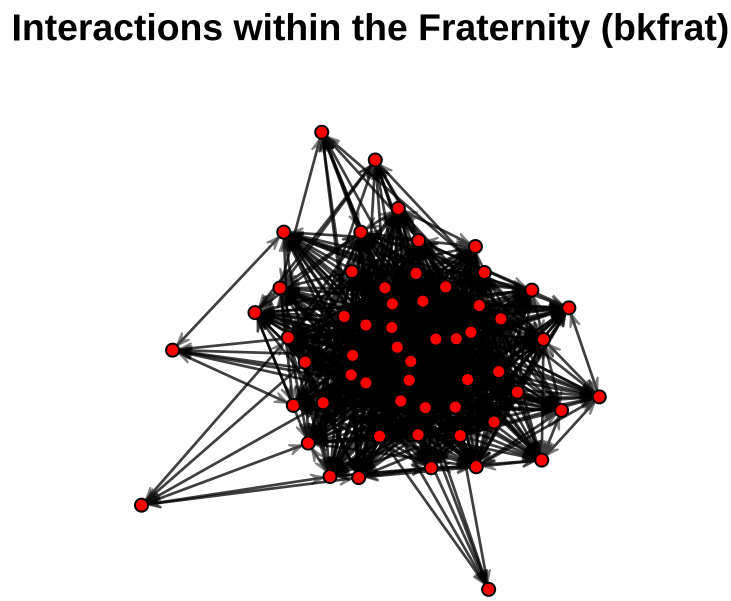
```r
# Function to plot the networks
plot.network.data = function(network.obj, title) {
  gplot(network.obj, edge.col = rgb(0,0,0,0.5), main = title)
}
```

```r
plot.network.data(bkham$Behavioral, "Radio Operator Communications (bkham)")
```



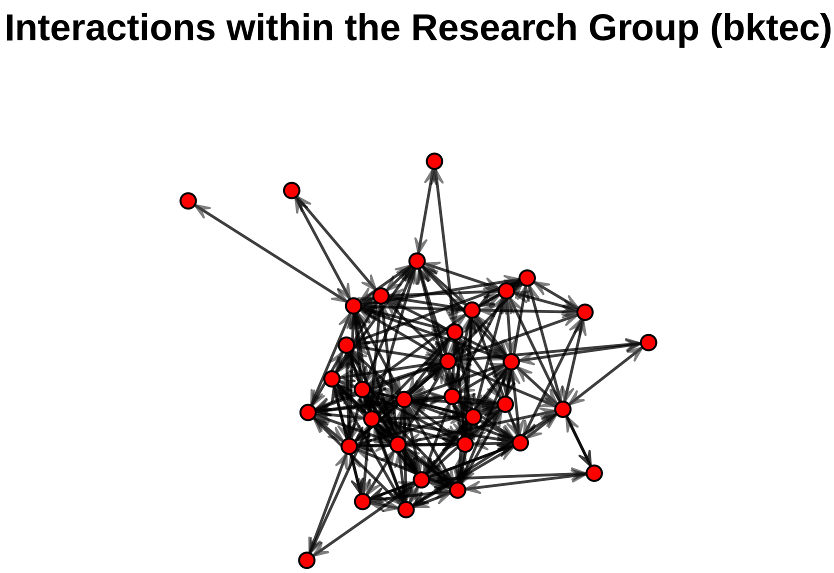
- From the plot above for the bkham network, it seems that there is a two-class block structure. It has a densely connected core, several connections from the core to the periphery, and only very few connections within the peripheral nodes. This resembles the directed (1,0,1,0) structure, with more connections outward from the core to periphery than there are inward from the periphery to the core.

```r
plot.network.data(bkfrat$Behavioral, "Interactions within the Fraternity (bkfrat)")
```



- The plot above for interactions within the fraternity (bkfrat) also seems to depict a two-class block structure. Here however, it seems that there is just a single densely populated core, because the number of nodes outside the core is very few. So, this depicts an idealized blockmodel structure (1,0,0,0)

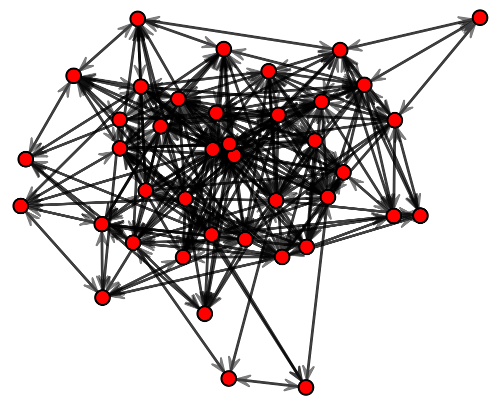
```
plot.network.data(bktec$Behavioral,
                  "Interactions within the Research Group (bktec)")
```



- The interactions among members of the research group (bktec) do not show a strong core-periphery structure. The core does not seem to be as densely connected as it were in the previous two networks. Further, there does not seem to be a clear distinction between the core and the periphery.

```
plot.network.data(bkoff$Behavioral, "Interactions within the Business (bkoff)")
```

Interactions within the Business (bkoff)



- The network of face to face interactions within a business (bkoff) also does not show a strong two-block structure. The core is not as densely connected and distinguishable as it was in the first two networks. This could at best be modeled as a (1,0,0,0) idealized blockmodel.

