Name: Prashant Pandey
ID: 16200112

STAT 40340 Assignment 2

Q1.

```
library(MASS)

hotelling_test <- function(homeData){
homesData.PA <- subset(homeData, homeData$Area =="PA")  #find the data with A
rea PA
homesData.MP <- subset(homeData, homeData$Area =="MP")    #find the data with
Area MP
covPA <- cov(homesData.PA[,-1])    # covariance for PA
covMP <- cov(homesData.MP[,-1])    # covariance for MP
estimate_cov <- (((nrow(homesData.PA)-1)*covPA) + ((nrow(homesData.MP)-1)*cov
MP))/((nrow(homesData.PA)+nrow(homesData.MP))-2)
meanPA <- colMeans(homesData.PA[,-1])
meanMP <- colMeans(homesData.MP[,-1])
mahalanobis_dist<- (t(meanPA-meanMP)%*%solve(estimate_cov)%*%(meanPA-meanMP))
numeratr <- nrow(homesData.PA) + nrow(homesData.MP) - ncol(homesData.PA[,-1])
- 1
denom <- (nrow(homesData.PA) + nrow(homesData.MP) - 2) * ncol(homesData.PA[,-
1])
hotelling_tsq <- (nrow(homesData.PA) * nrow(homesData.MP))/ (nrow(homesData.P
A) + nrow(homesData.MP))
hotelling_tsq <- hotelling_tsq * mahalanobis_dist
f_stat <- (numeratr / denom) * hotelling_tsq
p_val<-pf(f_stat,df1=ncol(homesData.PA[,-1]), df2=numeratr,lower.tail=FALSE)
print(p_val)
if(p_val < 0.05) {  # 0.05 significance level
print("The two communities are significantly different with respect to the ch
aracteristics of the properties available for sale")
} else
  {
  print("The two communities are NOT significantly different with respect to
the characteristics of the properties available for sale")
  }
}
data <- read.csv("prices.csv")    # read prices data
hotelling_test(data)  # test the function
```

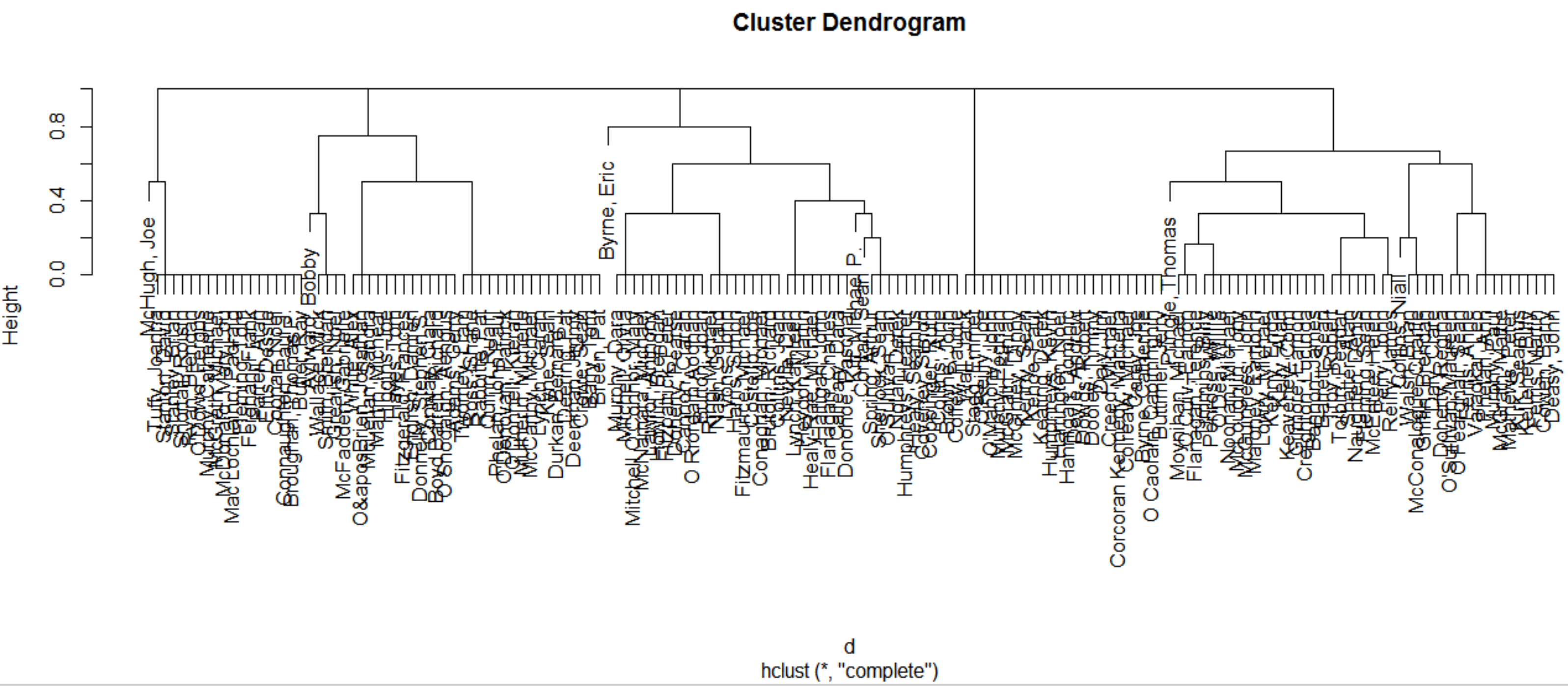
```
          [,1]
[1,] 0.2272253
[1] "The two communities are NOT significantly different with respect to the
characteristics of the properties available for sale"
```

The p-value, `0.2272253` is greater than 0.05 significance level, hence we **fail to reject the null hypothesis** and the two communities are not significantly different.

