Abortion by Petrella Riccardo

The data contain responses given by 410 individuals to four out of seven items concerning attitude to abortion. A small number of individual (379) did not answer to some of the questions and this data set contains only the complete cases.

POINT 1

```
library(ltm)
```

```
## Warning: package 'ltm' was built under R version 4.3.2

## Loading required package: MASS

## Loading required package: msm

## Warning: package 'msm' was built under R version 4.3.2

## Loading required package: polycor

## Warning: package 'polycor' was built under R version 4.3.2
```

```
data(Abortion)
dim(Abortion)
```

```
## [1] 379   4
```

LATENT VARIABLE = APTITUDE TOWARDS ABORTION

POINT 2 *Use the function descript to summarize the principal characteristics of the data set data set.*

We compute the descriptive statistics for dichotomous and polytomous data.

```
dsc <- descript(Abortion)
dsc
```

```
##
## Descriptive statistics for the 'Abortion' data-set
##
## Sample:
##  4 items and 379 sample units; 0 missing values
##
## Proportions for each level of response:
##             0      1   logit
## Item 1 0.5620 0.4380 -0.2493
## Item 2 0.4063 0.5937  0.3791
## Item 3 0.3641 0.6359  0.5575
## Item 4 0.3826 0.6174  0.4786
##
##
## Frequencies of total scores:
##        0  1  2  3   4
## Freq 103 33 37 65 141
##
##
## Point Biserial correlation with Total Score:
##        Included Excluded
## Item 1   0.8164   0.6673
## Item 2   0.8674   0.7541
## Item 3   0.8769   0.7737
## Item 4   0.8355   0.7025
##
##
## Cronbach's alpha:
##                   value
## All Items        0.8707
## Excluding Item 1 0.8573
## Excluding Item 2 0.8223
## Excluding Item 3 0.8148
## Excluding Item 4 0.8430
##
##
## Pairwise Associations:
##   Item i Item j p.value
## 1      1      4  <2e-16
## 2      1      3  <2e-16
## 3      2      4  <2e-16
## 4      1      2  <2e-16
## 5      2      3  <2e-16
## 6      3      4  <2e-16
```

We have many descriptive statistics. The more interesting are:

```
dsc$perc  #Percentage of people that replied 1 or 0 to each item. For each proportion we have
```

```
##                 0         1      logit
## Item 1 0.5620053 0.4379947 -0.2493044
## Item 2 0.4063325 0.5936675  0.3791478
## Item 3 0.3641161 0.6358839  0.5575432
## Item 4 0.3825858 0.6174142  0.4785874
```

```
#also the logit = log(percentage of 1 responses/percentage of 0 responses)
dsc$items  #Frequency distribution of each pattern. 0 means that 103 individuals replied 0 to
```

```
##        0  1  2  3   4
## Freq 103 33 37 65 141
```

```
#all the item, 1 means that 33 individual replied 1 to ONE ITEM but we don't know which one.
dsc$pw.ass  #The p-value corresponds to the chi-square test for the 2by2 table. We are testing
```

```
##   Item i Item j p.value
## 1      1      4  <2e-16
## 2      1      3  <2e-16
## 3      2      4  <2e-16
## 4      1      2  <2e-16
## 5      2      3  <2e-16
## 6      3      4  <2e-16
```

```
#independence between binary data. the association between 1 and 4 is significant. The fact
#that the item are associated makes sense to fit the model.
```

POINT 3

Use the function ltm to fit the 2 IRT model to the data. Comment the results.