(b) Blockmodels

For each of the BKS networks, fit each of the four non-degrenerate undirected two-class blockmodels. (You may omit the null graph and complete graph blockmodels.) In addition, fit the Borgatti and Everett variant in which only within-class edges are considered. Plot each blocked data matrix with the `plot.sociomatrix` function. Comment on your results.

```r
# Function to plot sociomatrix for data in the block model form
plot.blocked.matrix = function(model) {
    # extract labels in the right order
    labs<-model$block.membership[model$order.vector]
    # plot the sociomatrix
    plot.sociomatrix(model$blocked.data,labels=list(labs,labs),
                     main = paste("Block Model: c(",
                                  toString(as.vector(model$block.model)), ")",
                                  sep = ""),
                     cex.lab = 0.5)
}
```

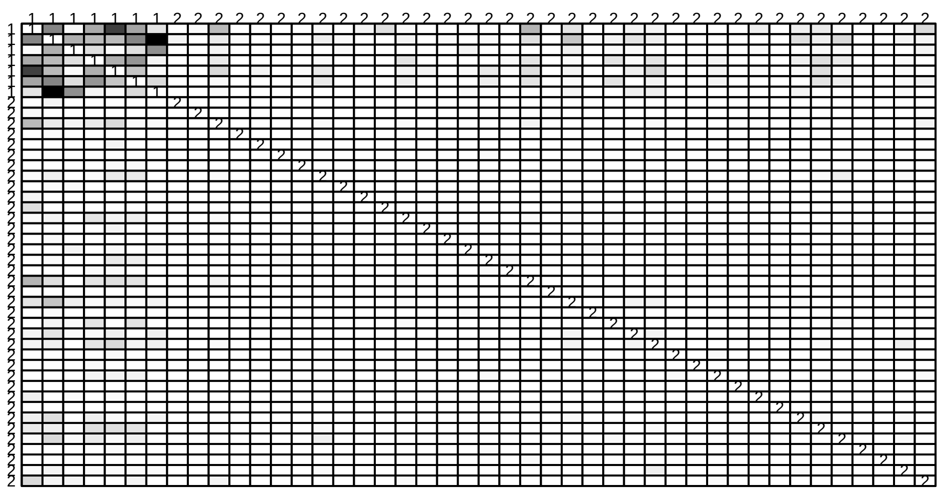
```r
# Function to fit different idealized block models to the passed network object
fit.block.model = function(network.obj) {
  # Fit the four undirected two class block models
  model1 = block.fit(network.obj,c(1,0,0,0), verbose = FALSE)
```

```
  model2 = block.fit(network.obj,c(1,0,0,1), verbose = FALSE)
  model3 = block.fit(network.obj,c(0,1,1,0), verbose = FALSE)
  model4 = block.fit(network.obj,c(1,1,1,0), verbose = FALSE)
  # Fit the Borgatti and Everett variant
  model5 = block.fit(network.obj,c(1,NA,NA,0), verbose = FALSE)
  # Plot the sociomatrix for each
  plot.blocked.matrix(model1)
  plot.blocked.matrix(model2)
  plot.blocked.matrix(model3)
  plot.blocked.matrix(model4)
  plot.blocked.matrix(model5)
  # Return the fitted blockmodels as a list
  list(m1 = model1, m2 = model2, m3 = model3, m4 = model4, m5 = model5)
}
```

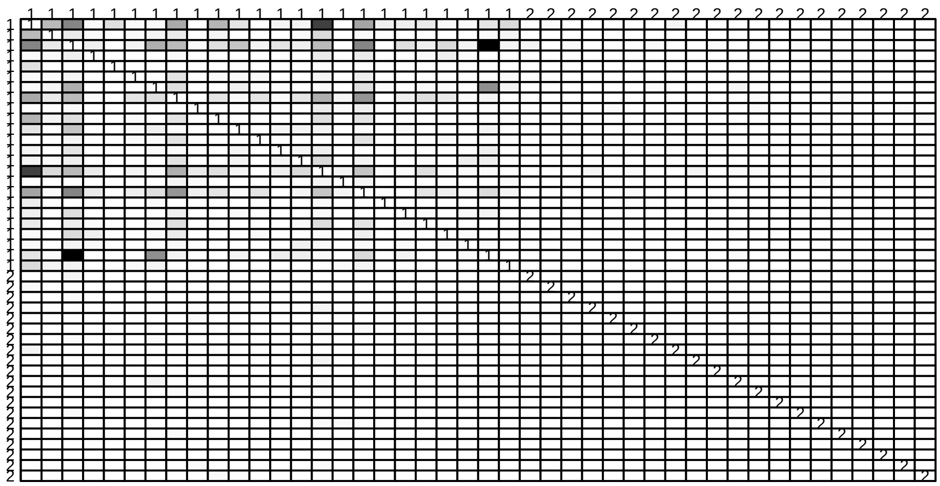
Radio Operator Communications (bkham)

```
models_radio = fit.block.model(bkham$Behavioral)
```

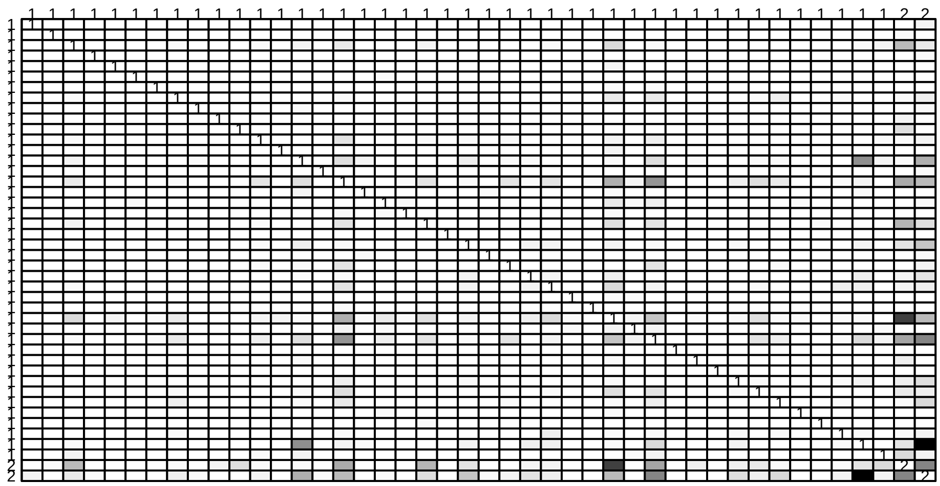
Block Model: c(1, 0, 0, 0)



