Using Ranked Survey Data in Education Research: Methods and Applications

Anna E. Bargagliotti, Susan E. Martonosi, Michael E. Orrison, Austin H. Johnson, Sarah A. Fefer

December 15, 2020

Overview

This file contains the R commands to reproduce the tables and figures of the paper "Using Ranked Survey Data in Education Research: Methods and Applications."

Libraries and Data to Load

The data are found in the file 'SurveyDataComplete.csv'. This dataset is not publicly available on GitHub, but can be requested of the authors for research purposes.

```
rm(list=ls())
library(Matrix)
library(MASS)
library(Rankcluster)
library(dplyr)
library(pmr)
library(caret)
library(nnet) # used for multinomial regression
library(mlogit)
library(rpart)
library(rpart.plot)
fulldata <- read.csv("SurveyDataComplete.csv", header=TRUE, sep=",",
                     na.strings=c("NA"), comment.char="")
```

Methods: Demographic Questions – Data Cleaning

Overall demographic information of respondents

```
fulldata$Gender <- ifelse(fulldata$X1_Male==1, "Male", ifelse(fulldata$X1_Female==1, "Female", NA))
fulldata$TroubleChar <- ifelse(fulldata$X3_Trouble==0, "0", ifelse(fulldata$X3_Trouble==1, "1-2",
                        ifelse(fulldata$X3_Trouble==2, "3-4",
                        ifelse(fulldata$X3_Trouble==3, "5 or more", NA))))
fulldata$TroubleCode <- ifelse(fulldata$X3_Trouble==0, 0, 1)
```

The five students who did not enter their demographic information are:

```
fulldata[which(is.na(fulldata$X1_Male) & is.na(fulldata$X3_Trouble)), 1:3]
```

```
    StudentID School Homeroom
92       1092      1      321
101      1101      1      211
173      1173      1      201
768      3197      3       NA
769      3198      3       NA
```

Interestingly, student 101 completed much of the second half of the survey, including Questions 24 and 25 and will be included in this analysis.

Students 1092, 1173, 3197, and 3198 did not complete any questions in the survey and will be discarded.

```
fulldata <- fulldata[-c(92, 173, 768, 769),]
addmargins(table(Gender=fulldata$Gender, Trouble=fulldata$TroubleChar, useNA="always"))
```

```
        Trouble
Gender     0 1-2 3-4 5 or more <NA> Sum
  Female 370  38  12         7    0 427
  Male   233  73  18        13    0 337
  <NA>     0   0   0         0    1   1
  Sum    603 111  30        20    1 765
```

Responses to Demographic Items (Table 2)

```
addmargins(table(School = fulldata$School, useNA = "always"))
```

```
School
   1    2    3 <NA>  Sum
 253  316  196    0  765
```

```
addmargins(table(Grade=fulldata$X2_Grade, useNA="always"))
```

```
Grade
   5    6    7    8    9   10   11   12 <NA>  Sum
 132  117  148  166   69   65   43   17    8  765
```

```
addmargins(table(PercentGrade=fulldata$X2_Grade, useNA="always")/sum(table(PercentGrade=fulldata$X2_Grade,
                                                                            useNA="always")))
```

```
PercentGrade
         5          6          7          8          9         10
0.17254902 0.15294118 0.19346405 0.21699346 0.09019608 0.08496732
        11         12       <NA>        Sum
0.05620915 0.02222222 0.01045752 1.00000000
```

Methods: Multiple Choice Questions – Descriptive Analysis of PBIS

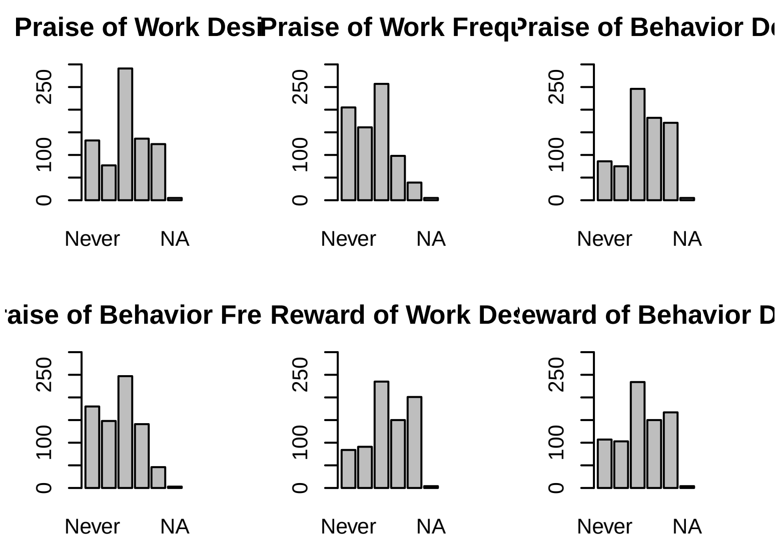
```
tallyPBISlabels <- c("Yes", "No", "IDK", "NA")
tallyPBIS <- c(sum(fulldata$X4_PBIS_Yes, na.rm=TRUE), sum(fulldata$X4_PBIS_No, na.rm=TRUE),
               sum(fulldata$X4_PBIS_IDK, na.rm=TRUE), dim(fulldata)[1]-
              (sum(fulldata$X4_PBIS_Yes, na.rm=TRUE)+sum(fulldata$X4_PBIS_No, na.rm=TRUE)+
              sum(fulldata$X4_PBIS_IDK, na.rm=TRUE)))
percents <- tallyPBIS/sum(tallyPBIS)
dfPBIS <- data.frame(Label = tallyPBISlabels, Count = tallyPBIS, Percent = percents)
print(dfPBIS)
```

```
  Label Count     Percent
1   Yes   422 0.551633987
2    No    91 0.118954248
3   IDK   248 0.324183007
4    NA     4 0.005228758
```

```
fulldata$pbislabel <- ifelse(fulldata$pbis == 1, "Yes", ifelse(fulldata$pbis==2, "No",
                                                      ifelse(fulldata$pbis==3, "IDK", NA)))
```

Histogram of Responses to Questions 9-14 (Figure 1)