Q2(a).

```
load("2016_First6Votes_YesNoAbsent.Rdata")
bin_data <- (votes==1)*1   #convert data in binary values
d<- dist(bin_data,method = "binary")  # binary dissimilarity
c<- hclust(d,method = "complete") #hierarchical clustering
plot(c)
hcl = cutree(c, k = 5)
print(table(hcl))
```

Hierarchical clustering can be used. We can find dissimilarity between binary data vectors using Jaccard dissimilarity measure.




```
1  2  3  4  5
24 46 42 35 19   #cluster assignment
```

K=5 seems to be a good option from the above dendrogram.

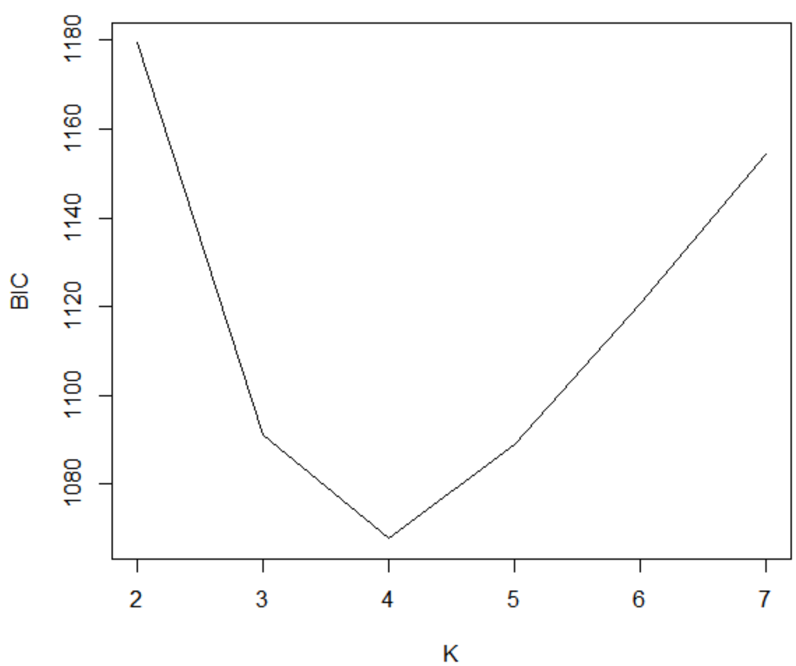
Q2(b)i) binary presence/absence data:

Code for BIC (binary data):

```
library(MASS)
library(poLCA)
load("2016_First6Votes_YesNoAbsent.Rdata")
bin_data <- ((votes==1)*1) +1    # 1 absent 2 present
bin_dataframe<-as.data.frame(bin_data)
rownames(bin_dataframe)<-NULL
f<- cbind(ED1,ED2,Credit,Confidence1,Confidence2,Trade)~1   #formula for poLCA
bic_array=vector()
aic_array=vector()
for(k in 2:7){              # check for cluster k=2 to 7
  min_bic=100000
```

```
  min_aic=100000
  for(j in 1:500){         # try to avoid local maxima by running multiple times
    res1<-poLCA(f, bin_dataframe, nclass = k, maxiter = 10000)
    if(res1$bic < min_bic)
    {
      min_bic = res1$bic
    }
    if(res1$aic < min_aic)
    {
      min_aic = res1$aic
    }
  }
  bic_array<-c(bic_array,c(min_bic))
  aic_array<-c(aic_array,c(min_aic))
}
plot(bic_array,x=c(2:7),t='l',xlab = "K", ylab="BIC")
```

Bayesian Information Criteria(BIC) is calculated for different values of K (number of clusters) and plotted against K using poLCA function. Looking at the plot, for K=4, the BIC is smallest. Hence **K=4** seems to be a good choice for **binary presence/absence data**. I ran poLCA for 10000 iterations. Also for each cluster, I ran PCA 500 times to avoid local maxima.



ii) **polytomous voting data:**

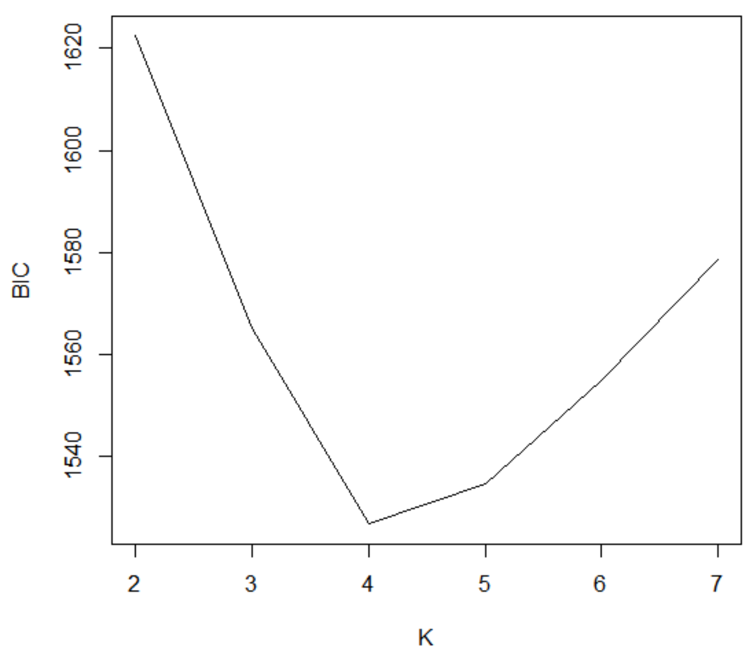
Code for BIC (polytomous data):

```
load("2016_First6Votes_YesNoAbsent.Rdata")

cat_dataframe<-as.data.frame(votes)
rownames(cat_dataframe)<-NULL
f<- cbind(ED1,ED2,Credit,Confidence1,Confidence2,Trade)~1   #formula for poLCA
bic_array=vector()
aic_array=vector()
for(k in 2:7){              # check for cluster k=2 to 7
  min_bic=100000
```

```
  min_aic=100000
  for(j in 1:500){        # try to avoid local maxima by running multiple times
    res<-poLCA(f, cat_dataframe, nclass = k, maxiter = 10000)
    if(res$bic < min_bic)
    {
      min_bic = res$bic
    }
    if(res$aic < min_aic)
    {
      min_aic = res$aic
    }
  }
  bic_array<-c(bic_array,c(min_bic))
  aic_array<-c(aic_array,c(min_aic))
}
plot(bic_array,x=c(2:7),t='l',xlab = "K", ylab="BIC")
```