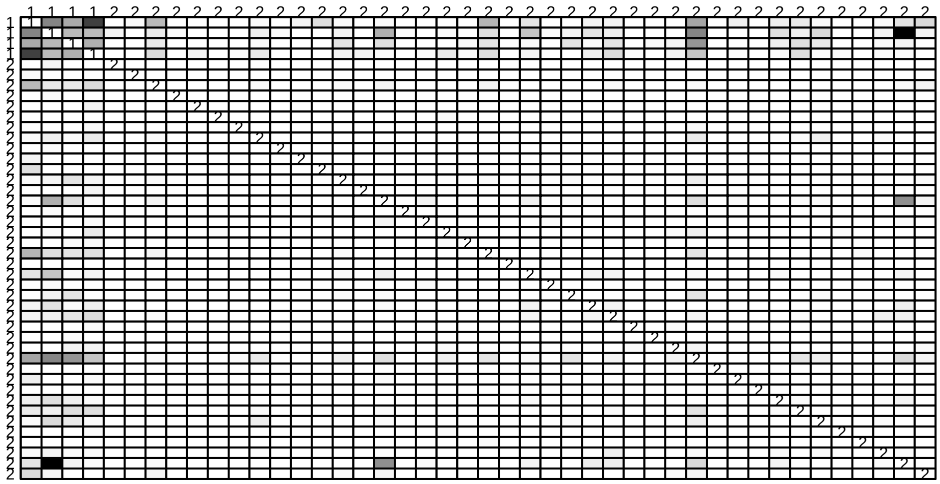
Block Model: c(1, 0, 0, 1)



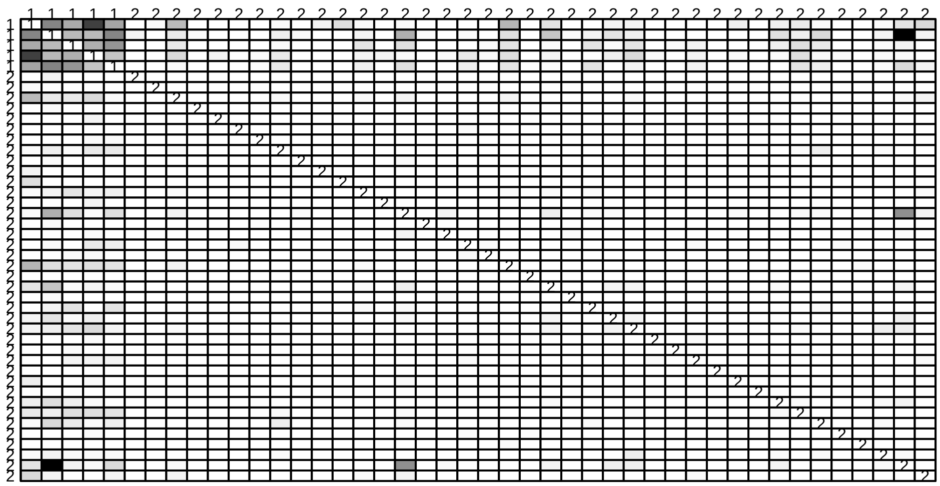
Block Model: c(0, 1, 1, 0)



Block Model: c(1, 1, 1, 0)



Block Model: c(1, NA, NA, 0)

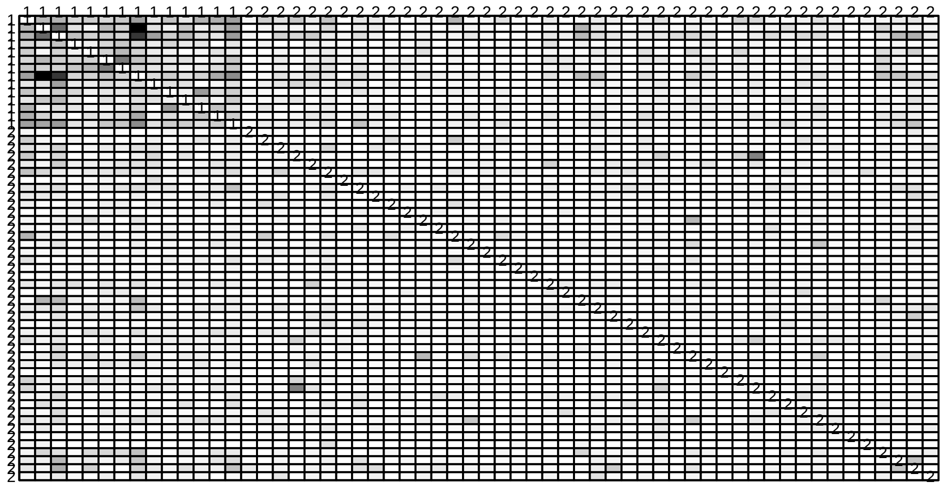


- Here we observe from the above socio-matrices that the radio operator communications data most closely fits with the c(1,0,0,0) or the c(1,NA,NA,0) idealized block structures. This is similar to what we felt intuitively in part a), where we had predicted an idealized structure of c(1,0,1,0). However for the undirected case this essentially translates to our result of c(1,0,0,0).