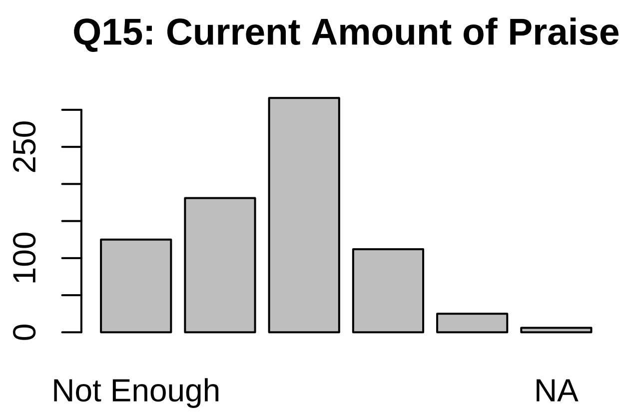
```
#jpeg("scaledresponse.jpg", width = 893, height = 440)
par(mfrow=c(2,3), oma = c(0,0,0,0), mar=c(3,3,3,3))
barplot(table(fulldata$X9_praise_work, useNA="always"),
        names.arg=c("Never", "", "Sometimes", "", "Always", "NA"),
        main = "Q9: Praise of Work Desired", ylim = c(0,300))
barplot(table(fulldata$X10_freq_work, useNA="always"),
        names.arg=c("Never", "", "Sometimes", "", "Always", "NA"),
        main = "Q10: Praise of Work Frequency", ylim = c(0,300))
barplot(table(fulldata$X11_praise_bx, useNA="always"),
        names.arg=c("Never", "", "Sometimes", "", "Always", "NA"),
        main = "Q11: Praise of Behavior Desired", ylim = c(0,300))
barplot(table(fulldata$X12_freq_bx, useNA="always"),
        names.arg=c("Never", "", "Sometimes", "", "Always", "NA"),
        main = "Q12: Praise of Behavior Frequency", ylim = c(0,300))
barplot(table(fulldata$X13_reward_work, useNA="always"),
        names.arg=c("Never", "", "Sometimes", "", "Always", "NA"),
        main = "Q13: Reward of Work Desired", ylim = c(0,300))
barplot(table(fulldata$X14_reward_bx, useNA="always"),
        names.arg=c("Never", "", "Sometimes", "", "Always", "NA"),
        main = "Q14: Reward of Behavior Desired", ylim = c(0,300))
```




```
#dev.off()
```

Histogram of Responses to Question 15 (Figure 2)

```
#jpeg("scaledresponse15.jpg", width = 506, height = 440)
par(mfrow=c(1,1))
barplot(table(fulldata$X15_current, useNA="always"),
        names.arg=c("Not Enough", "", "About Right", "", "Too Much", "NA"),
        main = "Q15: Current Amount of Praise", ylim = c(0,300))
```




```
#dev.off()
```