We estimate a one latent variable model because in this case we have too few observed variables (4).

```
m1 <- ltm(Abortion ~ z1)
summary(m1)
```

```
##
## Call:
## ltm(formula = Abortion ~ z1)
##
## Model Summary:
##    log.Lik      AIC      BIC
##  -706.3369 1428.674 1460.174
##
## Coefficients:
##                 value std.err  z.vals
## Dffclt.Item 1  0.1697  0.0659  2.5749
## Dffclt.Item 2 -0.2362  0.0618 -3.8198
## Dffclt.Item 3 -0.3428  0.0671 -5.1062
## Dffclt.Item 4 -0.3165  0.0654 -4.8411
## Dscrmn.Item 1  4.4532  1.0245  4.3468
## Dscrmn.Item 2  4.3226  0.6824  6.3340
## Dscrmn.Item 3  5.6639  0.9957  5.6883
## Dscrmn.Item 4  3.6254  0.5570  6.5087
##
## Integration:
## method: Gauss-Hermite
## quadrature points: 21
##
## Optimization:
## Convergence: 0
## max(|grad|): 0.0039
## quasi-Newton: BFGS
```

The first information that we have is the log-likelihood and the converges. Then we have the BIC and AIC that can be used in order to different model with different latent variables.

```
m1.rip <- ltm(Abortion ~ z1, IRT.param = FALSE)
summary(m1.rip)
```

```
##
## Call:
## ltm(formula = Abortion ~ z1, IRT.param = FALSE)
##
## Model Summary:
##    log.Lik      AIC      BIC
##  -706.3369 1428.674 1460.174
##
## Coefficients:
##                      value std.err  z.vals
## (Intercept).Item 1 -0.7555  0.2836 -2.6643
## (Intercept).Item 2  1.0211  0.2857  3.5738
## (Intercept).Item 3  1.9416  0.4197  4.6262
## (Intercept).Item 4  1.1474  0.2667  4.3021
## z1.Item 1           4.4532  1.0245  4.3468
## z1.Item 2           4.3226  0.6824  6.3340
## z1.Item 3           5.6639  0.9957  5.6883
## z1.Item 4           3.6254  0.5570  6.5087
##
## Integration:
## method: Gauss-Hermite
## quadrature points: 21
##
## Optimization:
## Convergence: 0
## max(|grad|): 0.0039
## quasi-Newton: BFGS
```

Log-likelihood, BIC and AIC don't change.

Verify the relation between the parameter estimates in the two parameterizations.

The relation between the two different parametrisation of the difficulty parameter is:

```
-summary(m1)$coefficients[1,1]*summary(m1)$coefficients[5,1]
```

```
## [1] -0.7555329
```

```
#Get the Latent trait parameter from the 2PL model that is equal to:
summary(m1.rip)$coefficients[1,1]
```

```
## [1] -0.7555329
```

POINT 4

Compute the standardized discriminant parameters and the probability of positive response of the median individual

```
alpha <- m1.rip$coeff[,2]
stalpha <- alpha/sqrt(1+alpha^2)
```

The highest standardize alpha correspond to item 3 that is the most discriminating parameter because it is the parameter with the highest value.

It is also possible to display the probabilities of positive response of the median individual. The option order=TRUE this means that the output is ordered with increasing probabilities.