Bayesian Information Criteria(BIC) is calculated for different values of K (number of clusters) and plotted against K using poLCA function. Looking at the plot, for K=4, the BIC is smallest. Hence **K=4** seems to be a good choice for **polytomous voting data**. I ran poLCA for 10000 iterations. Also for each cluster, I ran PCA 500 times to avoid local maxima.



Q2(c)

Comparison**:**

Between hierarchical clustering and LCA using binary presence/absent data:

```
adjustedRandIndex(hcl,res1$predclass)
[1] 0.4834326
```

Adjusted rand index is 0.4834 which is low so there is significant disagreement between two clustering's. There should be, as hierarchical clustering suggested five clusters while LCA suggested four based on BIC.

Partition of 166 TD's using hierarchical clustering (with 5 clusters):

```
 1  2  3  4  5
24 46 42 35 19
```

Between LCA using binary presence/absent data and polytomous voting data:

Mixing proportion for polytomous voting data

```
> res$P
[1] 0.1927711 0.3433735 0.2351865 0.2286689
```

Partition of 166 TD's using LCA for polytomous data (with 4 clusters):

```
> table(res1$predclass)

1  2  3  4
32 57 39 38
```

Mixing proportion for binary present/absent data:

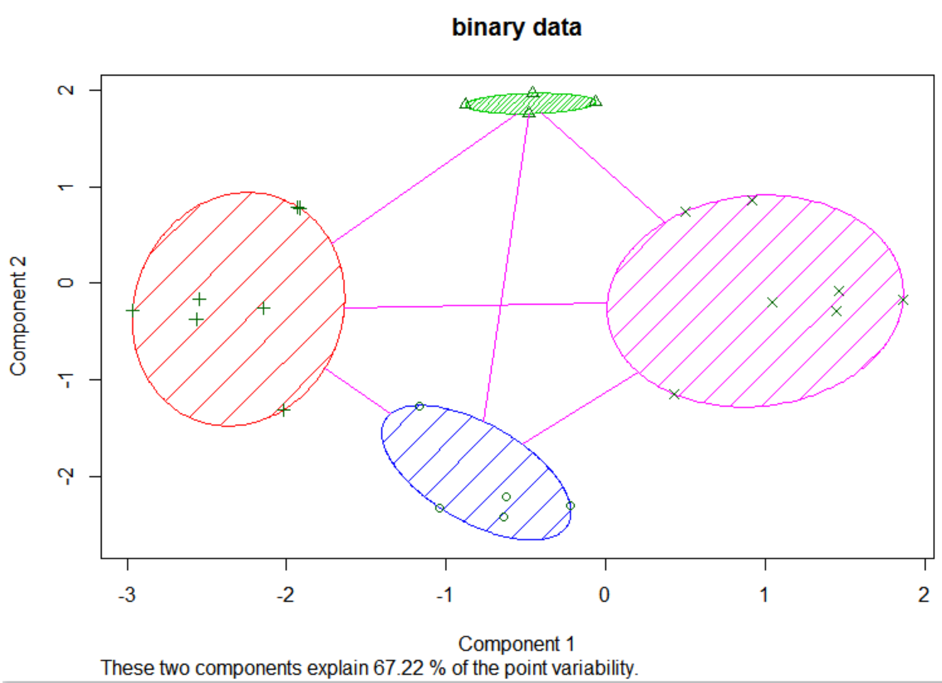
```
> res1$P
[1] 0.1385542 0.2347241 0.1749145 0.4518072
```