Counts of Ranking Responses by Type (Table 3)

```
# Get column indices for the relevant columns
maleind <- which(names(fulldata)=="X1_Male")
femaleind <-which(names(fulldata)=="X1_Female")
troubind <- which(names(fulldata)=="X3_Trouble")
studIDind <- which(names(fulldata)=="StudentID")
schoolind <- which(names(fulldata)=="School")
hrind <- which(names(fulldata)=="Homeroom")
q24ind <- grep("X24", names(fulldata))[1:4]
q25ind <- grep("X25", names(fulldata))[1:4]

# Determine complete ranks for each question and for both questions
# Row indices containing complete rankings
fullRowsQ24 <- lengths(lapply(apply(fulldata[,q24ind], 1, unique),sort))==4
fullRowsQ25 <- lengths(lapply(apply(fulldata[,q25ind], 1, unique),sort))==4
fullRowsBoth <- (fullRowsQ24 & fullRowsQ25)
# Complete rows
completeQ24 <- fulldata[fullRowsQ24,]
completeQ25 <- fulldata[fullRowsQ25,]
completeQ24andQ25 <- fulldata[fullRowsBoth, ]
# Count of complete rows for each question and for both questions
numcompQ24 <- sum(fullRowsQ24)
numcompQ25<- sum(fullRowsQ25)
numcompBoth<- sum(fullRowsBoth)

## Check for improper partial rankings
# Entire question is NA
allNARowsQ24 <- apply(fulldata[,q24ind],1, function(x) sum(is.na(x)))==4
allNARowsQ25 <- apply(fulldata[,q25ind],1, function(x) sum(is.na(x)))==4
allNARowsBoth <- apply(fulldata[,c(q24ind, q25ind)],1, function(x) sum(is.na(x)))==8

# Check if a value outside of 1 to 4 was given
maxQ24 <- apply(fulldata[,q24ind],1, max, na.rm=TRUE)
minQ24 <- apply(fulldata[,q24ind],1,min, na.rm=TRUE)
```

```
maxQ25 <- apply(fulldata[,q25ind],1, max, na.rm=TRUE)
minQ25 <- apply(fulldata[,q25ind],1,min, na.rm=TRUE)

badRowsQ24 <- allNARowsQ24 | maxQ24>4 | minQ24 < 1 | maxQ24 < 1 | minQ24 > 4
badRowsQ25 <- allNARowsQ25 | maxQ25>4 | minQ25 < 1 | maxQ25 < 1 | minQ25 > 4
badRowsBoth <- badRowsQ24 |  badRowsQ25
numbad24 <- sum(badRowsQ24)
numbad25 <- sum(badRowsQ25)
numbadBo <- sum(badRowsBoth)

# IDENTIFYING RANKINGS WITH TIES
# Step 1: Filter out NA and nonsensical rows
subdataClean24 <- fulldata[!badRowsQ24,]
subdataClean25 <- fulldata[!badRowsQ25,]
## Find rows where neither 24 nor 25 had ties
subdataCleanBoth <- fulldata[!badRowsBoth,]

# Step 2: Create in tabular form the uniqueRankTypes (i.e. how many first place votes,
# second place votes, etc. a voter gave) along with their frequencies
newTable24 <- matrix(data=rep(0, n=5*dim(subdataClean24)[1]),nrow=dim(subdataClean24)[1], ncol=5)
for(i in 1:dim(subdataClean24)[1]) {
   for(j in 1:4) {
     if(!is.na(subdataClean24[i,q24ind[1]+j-1])) {
       newTable24[i,subdataClean24[i,q24ind[1]+j-1]] <- newTable24[i, subdataClean24[i,q24ind[1]+j-1]]+1
     }
   }
    newTable24[i,5] <- 4 - sum(newTable24[i,1:4])
}
# All ranking types (how many 1st place objects, 2nd place objects, etc) including ties
freqNewVoterTypes24 <- frequence(newTable24)

# Count rankings that have no ties
numNoTies24 <- sum(freqNewVoterTypes24[apply(freqNewVoterTypes24[,1:4],1,max)==1,6])
numTies24 <- sum(freqNewVoterTypes24[apply(freqNewVoterTypes24[,1:4],1,max)>1,6])

newTable25<- matrix(data=rep(0, n=5*dim(subdataClean25)[1]),nrow=dim(subdataClean25)[1], ncol=5)
for(i in 1:dim(subdataClean25)[1]) {
   for(j in 1:4) {
     if(!is.na(subdataClean25[i,q25ind[1]+j-1])) {
       newTable25[i,subdataClean25[i,q25ind[1]+j-1]] <- newTable25[i, subdataClean25[i,q25ind[1]+j-1]]+1
     }
   }
    newTable25[i,5] <- 4 - sum(newTable25[i,1:4])
}
freqNewVoterTypes25 <- frequence(newTable25)
numNoTies25 <- sum(freqNewVoterTypes25[apply(freqNewVoterTypes25[,1:4],1,max)==1,6])
numTies25 <- sum(freqNewVoterTypes25[apply(freqNewVoterTypes25[,1:4],1,max)>1,6])

newTableBoth <- matrix(data=rep(0, n=10*dim(subdataCleanBoth)[1]),nrow=dim(subdataCleanBoth)[1], ncol=10)
for(i in 1:dim(subdataCleanBoth)[1]) {
   for(j in 1:4) {
     if(!is.na(subdataCleanBoth[i,q24ind[1]+j-1])) {
       newTableBoth[i,subdataCleanBoth[i,q24ind[1]+j-1]] <-
         newTableBoth[i, subdataCleanBoth[i,q24ind[1]+j-1]]+1
      }
     if(!is.na(subdataCleanBoth[i,q25ind[1]+j-1])) {
       newTableBoth[i,5+subdataCleanBoth[i,q25ind[1]+j-1]] <-
         newTableBoth[i, 5+subdataCleanBoth[i,q25ind[1]+j-1]]+1
```

```
    }
  }
   newTableBoth[i,5] <- 4 - sum(newTableBoth[i,1:4])
   newTableBoth[i,10] <- 4 - sum(newTableBoth[i,6:9])
}
# All ranking types (how many 1st place objects, 2nd place objects, etc) including ties
freqNewVoterTypesBoth <- frequence(newTableBoth)

# Count rankings that have no ties
numNoTiesBo <- sum(freqNewVoterTypesBoth[(apply(freqNewVoterTypesBoth[,1:4],1,max)==1)&
                                          (apply(freqNewVoterTypesBoth[,6:9],1,max)==1),11])
numTiesBo <- sum(freqNewVoterTypesBoth[(apply(freqNewVoterTypes24[,1:4],1,max)>1) |
                                        (apply(freqNewVoterTypes24[,1:4],1,max)>1),11])

## STEP 3: Append the voter type to each row of the cleaned data and filter out ties.
dataWithTallies24 <- cbind(subdataClean24, newTable24)
names(dataWithTallies24) <- c(names(subdataClean24),"numFirst", "numSecond", "numThird", "numFourth", "numNA")
noTiesData24 <- subset(dataWithTallies24, apply(dataWithTallies24[,which(names(dataWithTallies24)=="numFirst"):
                                                     which(names(dataWithTallies24)=="numFourth")],1,max)==1)

dataWithTallies25 <- cbind(subdataClean25, newTable25)
names(dataWithTallies25) <- c(names(subdataClean25),"numFirst", "numSecond", "numThird", "numFourth", "numNA")
noTiesData25 <- subset(dataWithTallies25, apply(dataWithTallies25[,which(names(dataWithTallies25)=="numFirst"):
                                                     which(names(dataWithTallies25)=="numFourth")],1,max)==1)

dataWithTalliesBoth <- cbind(subdataCleanBoth, newTableBoth)
names(dataWithTalliesBoth) <- c(names(subdataCleanBoth),"numFirstQ24", "numSecondQ24", "numThirdQ24",
  "numFourthQ24", "numNAQ24","numFirstQ25", "numSecondQ25", "numThirdQ25", "numFourthQ25", "numNAQ25")
noTiesDataBoth <- subset(dataWithTalliesBoth, apply(dataWithTalliesBoth[,
                  which(names(dataWithTalliesBoth)=="numFirstQ24"):
                    which(names(dataWithTalliesBoth)=="numFourthQ24")],
                  1,max)==1 & apply(dataWithTalliesBoth[,which(names(dataWithTalliesBoth)=="numFirstQ25"):
                                                  which(names(dataWithTalliesBoth)=="numFourthQ25")],1,max)==1)

## MOVING FORWARD, THE CLEANED DATA FRAME OBJECTS TO USE ARE
## FOR COMPLETE RANKINGS:
# completeQ24, completeQ25, completeQ24andQ25

## FOR PARTIAL RANKINGS WITH NO TIES
# noTiesData24, noTiesData25, noTiesDataBoth
```

– **Q24: Whose opinions do you value more regarding your work in school?**

– **Q25: Whose opinions do you value more regarding your behavior in school?**

– Of 765 respondents, the following types of rankings were received

Question	Complete	Partial (no ties)	Ties	Invalid
24	589	96	44	36
25	589	92	44	40
Both	575	93	49	48

Mean Rankings for Questions 24 and 25 (Table 4)

```
colMeans(completeQ24[,q24ind])
```

```
X24a_work_Teacher  X24b_work_Friend  X24c_work_Parent    X24d_work_self
```