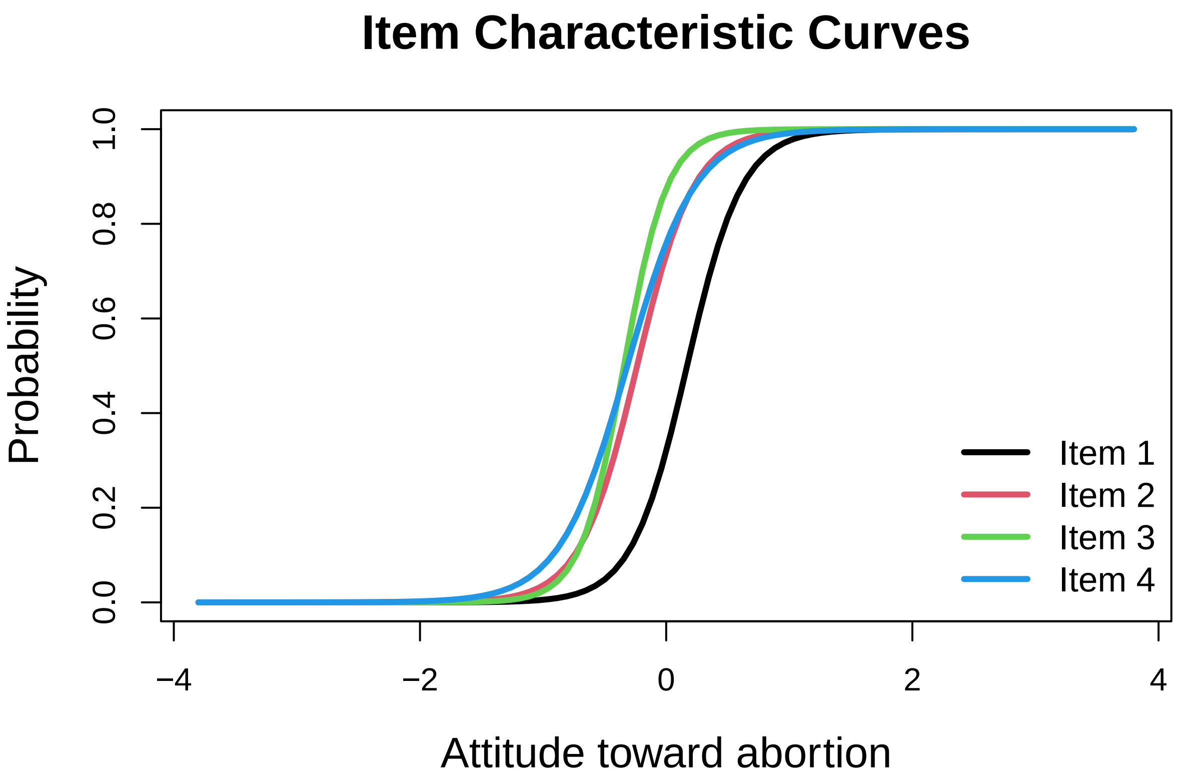
```
coef(m1.rip,prob=TRUE,order=TRUE)
```

```
##          (Intercept)        z1 P(x=1|z=0)
## Item 1  -0.7555329 4.453179  0.3196169
## Item 2   1.0211338 4.322628  0.7351934
## Item 4   1.1473662 3.625383  0.7590295
## Item 3   1.9416173 5.663928  0.8745297
```

POINT 6

Represent the characteristic curves of the four items for the estimated model. Comment the results.

```
plot(m1, legend = TRUE, cx = 'bottomright', xlab='Attitude toward abortion', lwd = 3,
     cex.main = 1.5, cex.lab = 1.3, cex = 1.1)
```



We have a different curve for each item. Item 1 is the most difficult, while the most discriminating item is item 3.

POINT 7 *Based on the obtained results evaluate if the model has a good fit to the data.* The statistics that we can compute are the chisquare and the LRT.

CHI-SQUARE

```
E <-fitted(m1)[,5] #Expected frequencies
O <- m1$patterns$obs #Observed frequencies
cbind(m1$pattern$X,O,E) #Verify if there are SPARSE DATA -- For patterns 9,11,12 the EXPECTED
```

```
##                 O           E
##  [1,] 0 0 0 0 103 103.4122369
##  [2,] 0 0 0 1  13  15.5202810
##  [3,] 0 0 1 0  10  11.1475316
##  [4,] 0 0 1 1  21  14.3156121
##  [5,] 0 1 0 0   9   8.9141331
##  [6,] 0 1 0 1   6   6.8099834
##  [7,] 0 1 1 0   7  12.0534014
##  [8,] 0 1 1 1  44  41.8185891
##  [9,] 1 0 0 0   1   1.3985312
## [10,] 1 0 1 1   6   7.1931090
## [11,] 1 1 0 0   3   0.8956743
## [12,] 1 1 0 1   3   2.5554663
## [13,] 1 1 1 0  12   7.1052145
## [14,] 1 1 1 1 141 142.7076472
```

```
#frequencies are very low, LOWER THAN 5, so we can have problems.
Chisq <- sum((E-O)*2/E)
DOF <- 14-2*4-1 #Are the same for both the tests. In order to compute it we need to know the
#possible response patterns which are 2^4=16. But since the observed response pattern are 14
#this means that we should take 14 as the value in the computation of the DOF.
#If theoretical pattern > observed pattern we should use the observed one.
pvalueC <- 1-pchisq(Chisq, DOF)
```