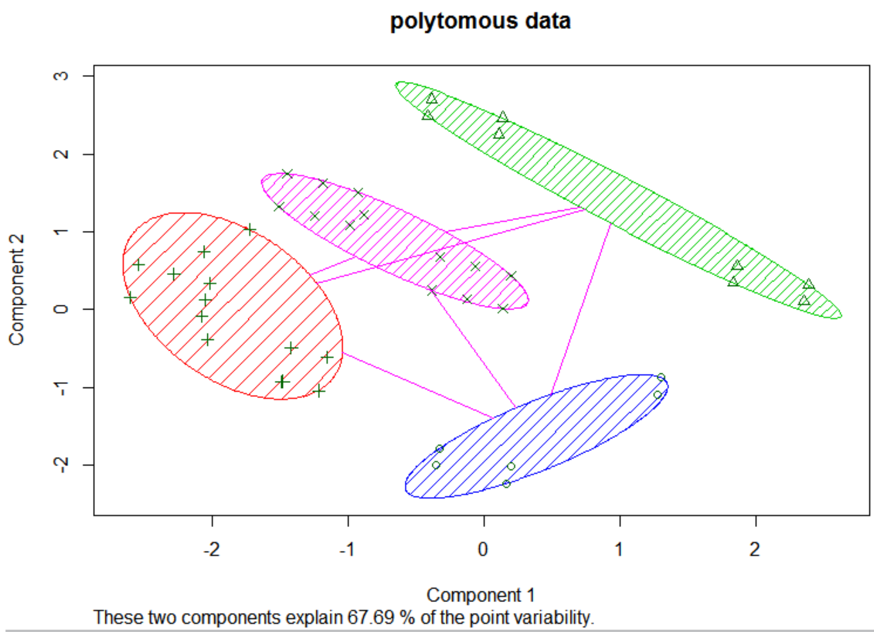
Partition of 166 TD's using LCA for binary data (with 4 clusters):

```
> table(res1$predclass)

 1  2  3  4
23 39 29 75

> adjustedRandIndex(res$predclass,res1$predclass)
[1] 0.4936148
```





With polytomous data, the mixing proportions are less skewed as compared to binary data.

Voting patterns in Dáil Éireann in January 2016

Abstract: The paper groups members of Dail Éireann, called Teachtaí Dála (TDs), to uncover similar voting patterns. It examines and discuss the membership of the clusters, the political affiliation of each TD, the cluster specific parameters and the clustering uncertainties, based on polytomous Latent Class Analysis(LCA). The basic latent class model is a finite mixture model in which the component distributions are assumed to be multi-way cross-classification tables with all variables mutually independent. Memberships are estimated using poLCA software package which finds latent class and latent class regression models for polytomous outcome variables, implemented in the R statistical computing environment.

Introduction: The method of election to each House is different. The 166 members of Dail Éireann (who are called Deputies or TDs) are directly elected by the people. Citizens aged 18 years and over may vote. By law, a General Election to Dáil Éireann must be held at least every five years. The votes of TDs are recorded and posted on www.oireachtas.ie. Between January 14th 2016 and January 21st 2016 inclusive there were 6 votes in Dáil Éireann. Data on whether each TD voted yes, no or was absent for each of these six votes has been scraped from the web. The data records, if a TD was absent for a vote (coded 1) or voted no (coded 2) or voted yes (coded 3). 6 votes are six variables of the data which are ED1, ED2, Credit, Confidence1, Confidence2, Trade.

Latent class analysis is a statistical technique for the analysis of multivariate categorical data. When observed, data take the form of a series of categorical responses—as, for example, in public opinion surveys, individual-level voting data, studies of inter-rater reliability, or consumer behavior and decision-making—it is often of interest to investigate sources of confounding between the observed variables, identify and characterize clusters of similar cases, and approximate the distribution of observations across the many variables of interest. Latent class models are a useful tool for accomplishing these goals. The model, in effect, probabilistically groups each observation into a "latent class," which in turn produces expectations about how that observation will respond on each manifest variable. Although the model does not automatically determine the number of latent classes in each data set, it does offer a variety of parsimony and goodness of fit statistics.

The latent class model is a type of finite mixture model. The component distributions in the mixture are cross-classification tables of equal dimension to the observed table of manifest variables, and, following the assumption of conditional independence, the frequency in each cell of each component table is simply the product of the respective class-conditional marginal frequencies (the parameters estimated by the latent class model are the proportion of observations in each latent class, and the probabilities of observing each response to each manifest variable, conditional on latent class). A weighted sum of these component tables forms an approximation (or, density estimate) of the distribution of cases across the cells of the observed table. Observations with similar sets of responses on the manifest variables will tend to cluster within the same latent classes. The model may also be fit to manifest variables that are ordinal, but they will be treated as nominal.

Polytomous LCA: Lets' say there are J polytomous categorical variables (the "manifest" variables), each of which contains Kj possible outcomes, for individuals i = 1, . . . , N. The manifest variables may have different numbers of outcomes, hence the indexing by j. Denote as Yijk the observed values of the J manifest variables such that Yijk = 1 if respondent i gives the kth response to the jth variable, and Yijk = 0 otherwise, where j = 1, . . . , J and k = 1, . . . , Kj. The latent class model approximates the observed joint distribution of the manifest variables as the weighted sum of a finite number, R, of constituent cross-classification tables. R is fixed prior to estimation based on either theoretical reasons or model fit. Let πjrk denote the class-conditional probability that an observation in class r = 1, . . . , R produces the kth outcome on the jth variable. Within each class, for each manifest variable, therefore, $\sum_{k=1}^{K_j} \pi_{jrk} = 1$. Further denote as pr the R mixing proportions that provide the weights in the weighted sum of the component tables, with $\sum_r p_r = 1$.The values of pr are also referred to as the "prior" probabilities of latent class membership, as they represent the unconditional probability that an individual will belong to each class before considering the responses Yijk provided on the manifest variables. The probability that an individual i in class r produces a set of J outcomes on the manifest variables, assuming conditional independence of the outcomes Y given class memberships, is the product

$$f(Y_i;\pi_r) = \prod_{j=1}^{J}\prod_{k=1}^{K_j}(\pi_{jrk})^{Y_{ijk}}.$$

The probability density function across all classes is the weighted sum

$$\mathrm{P}(Y_i|\pi,p) = \sum_{r=1}^{R} p_r \prod_{j=1}^{J}\prod_{k=1}^{K_j}(\pi_{jrk})^{Y_{ijk}}.$$