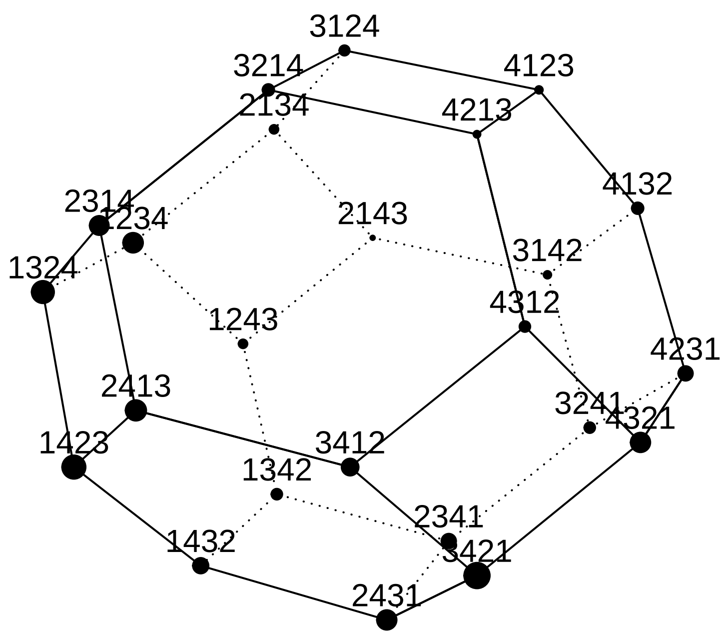
```
       2.244482           3.115450           2.222411           2.417657
```

```r
colMeans(completeQ25[,q25ind])
```

```
X25a_bx_Teacher  X25b_bx_Friend  X25c_bx_Parent    X25d_bx_self
       2.115450        3.083192        2.254669        2.546689
```

Permutahedron for Question 24 (Figure 3)

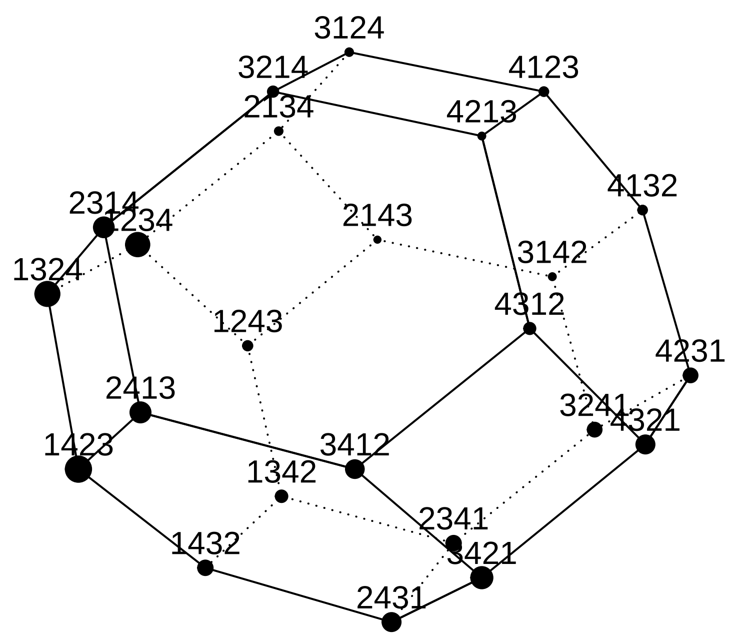
```r
source("rankplot_SM.R")
#jpeg("permutahedronQ24.jpg", width = 500, height = 500)
rankplot_SM(frequence(as.matrix(completeQ24[,q24ind])), label.type = "ranking", circle.bg=TRUE)
```



```
#dev.off()
```

Permutahedron for Question 25 (Figure 4)

```
#jpeg("permutahedronQ25.jpg", width = 500, height = 500)
rankplot_SM(frequence(as.matrix(completeQ25[,q25ind])), label.type = "ranking", circle.bg=TRUE)
```




```
#dev.off()
```

Marginal Distribution for Question 24 and 25 (Table 5)

Marginals: How many times was each object ranked 1st, 2nd, 3rd, 4th:

Question 24:

```r
freqs <- frequence(as.matrix(completeQ24[,q24ind]))
margs <- matrix(data = rep(x=0, times=16),nrow=4, ncol=4)
for(o in 1:4) {
  for(r in 1:4) {
    margs[o,r] <- sum(freqs[freqs[,o]==r,5])
  }
}
margs <- as.data.frame(margs)
row.names(margs) <- c("Teacher", "Friend", "Parent", "Self")
names(margs) <- c("1st", "2nd", "3rd", "4th")
margs
```

```
        1st 2nd 3rd 4th
Teacher 195 154 141  99
Friend   47 102 176 264
Parent  143 234 150  62
Self    204  99 122 164
```

Question 25:

```r
freqs25 <- frequence(as.matrix(completeQ25[,q25ind]))
margs25 <- matrix(data = rep(x=0, times=16),nrow=4, ncol=4)
for(o in 1:4) {
  for(r in 1:4) {
    margs25[o,r] <- sum(freqs25[freqs25[,o]==r,5])
  }
}
margs25 <- as.data.frame(margs25)
row.names(margs25) <- c("Teacher", "Friend", "Parent", "Self")
names(margs25) <- c("1st", "2nd", "3rd", "4th")
margs25
```

```
        1st 2nd 3rd 4th
Teacher 230 149 122  88
Friend   37 122 185 245
Parent  145 223 147  74
Self    177  95 135 182
```

Descriptive Analysis (Q24) (Table 6)

Pairs: How many times was each pair of objects ranked in each pair of positions:

Question 24:

```r
freqs <- frequence(as.matrix(completeQ24[,q24ind]))
pairsmat <- matrix(data = rep(x=0, times=36),nrow=6, ncol=6)
for(o1 in 1:3) {
  for(o2 in (o1+1):4) {
    for(r1 in 1:3) {
      for(r2 in (r1+1):4) {
        rowi <- 6-(4-o1)*(5-o1)/2 + (o2-o1)
        coli <- 6-(4-r1)*(5-r1)/2 + (r2-r1)
        pairsmat[rowi, coli] <-sum(freqs[((freqs[,o1]==r1)&(freqs[,o2]==r2))|
                                ((freqs[,o1]==r2)&(freqs[,o2]==r1)),5])
      }
    }
  }
}
pairsmat <- as.data.frame(pairsmat)
```

```r
row.names(pairsmat) <- c("Teacher/Friend", "Teacher/Parent", "Teacher/Self", "Friend/Parent",
                         "Friend/Self", "Parent/Self")
names(pairsmat) <- c("1st/2nd", "1st/3rd", "1st/4th", "2nd/3rd", "2nd/4th", "3rd/4th")
pairsmat
```

```
               1st/2nd 1st/3rd 1st/4th 2nd/3rd 2nd/4th 3rd/4th
Teacher/Friend      59      81     102      86     111     150
Teacher/Parent     191     110      37     127      70      54
Teacher/Self        99     145     155      82      72      36
Friend/Parent       36      72      82     155     145      99
Friend/Self         54      70     127      37     110     191
Parent/Self        150     111      86     102      81      59
```