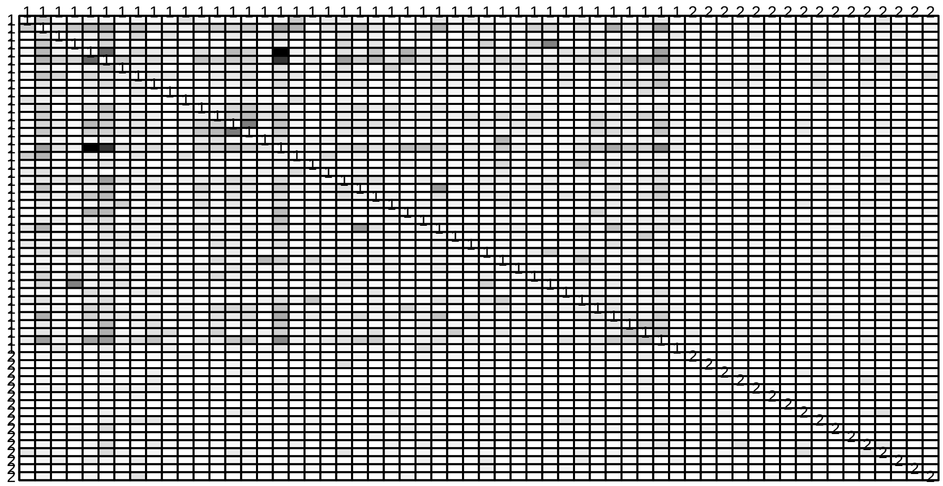
Interactions within the Fraternity (bkfrat)

```
models_frat = fit.block.model(bkfrat$Behavioral)
```

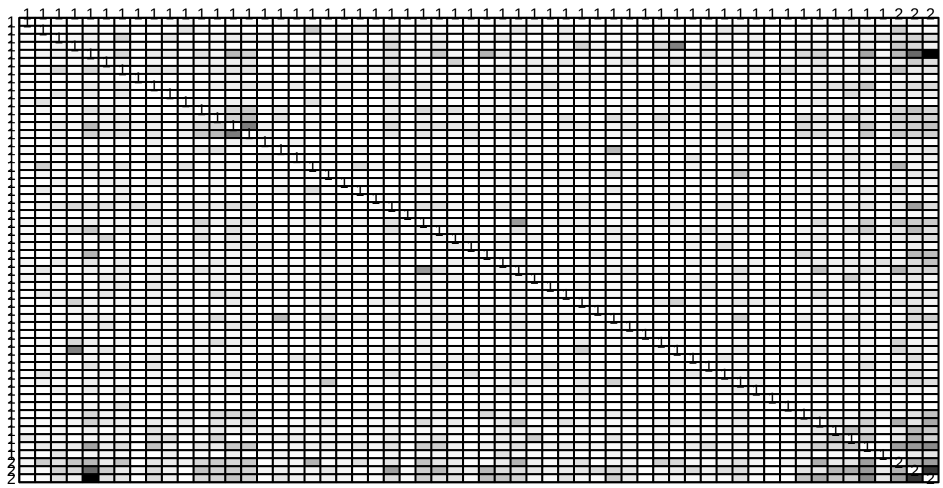
Block Model: c(1, 0, 0, 0)



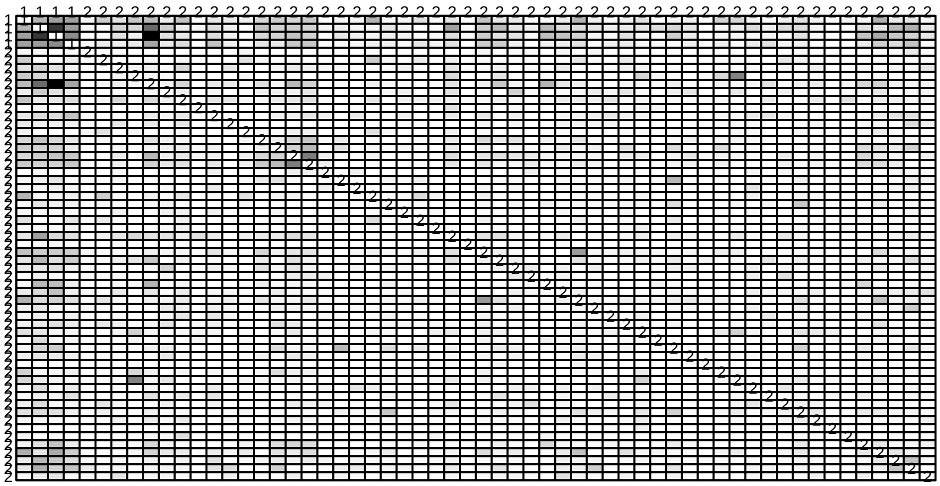
Block Model: c(1, 0, 0, 1)



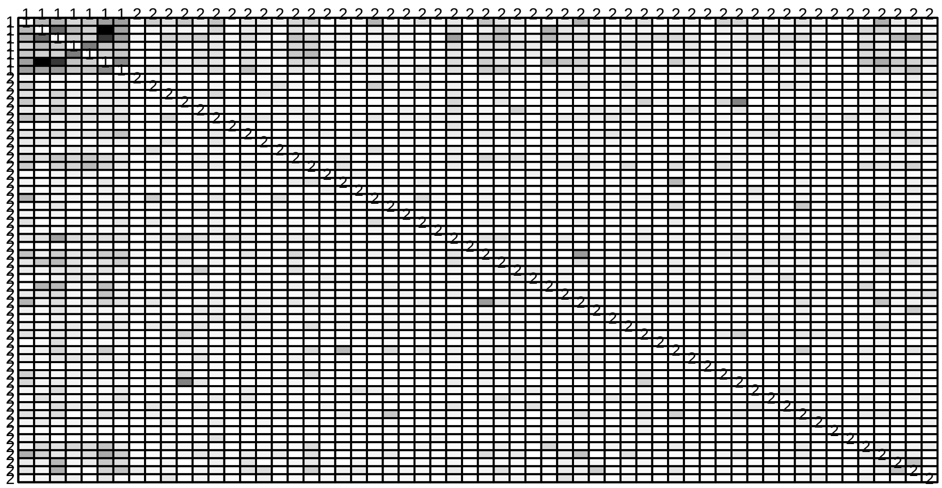
Block Model: c(0, 1, 1, 0)