The parameters estimated by the latent class model are pr and πjrk. Given estimates ˆpr and ˆπjrk of pr and πjrk, respectively, the posterior probability that each individual belongs to each class, conditional on the observed values of the manifest variables, can be calculated using Bayes' formula:

$$\widehat{\mathrm{P}}(r_i|Y_i) = \frac{\hat{p}_r f(Y_i;\hat{\pi}_r)}{\sum_{q=1}^{R}\hat{p}_q f(Y_i;\hat{\pi}_q)}.$$

where ri ∈ {1, . . . , R}. Recall that the ˆπjrk are estimates of outcome probabilities conditional on class r.

poLCA estimates the latent class model by maximizing the log-likelihood function

$$\ln L = \sum_{i=1}^{N} \ln \sum_{r=1}^{R} p_r \prod_{j=1}^{J}\prod_{k=1}^{K_j}(\pi_{jrk})^{Y_{ijk}}$$

with respect to pr and πjrk, using the expectation-maximization (EM) algorithm. This log-likelihood function is identical in form to the standard finite mixture model log-likelihood. As with any finite mixture model, the EM algorithm is applicable because each individual's class membership is unknown and may be treated as missing data. The EM algorithm proceeds iteratively. Begin with arbitrary initial values of ˆpr and ˆπjrk, and label them ˆpold r and ˆπoldjrk. In the expectation step, calculate the "missing" class membership probabilities, substituting in ˆpoldr and ˆπoldjrk. In the maximization step, update the parameter estimates by maximizing the log-likelihood function given these posterior ˆP(ri |Yi), with

$$\hat{p}_r^{new} = \frac{1}{N}\sum_{i=1}^{N}\widehat{\mathrm{P}}(r_i|Y_i)$$

as the new prior probabilities and

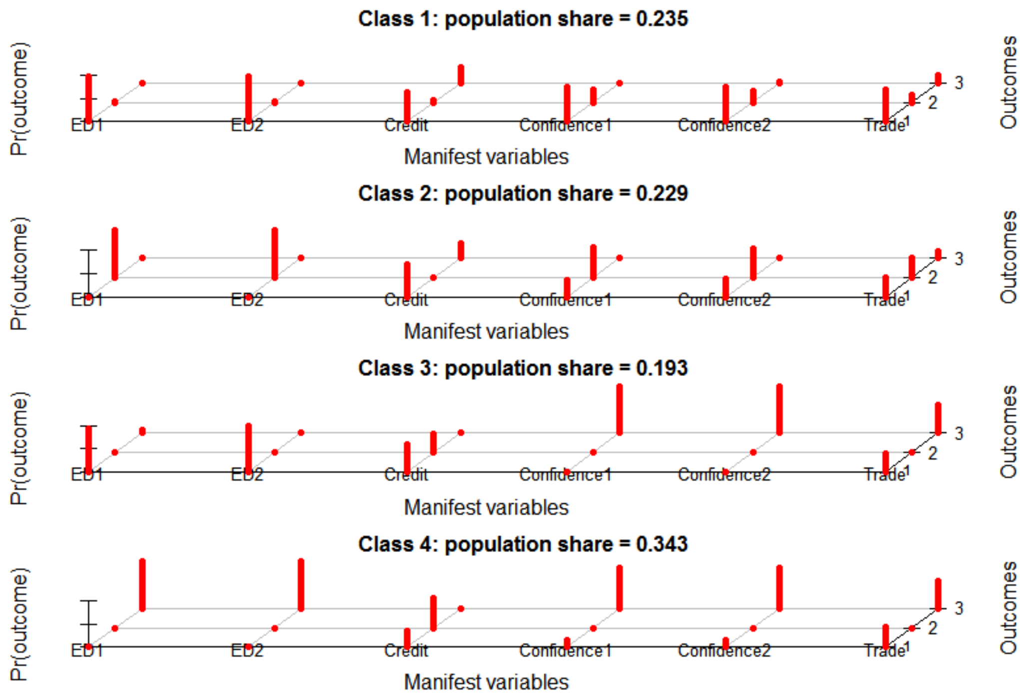
$$\hat{\pi}_{jr}^{new} = \frac{\sum_{i=1}^{N} Y_{ij}\widehat{\mathrm{P}}(r_i|Y_i)}{\sum_{i=1}^{N}\widehat{\mathrm{P}}(r_i|Y_i)}$$

as the new class-conditional outcome probabilities; ˆπnewjr is the vector of length Kj of class-r conditional outcome probabilities for the jth manifest variable; and Yij is the N × Kj matrix of observed outcomes Yijk on that variable. The algorithm repeats these steps, assigning the new to the old, until the overall log-likelihood reaches a maximum and ceases to increment beyond some arbitrarily small value.

Model selection is done by determining an appropriate number of latent classes R for a given data set. **Bayesian Information Criteria (BIC)** is used to locate the best fitting or most parsimonious model. Parsimony criteria seek to strike a balance between over- and under-fitting the model to the data by penalizing the log-likelihood by a function of the number of parameters being estimated. Let Λ represent the maximum log-likelihood of the model and Φ represent the total number of estimated parameters,

$$\mathrm{BIC} = -2\Lambda + \Phi \ln N.$$

Analysis of polytomous voting data: Estimated four clusters (latent classes) using minimum BIC criteria. Voting patterns by different TD's are plotted above. Taller bar shows higher probability of a type (absent 1 or NO 2 or Yes 3) of vote in that class. Class 1 are 23.5% of total TD's who **were mainly absent for different votes**. Few of them present voted No for Confidence votes and Yes for credit. Class 2 are 22.9% of the total TD's **who mainly voted No** except for few Yes credit votes.