We don't reject if we consider an alpha of 0.01

LRT

```
LR <- 2*sum(O*log(O/E))
pvalueLR<-1-pchisq(LR,DOF)
pvalueLR
```

```
## [1] 0.002319567
```

We reject the null hypothesis unless we consider an alpha of 0.001

POINT 7 *Use the function margins to compute the values of the two-way and three-way chi-square residuals and evaluate if the model has a local goodness of fit to the data*

We can check if these possible rejection (we would reject for an alpha of 0.05) depends on the sparse data or on the fact that this model is not a good model for these data. We can check by looking at the residuals for each possible pair of items and each possible combination.

```r
margins(m1) #two-way margins
```

```
##
## Call:
## ltm(formula = Abortion ~ z1)
##
## Fit on the Two-Way Margins
##
## Response: (0,0)
##   Item i Item j Obs    Exp (O-E)^2/E
## 1      1      4 129 135.53      0.31
## 2      2      3 117 121.46      0.16
## 3      2      4 114 117.99      0.13
##
## Response: (1,0)
##   Item i Item j Obs   Exp (O-E)^2/E
## 1      1      2   7 11.74      1.92
## 2      1      4  16 11.43      1.83
## 3      3      4  29 32.33      0.34
##
## Response: (0,1)
##   Item i Item j Obs   Exp (O-E)^2/E
## 1      3      4  22 26.01      0.62
## 2      1      4  84 78.46      0.39
## 3      1      2  66 69.60      0.19
##
## Response: (1,1)
##   Item i Item j Obs    Exp (O-E)^2/E
## 1      1      2 159 153.26      0.21
## 2      3      4 212 206.03      0.17
## 3      1      4 150 153.58      0.08
```

```r
margins(m1,type='three-way',nprint=2) #three-way=compare triplets of items
```

```
##
## Call:
## ltm(formula = Abortion ~ z1)
##
## Fit on the Three-Way Margins
##
## Response: (0,0,0)
##   Item i Item j Item k Obs    Exp (O-E)^2/E
## 1      1      2      3 116 118.93      0.07
## 2      1      2      4 113 114.56      0.02
##
## Response: (1,0,0)
##   Item i Item j Item k Obs  Exp (O-E)^2/E
## 1      1      2      4   1 3.43      1.72
## 2      1      3      4   4 2.29      1.27
##
## Response: (0,1,0)
##   Item i Item j Item k Obs   Exp (O-E)^2/E
## 1      1      3      4  17 23.20      1.66
## 2      1      2      4  16 20.97      1.18
##
## Response: (1,1,0)
##   Item i Item j Item k Obs  Exp (O-E)^2/E
## 1      1      2      4  15 8.00      6.12 ***
## 2      1      2      3   6 3.45      1.88
##
## Response: (0,0,1)
##   Item i Item j Item k Obs   Exp (O-E)^2/E
## 1      1      2      3  31 25.46       1.2
## 2      2      3      4  13 16.65       0.8
##
## Response: (1,0,1)
##   Item i Item j Item k Obs  Exp (O-E)^2/E
## 1      1      2      3   6 9.22      1.12
## 2      1      2      4   6 8.32      0.65
##
## Response: (0,1,1)
##   Item i Item j Item k Obs   Exp (O-E)^2/E
## 1      2      3      4  27 21.51       1.4
## 2      1      3      4  65 56.13       1.4
##
## Response: (1,1,1)
##   Item i Item j Item k Obs    Exp (O-E)^2/E
## 1      1      2      3 153 149.81      0.07
## 2      1      3      4 147 149.90      0.06
##
## '***' denotes a chi-squared residual greater than 3.5
```

```
margins(m1,type='three-way', nprint=2)
```

```
##
## Call:
## ltm(formula = Abortion ~ z1)
##
## Fit on the Three-Way Margins
##
## Response: (0,0,0)
##   Item i Item j Item k Obs    Exp (O-E)^2/E
## 1      1      2      3 116 118.93      0.07
## 2      1      2      4 113 114.56      0.02
##
## Response: (1,0,0)
##   Item i Item j Item k Obs  Exp (O-E)^2/E
## 1      1      2      4   1 3.43      1.72
## 2      1      3      4   4 2.29      1.27
##
## Response: (0,1,0)
##   Item i Item j Item k Obs   Exp (O-E)^2/E
## 1      1      3      4  17 23.20      1.66
## 2      1      2      4  16 20.97      1.18
##
## Response: (1,1,0)
##   Item i Item j Item k Obs  Exp (O-E)^2/E
## 1      1      2      4  15 8.00      6.12 ***
## 2      1      2      3   6 3.45      1.88
##
## Response: (0,0,1)
##   Item i Item j Item k Obs   Exp (O-E)^2/E
## 1      1      2      3  31 25.46       1.2
## 2      2      3      4  13 16.65       0.8
##
## Response: (1,0,1)
##   Item i Item j Item k Obs  Exp (O-E)^2/E
## 1      1      2      3   6 9.22      1.12
## 2      1      2      4   6 8.32      0.65
##
## Response: (0,1,1)
##   Item i Item j Item k Obs   Exp (O-E)^2/E
## 1      2      3      4  27 21.51       1.4
## 2      1      3      4  65 56.13       1.4
##
## Response: (1,1,1)
##   Item i Item j Item k Obs    Exp (O-E)^2/E
## 1      1      2      3 153 149.81      0.07
## 2      1      3      4 147 149.90      0.06
##
## '***' denotes a chi-squared residual greater than 3.5
```