Question 25:

```r
pairsmat25 <- matrix(data = rep(x=0, times=36),nrow=6, ncol=6)
for(o1 in 1:3) {
  for(o2 in (o1+1):4) {
    for(r1 in 1:3) {
      for(r2 in (r1+1):4) {
        rowi <- 6-(4-o1)*(5-o1)/2 + (o2-o1)
        coli <- 6-(4-r1)*(5-r1)/2 + (r2-r1)
        pairsmat25[rowi, coli] <-sum(freqs25[((freqs25[,o1]==r1)&(freqs25[,o2]==r2))|
                                       ((freqs25[,o1]==r2)&(freqs25[,o2]==r1)),5])
      }
    }
  }
}
pairsmat25 <- as.data.frame(pairsmat25)
row.names(pairsmat25) <- c("Teacher/Friend", "Teacher/Parent", "Teacher/Self", "Friend/Parent",
                         "Friend/Self", "Parent/Self")
names(pairsmat25) <- c("1st/2nd", "1st/3rd", "1st/4th", "2nd/3rd", "2nd/4th", "3rd/4th")
pairsmat25
```

```
               1st/2nd 1st/3rd 1st/4th 2nd/3rd 2nd/4th 3rd/4th
Teacher/Friend      76      86     105      94     101     127
Teacher/Parent     211     123      41      93      68      53
Teacher/Self        92     143     172      84      68      30
Friend/Parent       30      68      84     172     143      92
Friend/Self         53      68      93      41     123     211
Parent/Self        127     101      94     105      86      76
```

χ^2 Test for Differences

Using χ^2 tests (Marden), we asked whether student rankings in Questions 24 and 25 varied by

– Gender

– Behavior (reporting getting into trouble at least once versus never)

– Gender crossed with behavior

– Use of PBI (Yes, No, I don't know)