Absentees were in the credit, confidence and trade votes. Class 3 are 19.3% of the total TD's **who never voted No** except for the credit vote. Either they were absent like for the ED1, ED2, Trade and credit votes or **predominantly voted Yes for Confidence votes**. There is an equal chance that they are either absent for the Trade vote or vote yes for it. Class 4 are 34.3% of total TD's who were present most of the time except for few trade and credit votes. **They predominantly voted yes** for all the votes except for credit where they voted no more as compared to remaining absent. There is an equal chance that they are either absent for the Trade vote or vote yes for it.

The output of the poLCA from the model estimation is shown below.

```
$ED1
           Pr(1)  Pr(2)  Pr(3)
class 1:  0.9733 0.0267 0.0000
class 2:  0.0000 1.0000 0.0000
class 3:  0.9375 0.0000 0.0625
class 4:  0.0000 0.0000 1.0000
```

```
$Credit
           Pr(1)  Pr(2)  Pr(3)
class 1:  0.6153 0.0512 0.3335
class 2:  0.6844 0.0000 0.3156
class 3:  0.5938 0.4062 0.0000
class 4:  0.3509 0.6491 0.0000
```

```
$Confidence1
           Pr(1)  Pr(2)  Pr(3)
class 1:  0.7438 0.2562 0.0000
class 2:  0.3678 0.6322 0.0000
class 3:  0.0000 0.0000 1.0000
class 4:  0.1404 0.0000 0.8596
```

```
$Confidence2
           Pr(1)  Pr(2)  Pr(3)
class 1:  0.7438 0.2306 0.0256
class 2:  0.3941 0.6059 0.0000
class 3:  0.0000 0.0000 1.0000
class 4:  0.1404 0.0000 0.8596
```

```
$Trade
           Pr(1)  Pr(2)  Pr(3)
class 1:  0.6922 0.1538 0.1540
class 2:  0.4208 0.4214 0.1578
class 3:  0.4063 0.0000 0.5937
class 4:  0.4211 0.0000 0.5789
```

```
$ED2
           Pr(1)  Pr(2) Pr(3)
class 1:  0.9733 0.0267     0
class 2:  0.0000 1.0000     0
class 3:  1.0000 0.0000     0
class 4:  0.0000 0.0000     1
```

```
number of observations: 166
BIC: 1526.591
```

First, the estimated class conditional response probabilities ˆπjrk are reported for ED1, ED2, Credit, Confidence1, Confidence2, Trade, with each row corresponding to a latent class, and each column corresponding to a vote; absent in first column, No vote in the second and Yes vote in the third. For example, if a TD belongs to class 1, it has a 97.33% chance of remaining absent for ED1, 61.53% for credit, 74.38% for Confidence 1, 74.38% for Confidence 2, 69.22% for Trade and 97.33% for ED2. Similarly, if a TD belongs to class 3, it has a 6.25% chance of vo ting a Yes for ED1, 0% for credit, 100% for Confidence 1, 100% for Confidence2, 59.37% for Trade, 0% for Ed2. In this way, all voting probabilities of TD's belonging to different classes can be estimated.

Cluster memberships of different TD's:

class1

"O Fearghail, Sean","O Riordain Aodhan","Barrett, Sean","Burton, Joan","Calleary, Dara","Cannon, Ciaran","Collins, Aine","Conaghan, Michael", "Corcoran Kennedy,Marcella","Daly, Clare","Farrell, Alan","Ferris, Martin","Fitzgerald, Frances","Fitzpatrick, Peter","Fleming, Sean","Halligan, John", "Hannigan, Dominic","Higgins, Joe","Humphreys, Kevin", "Keating, Derek","Kelleher, Billy","Kenny, Sean","Lowry, Michael","Martin, Micheal", "McFadden, Gabrielle","McGrath, Mattie","McLellan, Sandra","Murphy, Catherine","Murphy, Paul","Nash, Gerald","O'Dowd, Fergus","Penrose, Willie","Perry, John", "Pringle, Thomas","Reilly, James","Shatter, Alan","Toibin, Peadar","Timmins, Billy","Troy, Robert"

class2

"O Caolain, Caoimhghin","O Cuiv, Eamon","Adams, Gerry","Bannon, James","Carey, Joe","Coonan, Noel","Costello, Joe","Coveney, Simon","Deasy, John","Deering, Pat","Donohoe, Paschal","Ellis, Dessie","Fitzmaurice, Michael","Flanagan, Charles","Harris, Simon","Humphreys, Heather","Kehoe, Paul","Lawlor, Anthony","Lynch, Ciaran","Lynch, Kathleen","Mac Lochlainn, Padraig","Maloney, Eamonn","McConalogue, Charlie","McDonald, Mary Lou","McEntee, Helen","McGrath, Finian","McNamara, Michael","Moynihan, Michael","O'Donnell, Kieran","O'Donovan, Patrick","O'Mahony, John","O'Sullivan, Jan","O'Sullivan, Maureen","Ross, Shane","Shortall, Roisin","Smith, Brendan","Stanley, Brian","Wallace, Mick"