Block Model: c(1, 1, 1, 0)



Block Model: c(1, NA, NA, 0)

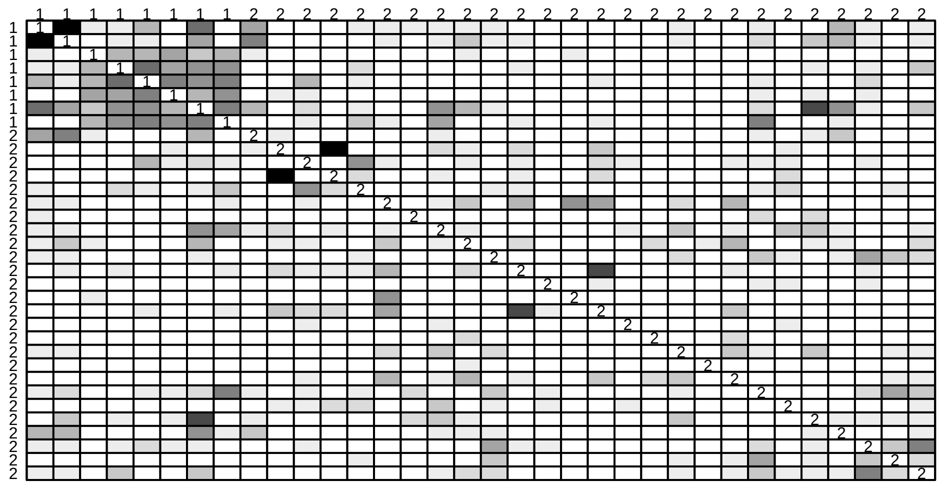


- Here as well, we observe that the model c(1,0,0,0), based on a single densely connected core, with no or very few interactions between the core-periphery or within the periphery fits our data better than the others. The model c(1,NA,NA,0) also fits well, although leading to a considerably small core.

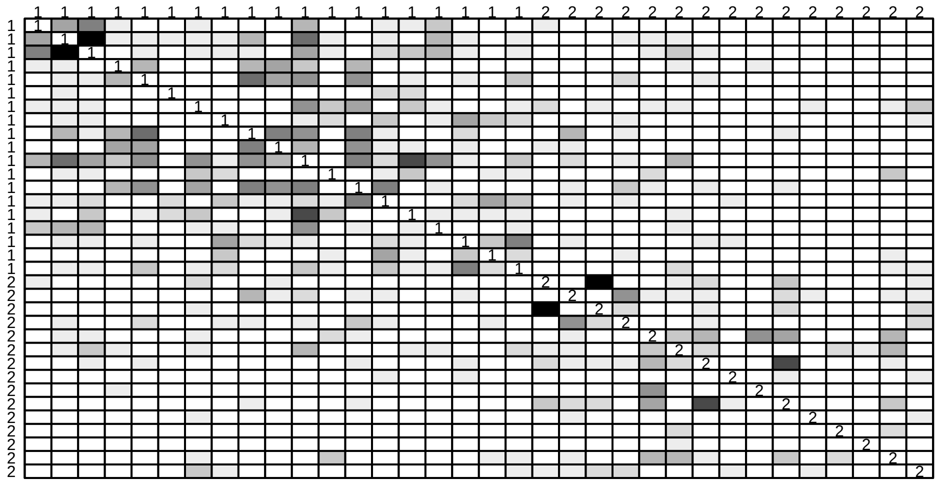
Interactions within the Research Group (bktec)

```
models_research = fit.block.model(bktec$Behavioral)
```

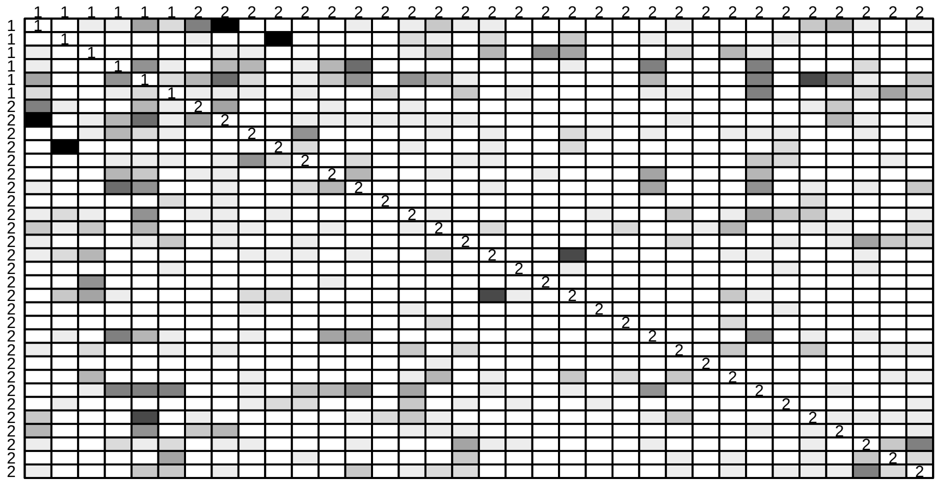
Block Model: c(1, 0, 0, 0)