The rule of thumb is that the residuals aren't good if they are greater than 3/4 depending on how we want to be restrictive. In this case for every possible couple of items we get very low residuals so the model is a good model and we've rejected only because of sparse data (or otherwise we can add another latent variable). Typically 2 or 3 residuals that are greater than 3/4 may represent a problem in the model.

POINT 9 *Use the function factor.scores to estimate the values of the latent variable for each individual.*

```
fs<-factor.scores(m1,method='EAP')
fs
```

```
##
## Call:
## ltm(formula = Abortion ~ z1)
##
## Scoring Method: Expected A Posteriori
##
## Factor-Scores for observed response patterns:
##    Item 1 Item 2 Item 3 Item 4 Obs     Exp     z1 se.z1
## 1       0      0      0      0 103 103.412 -1.163 0.550
## 2       0      0      0      1  13  15.520 -0.643 0.285
## 3       0      0      1      0  10  11.148 -0.449 0.337
## 4       0      0      1      1  21  14.316 -0.085 0.244
## 5       0      1      0      0   9   8.914 -0.585 0.297
## 6       0      1      0      1   6   6.810 -0.188 0.310
## 7       0      1      1      0   7  12.053 -0.048 0.214
## 8       0      1      1      1  44  41.819  0.121 0.280
## 9       1      0      0      0   1   1.399 -0.573 0.301
## 10      1      0      1      1   6   7.193  0.131 0.288
## 11       1      1      0      0   3   0.896 -0.118 0.270
## 12       1      1      0      1   3   2.555  0.047 0.216
## 13       1      1      1      0  12   7.105  0.199 0.334
## 14       1      1      1      1 141 142.708  0.961 0.602
```

We have the values of the factor scores for each indiviual.

```
Comp <-factor.scores(m1,method='Component')
Comp
```

```
##
## Call:
## ltm(formula = Abortion ~ z1)
##
## Scoring Method: Component
##
## Factor-Scores for observed response patterns:
##    Item 1 Item 2 Item 3 Item 4 Obs     Exp     z1
## 1       0      0      0      0 103 103.412  0.000
## 2       0      0      0      1  13  15.520  3.625
## 3       0      0      1      0  10  11.148  5.664
## 4       0      0      1      1  21  14.316  9.289
## 5       0      1      0      0   9   8.914  4.323
## 6       0      1      0      1   6   6.810  7.948
## 7       0      1      1      0   7  12.053  9.987
## 8       0      1      1      1  44  41.819 13.612
## 9       1      0      0      0   1   1.399  4.453
## 10      1      0      1      1   6   7.193 13.742
## 11      1      1      0      0   3   0.896  8.776
## 12      1      1      0      1   3   2.555 12.401
## 13      1      1      1      0  12   7.105 14.440
## 14      1      1      1      1 141 142.708 18.065
```