class3

"Barry,Tom","Boyd,Barrett,Richard","Bruton,Richard","Butler,Ray","Byrne,Catherine","Byrne,Eric","Coffey,Paudie","Connaughton,Paul,J.","Conway,Ciara","Deenihan,Jimmy","Feighan,Frank","Ferris,Anne","Grealish,Noel","Harrington,Noel","Hayes,Tom","HealyRae,Michael","Kenny,Enda","Kitt,Michael,P.","Kyne,Sean","McGinley,Dinny","McGrath,Michael","McGuinness,John","McHugh,Joe","Mitchell,O'Connor,Mary","Mulherin, Michelle","Murphy,Dara","Murphy,Eoghan","O'Dea,Willie","O'Reilly,Joe","Ring,Michael","Sherlock,Sean","Spring,Arthur"

class4

"OSnodaigh,Aengus","Aylward,Bobby","Breen,Pat","Broughan,ThomasP.","Browne,John","Buttimer,Jerry","Collins,Joan","Collins,Niall","Colreavy,Michael","Conlan,Sean","Coppinger,Ruth","Cowen,Barry","Creed,Michael","Creighton,Lucinda","Crowe,Sean","Daly,Jim","Doherty,Pearse","Doherty,Regina","Donnelly,Stephen S.","Dooley,Timmy", "Dowds,Robert","Doyle,Andrew","Durkan,BernardJ","English,Damien","Flanagan,Terence","Fleming,Tom","Gilmore,Eamon","Griffin,Brendan","Healy,Seamus","Heydon,Martin","Howlin,Brendan","Keaveney,Colm","Kelly,Alan","Kirk, Seamus","Lyons,John","Mathews,Peter","McCarthy,Michael","McLoughlin,Tony","Mitchell,Olivia","Naughten,Denis","Neville,Dan","Nolan,Derek","Noonan,Michael","OaposBrien,Jonathan","Phelan,Ann","Phelan,JohnPaul","Quinn,Ruairi","Rabbitte,Pat","Ryan,Brendan","Stagg,Emmet","Stanton,David","Tuffy,Joanna","Twomey,Liam","Varadkar,Leo","Wall,Jack","Walsh,Brian","White,Alex"

Entropy of the model gives an estimate of the disagreement among different TD's. Zero value indicates perfect clustering without any randomness. The entropy of this model is 3.81344 which shows significant disagreement among the TD's while voting.

Clustering based on party of TD's:

Party	**1 2 3 4 (Latent Class or cluster)**
Anti-Austerity Alliance	0 1 0 0
Ceann Comhairle	1 0 0 0
Fianna Fail	10 10 0 0
Fine Gael	13 1 18 41
Independent	6 8 0 1
Labour	5 3 14 15
People Before Profit Alliance	1 1 0 0
Sinn Fein	2 12 0 0
Socialist Party	1 2 0 0

Sinn Fein party seems to have low disagreement in voting among its members. While there is a high disagreement in *Fine Gael* and *Labour* parties. A moderate disagreement in there in *Fianna Fail* and *Independent* party. *Fin Gael* and *Labour* party members belong to cluster 4. These members vote predominantly yes to all votes except credit where they vote No. Both these parties also have members who never voted No except for the credit vote. Either they were absent like for the ED1, ED2, Trade and credit votes or predominantly voted Yes for Confidence votes. There is an equal chance that they are either absent for the Trade vote or vote yes for it. Significant number of members from *Sinn Fein, Independent* and *Fianna Fail* belong to cluster 2 who mainly voted No for all votes except for few Yes credit votes. Some members from *Fine Gael*, *Independent, Labour* and *Fianna Fail* belong to cluster 1 who were mainly absent for different votes. Few of them present voted No for Confidence votes and Yes for credit. Similarly rest of the parties vote according to the assigned clusters.

Standard errors of the estimated class-conditional response probabilities:

$ED1

	1	2	3
class1	0.02935616	0.02935616	0.0000000
class2	0.04384110	0.00000000	0.0438411
class3	0.00000000	0.00000000	0.0000000
class4	0.00000000	0.00000000	0.0000000

$ED2

	1	2	3
class1	2.671080e-02	0.0267108	0.000000e+00
class2	0.000000e+00	0.0000000	0.000000e+00
class3	3.875798e-42	0.0000000	0.000000e+00
class3	1.478374e-20	0.0000000	1.478378e-20

$Credit

	1	2	3
class1	0.09361868	0.04219373	0.08788006
class2	0.08695705	0.08695705	0.00000000
class3	0.07846879	0.00000000	0.07846879
class4	0.06727140	0.06727140	0.00000000

$Confidence1

	1	2	3
class1	1.961454e-01	0.1961454	0.000000e+00
class2	2.725327e-29	0.0000000	2.999737e-29
class3	2.518096e-01	0.2518096	0.000000e+00
class4	2.452011e-02	0.0000000	2.452011e-02