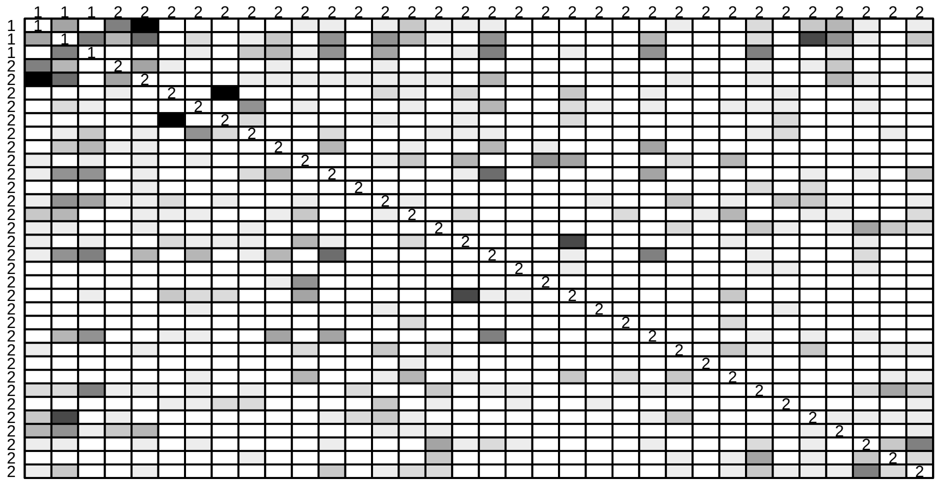
Block Model: c(1, 0, 0, 1)



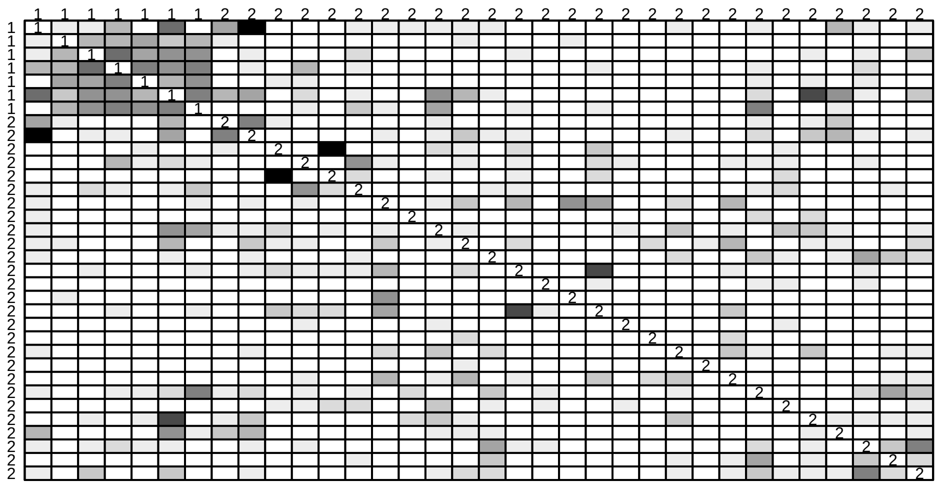
Block Model: c(0, 1, 1, 0)



Block Model: c(1, 1, 1, 0)



Block Model: c(1, NA, NA, 0)

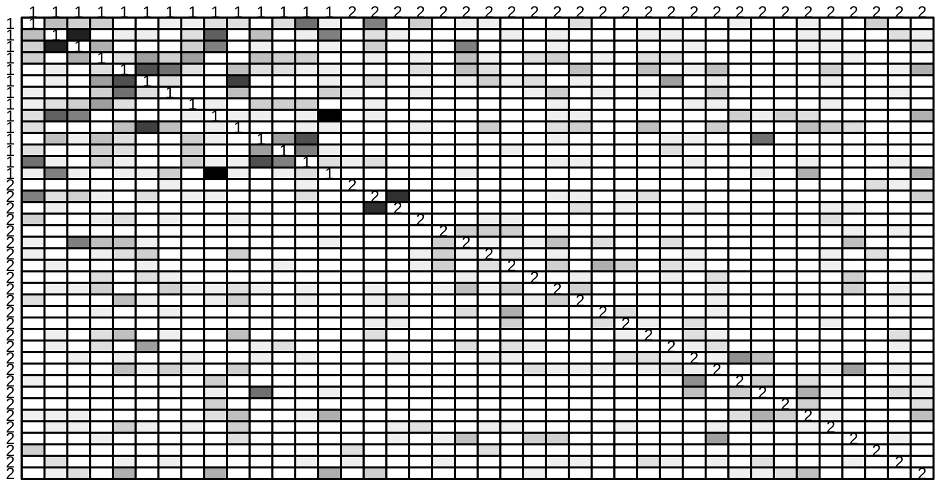


- None of the 5 block models seem to apply terribly well to our research group interactions network (bktec). The models that are comparatively better in explaining the group's interactions are c(1,0,0,0) and c(1,NA,NA,0), i.e., those that depict an isolated core.

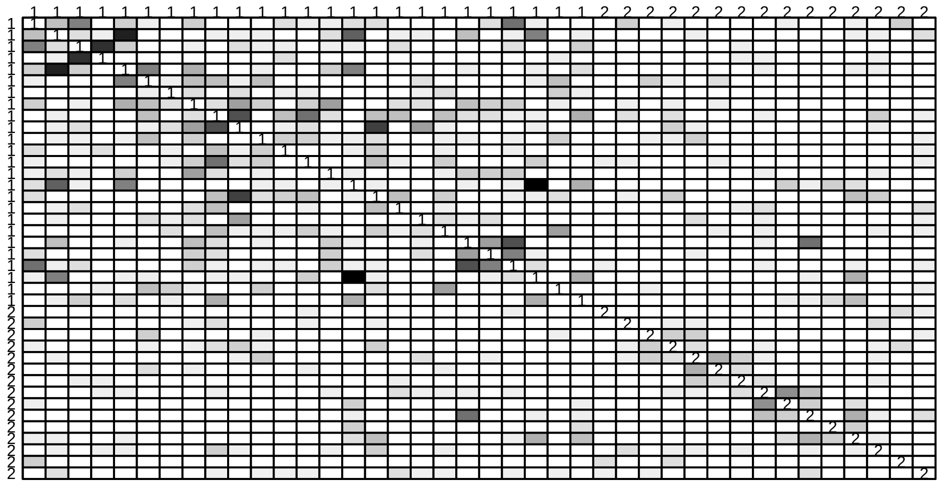
Interactions within the Business (bkoff)

```
models_business = fit.block.model(bkoff$Behavioral)
```