Here we compute the total scores. The total score is the sum of the rows for each response pattern.

```
resp.pattern <- fs$score.dat[,1:4]
total.score <- apply(resp.pattern, 1, sum)
total.score
```

```
##  [1] 0 1 1 2 1 2 2 3 1 3 2 3 3 4
```

```
Cp <- Comp$score.dat[,7]
tab <- cbind(fs$score.dat,Cp,total.score)
round(tab,3)
```

```
##    Item 1 Item 2 Item 3 Item 4 Obs     Exp     z1 se.z1     Cp total.score
## 1       0      0      0      0 103 103.412 -1.163 0.550  0.000           0
## 2       0      0      0      1  13  15.520 -0.643 0.285  3.625           1
## 3       0      0      1      0  10  11.148 -0.449 0.337  5.664           1
## 4       0      0      1      1  21  14.316 -0.085 0.244  9.289           2
## 5       0      1      0      0   9   8.914 -0.585 0.297  4.323           1
## 6       0      1      0      1   6   6.810 -0.188 0.310  7.948           2
## 7        0      1      1      0   7  12.053 -0.048 0.214  9.987           2
## 8        0      1      1      1  44  41.819  0.121 0.280 13.612           3
## 9        1      0      0      0   1   1.399 -0.573 0.301  4.453           1
## 10       1      0      1      1   6   7.193  0.131 0.288 13.742           3
## 11       1      1      0      0   3   0.896 -0.118 0.270  8.776           2
## 12       1      1      0      1   3   2.555  0.047 0.216 12.401           3
## 13       1      1      1      0  12   7.105  0.199 0.334 14.440           3
## 14       1      1      1      1 141 142.708  0.961 0.602 18.065           4
```

```
round(tab[order(Cp),],3)
```

```
##    Item 1 Item 2 Item 3 Item 4 Obs     Exp     z1 se.z1     Cp total.score
## 1       0      0      0      0 103 103.412 -1.163 0.550  0.000           0
## 2       0      0      0      1  13  15.520 -0.643 0.285  3.625           1
## 5       0      1      0      0   9   8.914 -0.585 0.297  4.323           1
## 9       1      0      0      0   1   1.399 -0.573 0.301  4.453           1
## 3       0      0      1      0  10  11.148 -0.449 0.337  5.664           1
## 6       0      1      0      1   6   6.810 -0.188 0.310  7.948           2
## 11      1      1      0      0   3   0.896 -0.118 0.270  8.776           2
## 4       0      0      1      1  21  14.316 -0.085 0.244  9.289           2
## 7       0      1      1      0   7  12.053 -0.048 0.214  9.987           2
## 12      1      1      0      1   3   2.555  0.047 0.216 12.401           3
## 8       0      1      1      1  44  41.819  0.121 0.280 13.612           3
## 10      1      0      1      1   6   7.193  0.131 0.288 13.742           3
## 13      1      1      1      0  12   7.105  0.199 0.334 14.440           3
## 14      1      1      1      1 141 142.708  0.961 0.602 18.065           4
```

The lowest score according to each method (IAP, Component, Total score) corresponds to the 103 individuals that replied 000 to the three items. If you use the IAP or the component method the result is the same in latent trait model, but this is not necessary true for total score method.

POINT 10 *Estimate the abilities of the response pattern that do not occur in the data set.* The last thing to do is computing the values of the ability for the patterns that we haven't observed (1-0-0-1 and 1-0-1-0). We get the expected frequencies of these patterns.

```
fitted(m1,resp.pattern = rbind(c(1,0,0,1),c(1,0,1,0)))
```

```
##      Item 1 Item 2 Item 3 Item 4      Exp
## [1,]      1      0      0      1 1.125171
## [2,]      1      0      1      0 2.027418
```