$Confidence2

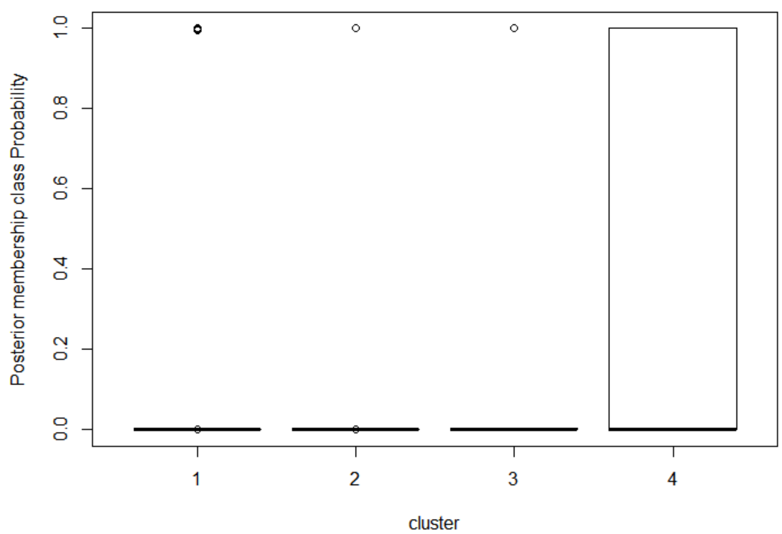
	1	2	3
class1	0.18584690	0.1917890	0.02658890
class2	0.00000000	0.0000000	0.00000000
class3	0.25571388	0.2557139	0.00000000
class4	0.02452011	0.0000000	0.02452011

$Trade

	1	2	3
class1	0.09029714	0.07053941	0.07240807
class2	0.08890264	0.00000000	0.08890264
class3	0.08219574	0.08180223	0.06151304
class4	0.06563811	0.00000000	0.06563811

Error or uncertainty for class conditional response probabilities ˆπjrk are reported for ED1, ED2, Credit, Confidence1, Confidence2, Trade, with each row corresponding to a latent class, and each column corresponding to a vote; absent in first column, No vote in the second and Yes vote in the third. For example, if a TD belongs to class 1, it has a 2.9% uncertainty of remaining absent for ED1, 9.3% for credit, 19.6% for Confidence 1, 18.5% for Confidence 2, 9.09% for Trade and 2.67% for ED2. Similarly, if a TD belongs to class 3, it has a 0% uncertainty of voting a Yes for ED1, 7.84% for credit, 0% for Confidence 1, 0% for Confidence2, 6.15% for Trade, 0% for ED2. In this way, all voting uncertainties of TD's belonging to different classes can be estimated.
The uncertainty on estimated mixing proportions (for 4 latent class) are 0.0326213, 0.0306171, 0.03294238, 0.03685433 respectively.

Posterior class membership probabilities are plotted using box-plot:



Cluster 4 seems to be vary too much for the range of posterior probabilities. This make this cluster unstable and member from this cluster can move to other clusters (i.e. they can change memberships).

Conclusion:

Polytomous Latent Class Analysis is a powerful method to determine patterns in multivariate categorical data which is shown on the voting patterns of TD's. In the voting example presented, there is significant variation in the way TD's vote and remain absent for different votes. There is also disagreement in a party regarding voting as shown in the paper. With poLCA we can predict which party and its members gives importance to what votes. Vote for variable like Credit is voted very differently as compared to other votes. The manifest variables(votes) may contain any number of possible (polytomous) outcomes. poLCA package in R also includes tools for visualizing model results, estimating class conditional probabilities, mixture probabilites and postprocessing the model results to produce various other quantities of interest, including posterior probabilities of latent class membership for either observed or hypothetical cases, uncertainties or error in different parameters.
Model is simple to use and easy to implement in R.

Appendix

```
load("2016_First6Votes_YesNoAbsent.Rdata")

cat_dataframe<-as.data.frame(votes)
rownames(cat_dataframe)<-NULL
f<- cbind(ED1,ED2,Credit,Confidence1,Confidence2,Trade)~1   #formula for poLCA
bic_array=vector()
aic_array=vector()
for(k in 2:7){             # check for cluster k=2 to 7
  min_bic=100000
  min_aic=100000
  for(j in 1:500){         # try to avoid local maxima by running multiple times
    res<-poLCA(f, cat_dataframe, nclass = k, maxiter = 10000)
    if(res$bic < min_bic)
    {
      min_bic = res$bic
    }
    if(res$aic < min_aic)
    {
      min_aic = res$aic
    }
  }
  bic_array<-c(bic_array,c(min_bic))
  aic_array<-c(aic_array,c(min_aic))
}
plot(bic_array,x=c(2:7),t='l',xlab = "K", ylab="BIC")
load("PartyMembership.Rdata")
res<-poLCA(f, cat_dataframe, nclass = 4, maxiter = 10000, graphs = TRUE)
tab<-table(members.party$TD,res$predclass)
cluster1=vector()  # create empty cluster to hold each member
cluster2=vector()
cluster3=vector()
cluster4=vector()
for(i in 1:166){
  for(j in 1:4){
    if(tab[i,][j]==1)
    {
      if(j==1)
      {
        cluster1=c(cluster1,c(members.party[i,]$TD))
      }
      else if(j==2)
      {
        cluster2=c(cluster2,c(members.party[i,]$TD))
      }
      else if(j==3)
      {
        cluster3=c(cluster3,c(members.party[i,]$TD))
      }
      else
      {
        cluster4=c(cluster4,c(members.party[i,]$TD))
      }
    }
  }
}
print(table(members.party$Party,res$predclass))
print(res$probs.se)

yvect<-vector()  # boxplot of posterior probabilities
xvect<-vector()
for(clust in 1:4)
{
  yvect=c(yvect,c(as.numeric(unlist(res$posterior[,clust]))))
  xvect=c(xvect,c(rep(clust,length(unlist(res$posterior[,clust])))))
}
newdf<-data.frame(y=yvect,x=xvect)  #reshape the dataframe
with(newdf,    boxplot(y~x,xlab="cluster",    ylab="Posterior    membership    class
Probability",ylim = c(0, 1)))
```