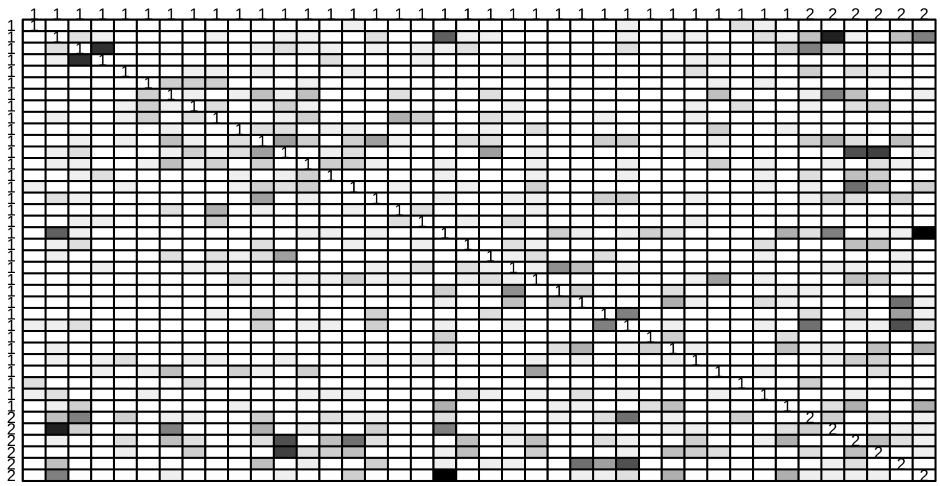
Block Model: c(1, 0, 0, 0)



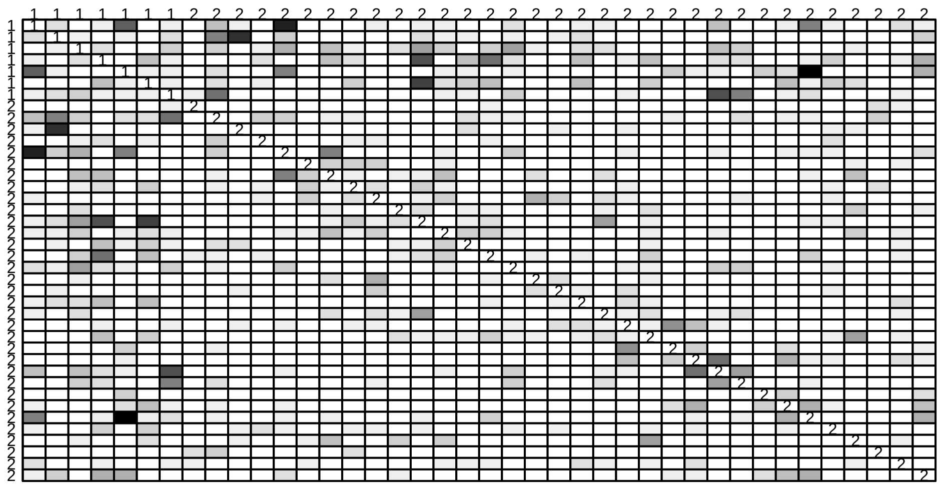
Block Model: c(1, 0, 0, 1)

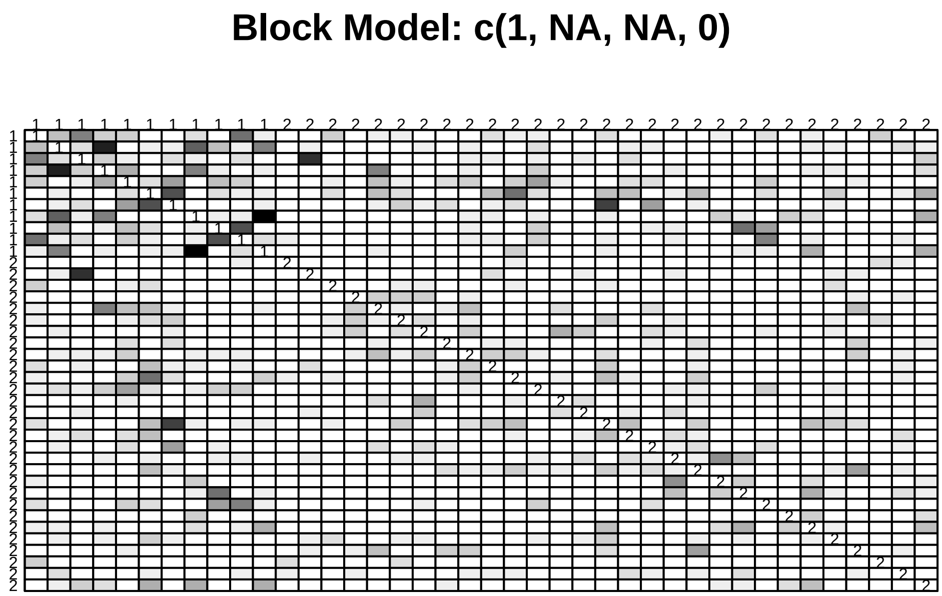


Block Model: c(0, 1, 1, 0)



Block Model: c(1, 1, 1, 0)





- Here again, none of the five block models seem to explain the variations in input data for interactions within the business. The closest that we get to an idealized structure is for the model c(1, NA, NA, 0), where we ignore the interactions between the core and periphery.