– School level (Middle school versus High school)

```r
chisqranks <- function(votes, groups) {
  # votes is an n x m matrix, where n is the number of voters
  # m is the number of objects being ranked
  # Each column corresponds to one object, and the values in the
  # column are the ranks given by each voter for that object.
```

```r
  #
  # groups is an nx1 vector (could be a factor)
  # of groups of voters to be compared. Assumption is that the rows
  # of groups match the rows of voters in votes.
  #
  # chisqranks returns the p-values on tests of differences between
  # the groups of voters based on mean rank and marginal distribution
  #
  # coming soon: pairs comparisons.
  #
  # RESTRICT ONLY TO COMPLETE CASES
  #

  n <- dim(votes)[1] # total number of complete ranks
  m <- dim(votes)[2]   # number of objects being ranked

  labels <- array(unique(groups)) # labels for the unique groups
  L <- length(labels) # number of groups
  # vector with number of voters in each group
  ngroups <- apply(labels, 1, function(x) {sum(groups==x)})
  # List of L vectors of row indices showing which rows belong to each group
  rowInd <- lapply(labels, function(x) {which(groups==x)})

  # MEANS ANALYSIS
  # Create mxL matrix of mean ranks for each group
  meanRanks <-apply(labels, 1, function(x) {colMeans(votes[groups==x,])})
  # dimension of chi-square test = m for means, m^2 for marginals, (m choose 2)^2 for pairs
  p <- dim(meanRanks)[1]

  # Columns are objects, so column mean gives mean rank of each object
  overallMeanRanks <- matrix(colMeans(votes), nrow=m, ncol=1, byrow=T)

  # Create W matrix of differences between group means and overall means
  # W is (m-1)xL
  # We drop the last row so that sigmazhat below will be invertible.
  W <- apply(labels,1, function(x) {meanRanks[,which(labels==x)]-overallMeanRanks})[1:(m-1), ]

  # For each group l create a matrix that sums over the members of that group, i, the value of
  # z(y^(li))-zbar^(l))(z(y^(li))-zbar^(l))^T
  # where z is the vector of ranks given to the objects
  # and zbar is the mean rank given to the objects by people in group l
  # Then sum over the groups and divide by degrees of freedom
  # We skip the m^{th} item otherwise sigmazhat will be singular.
  # Marden p. 99
sigmazhat <- (1/(n-L))*Reduce("+",lapply(labels, function(l)
  {Reduce("+", lapply(rowInd[[which(labels==l)]],function(i)
    {matrix(t(votes[i,(1:(m-1))])-meanRanks[(1:(m-1)),which(labels==l)])%*%
      t(matrix(t(votes[i,(1:(m-1))])-meanRanks[(1:(m-1)),which(labels==l)]))}))}))

omegan <- diag(ngroups) # Diagonal matrix of counts in each group
T2mat <- t(W)%*%solve(sigmazhat)%*%W%*%omegan  # Marden p. 100
trace <- sum(diag(T2mat))  # Marden p. 100
# p-value for difference in mean rankings between groups
pchisqMeans<-pchisq(trace, df = p*(L-1), lower.tail=FALSE)

## MARGINALS ANALYSIS
  # List of L matrices, where meanRanksMar[k,q] = percentage of times object k was ranked q for each group.
meanRanksMar <- lapply(labels, function(x){matrix(unlist(lapply(colnames(votes),
```

```r
                function(k){table(votes[groups==x, k])})),nrow=m, ncol=m, byrow=TRUE)/
                ngroups[which(labels==x)]})


pMar <- m^2 # dimension of chi-square test = m for means, m^2 for marginals, (m choose 2)^2 for pairs

# Overall marginals aggregated over all groups
# overallMeanRanksMar[k,q] = percentage of all voters who ranked object k in position q
overallMeanRanksMar <- matrix(unlist(lapply(colnames(votes),
                      function(k){table(votes[, k])})),nrow=m, ncol=m, byrow=TRUE)/n

# Create W matrix of differences between group marginals and overall marginals
# W is (m-1)^2xL
# We drop the last row and col so that sigmazhat below will be invertible.
WMar <- apply(labels,1, function(x) {meanRanksMar[[which(labels==x)]][1:(m-1), 1:(m-1)]-
    overallMeanRanksMar[1:(m-1), 1:(m-1)]})
WMar2 <- lapply(labels,function(x) {meanRanksMar[[which(labels==x)]][1:(m-1), 1:(m-1)]-
    overallMeanRanksMar[1:(m-1), 1:(m-1)]})
# Relationship between WMar and WMar2:
# WMar2 is a list of L, (m-1)*(m-1) matrices, where the [k,q] element of matrix l is the
# percentage of voters in group l who ranked object k in position q.
# WMar is the flattened (m-1)*(m-1) X L matrix where WMar[[l]][(q-1)*(m-1)+k] = WMar2[[l]][k,q]

# For each group create a matrix that sums over the members of that group the value of
# z(y^(li))-zbar^(l))(z(y^(li))-zbar^(l))^T
# z(y^(li))[(q-1)*(m-1)+k] = 1 if person i in group l ranked object k in position q
# and zero otherwise.
# Then sum over the groups and divide by degrees of freedom
# We skip the m^{th} item and m^{th} position otherwise sigmazhat will be singular.
# Marden p. 99

positions <- as.numeric(1:(m-1))
sigmazhatMar <- (1/(n-L))*Reduce("+",lapply(labels, function(l) {Reduce("+",lapply(rowInd[[which(labels==l)]],
              function(i) {matrix(unlist(lapply(positions,
              function(p){lapply(colnames(votes)[1:(m-1)],
              function(k){ifelse(votes[i, k]==p,1,0)})}))-
                matrix(meanRanksMar[[which(labels==l)]][(1:(m-1)),(1:(m-1))],
                       nrow=(m-1)^2, ncol=1, byrow=FALSE))%*%
                       t(matrix(unlist(lapply(positions, function(p){lapply(colnames(votes)[1:(m-1)],
                       function(k){ifelse(votes[i, k]==p,1,0)})}))-
                         matrix(meanRanksMar[[which(labels==l)]][(1:(m-1)),(1:(m-1))],
                       nrow=(m-1)^2, ncol=1, byrow=FALSE)))})))}))

omeganMar <- diag(ngroups)
T2matMar <- t(WMar)%*%solve(sigmazhatMar)%*%WMar%*%omeganMar
traceMar <- sum(diag(T2matMar))
pchisqMar<-pchisq(traceMar, df = pMar*(L-1), lower.tail=FALSE)


## RETURN RESULTS

#c(pchisqMeans, pchisqMar)
  list(pchisqMeans, pchisqMar, meanRanks, meanRanksMar)
}


# Question 24 Gender Comparison:
g <- as.numeric(completeQ24[, "X1_Female"])
```

```r
v <- data.matrix(completeQ24[, c("X24a_work_Teacher", "X24b_work_Friend", "X24c_work_Parent",
                                 "X24d_work_self")], rownames.force=FALSE)
resgendQ24 <- chisqranks(v,g)
pchisqgendQ24 <- c(resgendQ24[[1]], resgendQ24[[2]])
#
# Question 24 Behavior Comparison:
 g <- as.numeric(completeQ24[, "TroubleCode"]==0)
 resbehaviorQ24 <- chisqranks(v,g)
pchisqbehaviorQ24<-c(resbehaviorQ24[[1]], resbehaviorQ24[[2]])

# Question 24 PBIS:
 g <- as.numeric(ifelse(completeQ24[,"X4_PBIS_Yes"]==1, 1, ifelse(completeQ24[,"X4_PBIS_No"]==1, 2,
                                                  ifelse(completeQ24[,"X4_PBIS_IDK"]==1, 3, NA))))
 goodRows <- !is.na(g)
 votesPBIS <- v[goodRows,]
 gPBIS <- g[goodRows]
 resPBISQ24 <- chisqranks(votesPBIS,gPBIS)
pchisqPBISQ24<- c(resPBISQ24[[1]], resPBISQ24[[2]])

# Question 24 School Type:
 g2 <- as.numeric(ifelse(completeQ24[,"School"]<3, 1, ifelse(completeQ24[,"School"]==3, 2, NA)))
 resGrade2Q24 <- chisqranks(v, g2)
pchisqGrade2Q24 <- c(resGrade2Q24[[1]], resGrade2Q24[[2]])


# Question 24 Gender and Behavior:
g <- as.numeric(ifelse((completeQ24[,"X1_Female"]==1 & completeQ24[,"TroubleCode"]==0), 1,
                ifelse((completeQ24[,"X1_Female"]==1 & completeQ24[,"TroubleCode"]!=0), 2,
                ifelse((completeQ24[,"X1_Female"]==0 & completeQ24[,"TroubleCode"]==0), 3,4))))
resgendbehaviorQ24 <- chisqranks(v,g)
pchisqgendbehaviorQ24<-c(resgendbehaviorQ24[[1]], resgendbehaviorQ24[[2]])


genderRes <- c(pchisqgendQ24)
behaviorRes <- c(pchisqbehaviorQ24)
bothRes <- c(pchisqgendbehaviorQ24)
PBIRes <- c(pchisqPBISQ24)
Grade2Res <- c(pchisqGrade2Q24)
```

Mean Rankings of Items by PBIS for Question 24 (Table 8)

Mean ranks given, by PBIS Use group:

```r
meanRanksPBISQ24 <- resPBISQ24[[3]]
colnames(meanRanksPBISQ24) <- c("Yes", "No", "IDK")
rownames(meanRanksPBISQ24) <- c("Teacher", "Friend", "Parent", "Self")
meanRanksPBISQ24
```

```
             Yes       No      IDK
Teacher 2.099698 2.484211 2.298507
Friend  3.211480 3.057895 2.820896
Parent  2.169184 2.252632 2.388060
Self    2.519637 2.205263 2.492537
```

Marginals Rankings of Items by PBIS Response for Question 24 (Table 9)

Percentage of votes ranking each item in each position, by PBIS Use group:

PBIS - Yes

```
margRanksPBISQ24Yes <- resPBISQ24[[4]][[1]]
colnames(margRanksPBISQ24Yes) <- c("1st", "2nd", "3rd", "4th")
rownames(margRanksPBISQ24Yes) <- c("Teacher", "Friend", "Parent", "Self")
margRanksPBISQ24Yes
```

```
              1st       2nd       3rd        4th
Teacher 0.3776435 0.2688822 0.2296073 0.12386707
Friend  0.0755287 0.1450151 0.2719033 0.50755287
Parent  0.2567976 0.4108761 0.2386707 0.09365559
Self    0.2900302 0.1752266 0.2598187 0.27492447
```