(c) Goodness-of-Fit

Examine the goodness-of-fit scores (in this case, maximized correlations) for each model on each network. Which model fits best (among those which seek to explain all edges)? How much variance is accounted for by each model?

```r
# Function to print the goodness of fit and r-squared statistic for a list of models
print.gof = function(models) {
  for (i in 1:length(models)) {
    gof = models[[i]]$block.gof
    cat("Block Model: c(", as.vector(models[[i]]$block.model), "),",
        "GOF: ", gof, ",", "r-squared: ", gof^2, "\n")
  }
}
```

Note: We are not considering the structure c(1, NA, NA, 0), as indicated in the question, while evaluating model fit.

Radio Operator Communications (bkham)

```r
print.gof(models_radio)
```

```
## Block Model: c( 1 0 0 0 ), GOF:  0.6959562 , r-squared:  0.484355
## Block Model: c( 1 0 0 1 ), GOF:  0.2096077 , r-squared:  0.04393539
## Block Model: c( 0 1 1 0 ), GOF:  0.3273251 , r-squared:  0.1071417
## Block Model: c( 1 1 1 0 ), GOF:  0.3625934 , r-squared:  0.1314739
## Block Model: c( 1 NA NA 0 ), GOF:  0.8759063 , r-squared:  0.7672119
```

- Among the models that seek to explain all the edges, the idealized structure c (1,0,0,0) seems to fit the best, with a Goodness of Fit value of 0.69. Since the goodness of fit is computed based on Pearson Correlation, squaring it gives us the r-squared statistic, or the amount of variance accounted for by the model, which is 0.48. Thus, the structure c(1,0,0,0) explains roughly 48% percent of variation in the radio operators communication network. The model c(1,0,0,1) performs the worst, with GOF 0.21 and 4% variance in the data accounted, likely because there are very few interactions within the periphery.

Interactions within the Fraternity (bkfrat)

```r
print.gof(models_frat)
```

```
## Block Model: c( 1 0 0 0 ), GOF:  0.5444052 , r-squared:  0.296377
## Block Model: c( 1 0 0 1 ), GOF:  0.2982005 , r-squared:  0.08892353
## Block Model: c( 0 1 1 0 ), GOF:  0.3730622 , r-squared:  0.1391754
## Block Model: c( 1 1 1 0 ), GOF:  0.4207104 , r-squared:  0.1769973
## Block Model: c( 1 NA NA 0 ), GOF:  0.6707011 , r-squared:  0.4498399
```

- Face to face interactions in the Fraternity are at best explained by the model c(1,0,0,0) with a GOF value of 0.54 and corresponding variance in the input data accounted for by the model is 29%. The model c(1,0,0,1) again performs the worst, with only 8% of the variance captured, because of the limited interactions among members in the periphery.

Interactions within the Research Group (bktec)

```r
print.gof(models_research)
```

```
## Block Model: c( 1 0 0 0 ), GOF:  0.4442833 , r-squared:  0.1973877
## Block Model: c( 1 0 0 1 ), GOF:  0.3199294 , r-squared:  0.1023548
## Block Model: c( 0 1 1 0 ), GOF:  0.2044381 , r-squared:  0.04179494
## Block Model: c( 1 1 1 0 ), GOF:  0.2394306 , r-squared:  0.05732702
## Block Model: c( 1 NA NA 0 ), GOF:  0.5147588 , r-squared:  0.2649766
```

- None of the models explain the interactions in the research group fairly well, leading us to believe that this network (bktec) represents a weak core-periphery structure. Here again the model c(1,0,0,0) performs compartively better than the others, with a GOF of 0.44 and roughly 20% variation in the data accounted for. The worst performing model is c(0,1,1,0), which captures only 4% variation in the data.

Interactions within the Business (bkoff)

```r
print.gof(models_business)
```

```
## Block Model: c( 1 0 0 0 ), GOF:  0.3523318 , r-squared:  0.1241377
## Block Model: c( 1 0 0 1 ), GOF:  0.2397224 , r-squared:  0.05746684
## Block Model: c( 0 1 1 0 ), GOF:  0.1924946 , r-squared:  0.03705417
## Block Model: c( 1 1 1 0 ), GOF:  0.2258117 , r-squared:  0.05099091
## Block Model: c( 1 NA NA 0 ), GOF:  0.4253251 , r-squared:  0.1809015
```

- The bkoff network of interactions within the business is the one least explained by any of the idealized blockmodels. Even the model that assumes an isolated core, i.e. c(1,0,0,0) only achieves a GOF of 0.35

and corresponding variance captured is 12%. The model c(0,1,1,0) fits the worst, with only 2% variance in the data accounted for.

(d) Discussion

Based on the above results, how would you describe the overall structure of these data sets? Are they ultimately similar in form or are there notable differences?

- As per our analysis, the networks bkham (Radio operator interactions) and bkfrat (Interactions among fraternity members) seem to depict a pretty distinct 2-class structure, with strong interactions within the core and limited ties within the periphery or between the core and the periphery.
- Further, the networks bktech (Interactions within the research group) and bkoff (Interactions within the business) are not strongly associated with any of the idealized 2-class models, leading us to believe that they don't have a typical core-periphery structure
- For all the networks, the idealized blockmodel of an isolated core c(1,0,0,0) seems to fit the best (relatively). The goodness of fit for the ideal model however decreases in the order bkham, bkfrat, bktec and bkoff.