PBIS - No

```
margRanksPBISQ24No <- resPBISQ24[[4]][[2]]
colnames(margRanksPBISQ24No) <- c("1st", "2nd", "3rd", "4th")
rownames(margRanksPBISQ24No) <- c("Teacher", "Friend", "Parent", "Self")
margRanksPBISQ24No
```

```
               1st       2nd       3rd        4th
Teacher 0.24736842 0.2473684 0.2789474 0.22631579
Friend  0.08421053 0.1789474 0.3315789 0.40526316
Parent  0.21052632 0.4210526 0.2736842 0.09473684
Self    0.45789474 0.1526316 0.1157895 0.27368421
```

PBIS - IDK

```
margRanksPBISQ24IDK <- resPBISQ24[[4]][[3]]
colnames(margRanksPBISQ24IDK) <- c("1st", "2nd", "3rd", "4th")
rownames(margRanksPBISQ24IDK) <- c("Teacher", "Friend", "Parent", "Self")
margRanksPBISQ24IDK
```

```
               1st       2nd       3rd       4th
Teacher 0.32835821 0.2686567 0.1791045 0.2238806
Friend  0.08955224 0.2835821 0.3432836 0.2835821
Parent  0.26865672 0.2686567 0.2686567 0.1940299
Self    0.31343284 0.1791045 0.2089552 0.2985075
```

p-values from χ^2 Tests for Mean and Marginal Rankings for Factors, by Question (Table 10)

Question 24:

```
resultsdf <- data.frame(Gender = genderRes, Behavior=behaviorRes, Both=bothRes, PBIS = PBIRes,
                        SchoolLevel = Grade2Res)
row.names(resultsdf) <- c("Q24 Means", "Q24 Marginals")
print(resultsdf)
```

```
                 Gender  Behavior      Both        PBIS SchoolLevel
Q24 Means     0.9294843 0.1141335 0.6686137 0.000939866 0.002653869
Q24 Marginals 0.8357500 0.7444883 0.2304817 0.047163451 0.353491176
```

Question 25:

```
# Question 25 Gender Comparison:
g <- as.numeric(completeQ25[, "X1_Female"])
v <- data.matrix(completeQ25[, c("X25a_bx_Teacher", "X25b_bx_Friend", "X25c_bx_Parent", "X25d_bx_self")],
                 rownames.force=FALSE)
resgendQ25 <- chisqranks(v,g)
pchisqgendQ25 <- c(resgendQ25[[1]], resgendQ25[[2]])
#
```

```r
# Question 25 Behavior Comparison:
 g <- as.numeric(completeQ25[, "TroubleCode"]==0)
 resbehaviorQ25 <- chisqranks(v,g)
pchisqbehaviorQ25 <- c(resbehaviorQ25[[1]], resbehaviorQ25[[2]])

# Question 25 PBIS:
 g <- as.numeric(ifelse(completeQ25[,"X4_PBIS_Yes"]==1, 1, ifelse(completeQ25[,"X4_PBIS_No"]==1, 2,
                                                  ifelse(completeQ25[,"X4_PBIS_IDK"]==1, 3, NA))))
 goodRows <- !is.na(g)
 votesPBIS <- v[goodRows,]
 gPBIS <- g[goodRows]
resPBISQ25 <- chisqranks(votesPBIS, gPBIS)
 pchisqPBISQ25 <- c(resPBISQ25[[1]], resPBISQ25[[2]])

# Question 25 School Type:
 g2 <- as.numeric(ifelse(completeQ25[,"School"]<3, 1, ifelse(completeQ25[,"School"]==3, 2, NA)))
 resGrade2Q25 <- chisqranks(v, g2)
pchisqGrade2Q25 <- c(resGrade2Q25[[1]], resGrade2Q25[[2]])


# Question 25 Gender and Behavior:
g <- as.numeric(ifelse((completeQ25[,"X1_Female"]==1 & completeQ25[,"TroubleCode"]==0), 1,
                ifelse((completeQ25[,"X1_Female"]==1 & completeQ25[,"TroubleCode"]!=0), 2,
                ifelse((completeQ25[,"X1_Female"]==0 & completeQ25[,"TroubleCode"]==0), 3,4))))
resgendbehaviorQ25 <- chisqranks(v,g)
pchisqgendbehaviorQ25<-c(resgendbehaviorQ25[[1]], resgendbehaviorQ25[[2]])


genderRes25 <- c(pchisqgendQ25)
behaviorRes25 <- c(pchisqbehaviorQ25)
bothRes25 <- c(pchisqgendbehaviorQ25)
PBIRes25 <- c(pchisqPBISQ25)
Grade2Res25 <- c(pchisqGrade2Q25)

## Summary Table of results
resultsdf25 <- data.frame(Gender = genderRes25, Behavior=behaviorRes25, Both=bothRes25,
                          PBIS = PBIRes25, SchoolLevel = Grade2Res25)
row.names(resultsdf25) <- c("Q25 Means", "Q25 Marginals")
print(resultsdf25)
```

```
                 Gender  Behavior      Both       PBIS SchoolLevel
Q25 Means     0.3114846 0.1939193 0.3449203 0.01577763   0.1398337
Q25 Marginals 0.9440437 0.8554678 0.9175744 0.24208702   0.8195464
```

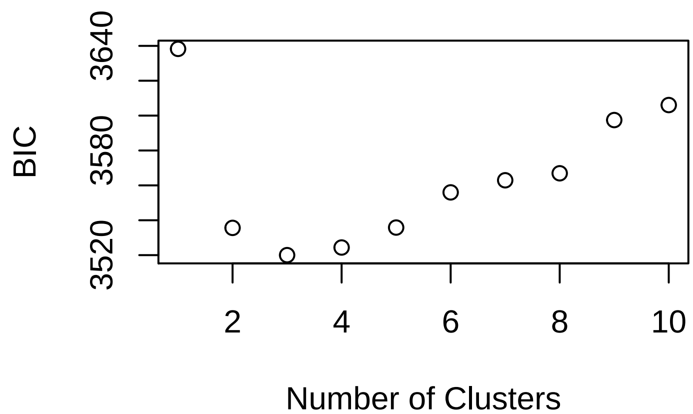
BIC Minimization for Number of Clusters (Figure 7)

Using only complete rankings for Question 24, the optimal number of clusters found in the data is achieved by the minimum point in this graph:

```r
set.seed(123)
# The first time this RMD file is compiled, uncomment the calls to "rankclust" and "save"
# to save the results of rankclust, and comment out the call to "load".
# On subsequent runs, the call to "save" and rankclust" can be commented out and "load" can be uncommented
# to simply load in the fitted clustering model. This will save compilation time.
#result24Comp = rankclust(as.matrix(completeQ24[,q24ind]), m=4, maxTry=10, K=c(1:10), Ql=200, run=5)
# m = ncol, K=nclusters, Ql = Gibbs sampler iterations, run = # independent runs of the algorithm

#save(result24Comp, file="clusterResult24Comp.rda")
```

```r
load("clusterResult24Comp.rda")
bics24Comp <- c(result24Comp@results[[1]]@bic, result24Comp@results[[2]]@bic, result24Comp@results[[3]]@bic,
                result24Comp@results[[4]]@bic, result24Comp@results[[5]]@bic, result24Comp@results[[6]]@bic,
                result24Comp@results[[7]]@bic, result24Comp@results[[8]]@bic, result24Comp@results[[9]]@bic,
                result24Comp@results[[10]]@bic)
plot(bics24Comp, xlab = "Number of Clusters", ylab="BIC")
```



Step 5: Cluster Analysis (Q24) - Complete Rankings (Table 11)

Using $K = 3$ clusters on the complete rankings of Q24, we can identify the cluster centers and the proportion of observations in each cluster.

– Modal rankings for each cluster (Teacher, Friend, Parent Self):

```r
result24Comp@results[[3]]@mu
```

```
    dim1
cl1    1 2 3 4
cl2    1 4 2 3
cl3    3 4 2 1
```

– Proportion of students in each cluster:

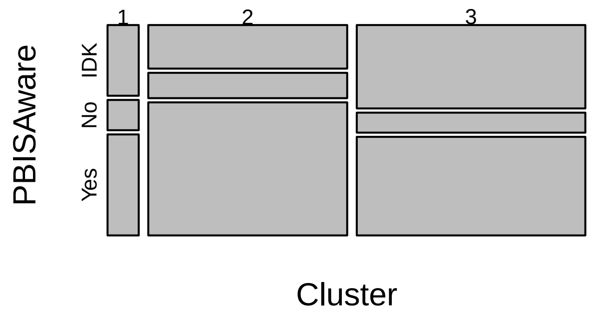
```r
result24Comp@results[[3]]@proportion
```

```
[1] 0.09113314 0.44602662 0.46284024
```

Mosaic Plot Showing Cluster Composition Broken Down by PBIS Awareness: Yes, No, I don't know (IDK) (Figure 8)

For each cluster, we can examine the distribution of students in that cluster by whether or not the student reported awareness of PBIS interventions:

```r
clusterDataComp24 <- cbind(completeQ24, result24Comp@results[[3]]@probability)
#jpeg("MosaicPBIS24.jpg", width = 500, height = 500)
mosaicplot(table(Cluster = clusterDataComp24$cluster, PBISAware = clusterDataComp24$pbislabel), main="")
```



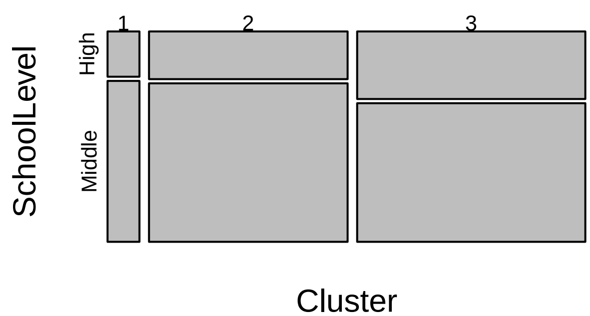

```
#dev.off()
```

Mosaic Plot Showing Cluster Composition Broken Down by School Level: Middle School, High School (Figure 9)

For each cluster, we can examine the distribution of students in that cluster by School Type (Middle School = grades 5 through 8; High School = grades 9 through 12):

```
clusterDataComp24$SchoolType <- ifelse(clusterDataComp24[,"School"]<3, "Middle",
                                ifelse(clusterDataComp24[,"School"]==3, "High",  NA))

#jpeg("MosaicSchoolType24.jpg", width = 500, height = 500)
mosaicplot(table(Cluster = clusterDataComp24$cluster, SchoolLevel = clusterDataComp24$SchoolType), main="")
```




```
#dev.off()
```

Multinomial Logit Cluster Prediction with Important Factors for Question 24 (Table 12)

We fit a multinomial regression model to predict the assigned cluster for question 24 as a function of demographic information and student responses to questions 9 - 15:

Q9. Do you think that you should be praised for doing your work?

Q10. How often are you praised for doing your work?

Q11. Do you think you should be praised for being well behaved in school?

Q12. How often are you praised for being well behaved in school?

Q13. Do you think that you should be rewarded for doing your work?

Q14. Do you think that you should be rewarded for being well behaved in school?

Q15. Overall, do you feel that the amount of praise you currently receive in school is (0 = not enough, 2 = about right, 4 = too much)?

```
modelDatComp24 <- select(clusterDataComp24, X1_Male, SchoolType, pbis,reward, cluster, TroubleCode,
                        X9_praise_work, X10_freq_work, X11_praise_bx, X12_freq_bx, X13_reward_work,
                        X14_reward_bx, X15_current)
modelDatComp24$cluster <- relevel(as.factor(modelDatComp24$cluster), 2)
modelDatComp24<- modelDatComp24[complete.cases(modelDatComp24),]
modComp24 <- multinom(cluster~X1_Male+as.factor(SchoolType)+as.factor(pbis)+as.factor(reward)+
                        TroubleCode + X9_praise_work+X10_freq_work+
                        X11_praise_bx+X12_freq_bx+X13_reward_work+X14_reward_bx+
                        + X15_current, data=modelDatComp24)
bestmodComp24 <-step(modComp24, direction="both", trace=FALSE)
summary(bestmodComp24)
redmodComp24 <- multinom(cluster~as.factor(pbis) + X9_praise_work  +    X11_praise_bx, data=modelDatComp24)
summary(redmodComp24)
```

```
summary(bestmodComp24)$coefficients
```

```
  (Intercept) as.factor(pbis)2 as.factor(pbis)3 X9_praise_work
1  -2.3265469       0.37781651        0.7236319      0.3998337
3   0.2991352      -0.03041334        0.8812911      0.0136852
  X11_praise_bx
1    -0.2853041
3    -0.1856045
```

Negative coefficients indicate a lower likelihood of being in the given cluster than in cluster 1.