MFR

李峰

2017年9月13日

TAM的手册介绍了多面RASCH的几个例子（pp.128），其中，EXAMPLE 9这个例子用的数据是data.sim.mfr和data.sim.facets。

1. 数据结构

例子中所用数据，是一个100行5列的数据，作为resp，显示有100位受评者。评价者的信息储存在另一个facets数据中，也是100行，和resp数据对应，有三列变量，分别是评价者的ID、评价者的性别和评价主题。

resp数据的前十行如下：

```
##      [,1] [,2] [,3] [,4] [,5]
## V1      3    3    3    3    1
## V1.1    2    3    2    3    2
## V1.2    1    3    0    3    0
## V1.3    1    3    3    3    0
## V1.4    0    1    3    2    1
## V1.5    1    3    1    3    0
## V1.6    0    1    3    3    0
## V1.7    1    3    3    3    2
## V1.8    0    0    3    3    1
## V1.9    0    0    3    3    3
```

每个题目是4计分，最小是0，最大是3。

facets数据的前十行如下：

```
##      rater topic female
## V1       1     3      2
## V1.1     2     1      2
## V1.2     3     3      1
## V1.3     4     1      2
## V1.4     5     3      1
## V1.5     1     2      1
## V1.6     2     3      2
## V1.7     3     2      1
## V1.8     4     2      2
## V1.9     5     1      1
```

我们可以看到，共有5个评价者，每个评价者评价的次数如下：

```
##
##  1  2  3  4  5
## 20 20 20 20 20
```

换言之，这个评分数据，是五个评价者每人在五个项目上评价了20个人，每个受评人只被评价了一次。

2. 把数据作为普通测验数据来看

如果我们把这个评价数据作为partial credit模型来进行估计，即当作一般的测验数据来对待（GPCM,在CONQUEST里，模型为“item+item*step”），我们可以看到结果里包含以下内容：

```
mod6_1 <- TAM::tam.mml( resp= data.sim.mfr, irtmodel="PCM2")
```

```
##  [1] "xsi"                "beta"               "variance"
##  [4] "item"               "person"             "pid"
##  [7] "EAP.rel"            "post"               "rprobs"
## [10] "itemweight"         "theta"              "n.ik"
## [13] "pi.k"               "Y"                  "resp"
## [16] "resp.ind"           "group"              "G"
## [19] "groups"             "formulaY"           "dataY"
## [22] "pweights"           "time"               "A"
## [25] "B"                  "se.B"               "nitems"
## [28] "maxK"               "AXsi"               "AXsi_"
## [31] "se.AXsi"            "nstud"              "resp.ind.list"
## [34] "hwt"                "like"               "ndim"
## [37] "xsi.fixed"          "xsi.fixed.estimated" "B.fixed.estimated"
## [40] "beta.fixed"         "Q"                  "variance.fixed"
## [43] "nnodes"             "deviance"           "ic"
## [46] "deviance.history"   "control"            "irtmodel"
## [49] "iter"               "printxsi"           "YSD"
## [52] "CALL"               "prior_list_xsi"     "penalty_xsi"
```

2.1 δ和τ的值在*xsi*里

PCM的*xsi*有15个值，前五个是项目的δ的值，后面十个分别是五个题目在step1和step2的τ的值，这里只给了τ_1和τ_2的值，而τ_3等于$0 - \tau_1 - \tau_2$，事实上也是给了的。

```
mod6_1$xsi
```

```
##                 xsi    se.xsi
## I1         1.15617044 0.1353443
## I2        -0.60806022 0.1116199
## I3        -0.54537932 0.1092631
## I4        -1.41823662 0.1463187
## I5         1.48246707 0.1450160
## I1_step1 -0.66185140 0.2168303
## I1_step2 -0.01181249 0.3058414
## I2_step1 -0.06367339 0.2269656
## I2_step2  0.61043625 0.2999946
## I3_step1  0.62041568 0.2368003
## I3_step2 -0.12568176 0.2832351
## I4_step1  0.47663008 0.2455892
## I4_step2 -0.54304436 0.2610788
## I5_step1 -0.37890977 0.2314136
## I5_step2 -0.25734152 0.3414570
```

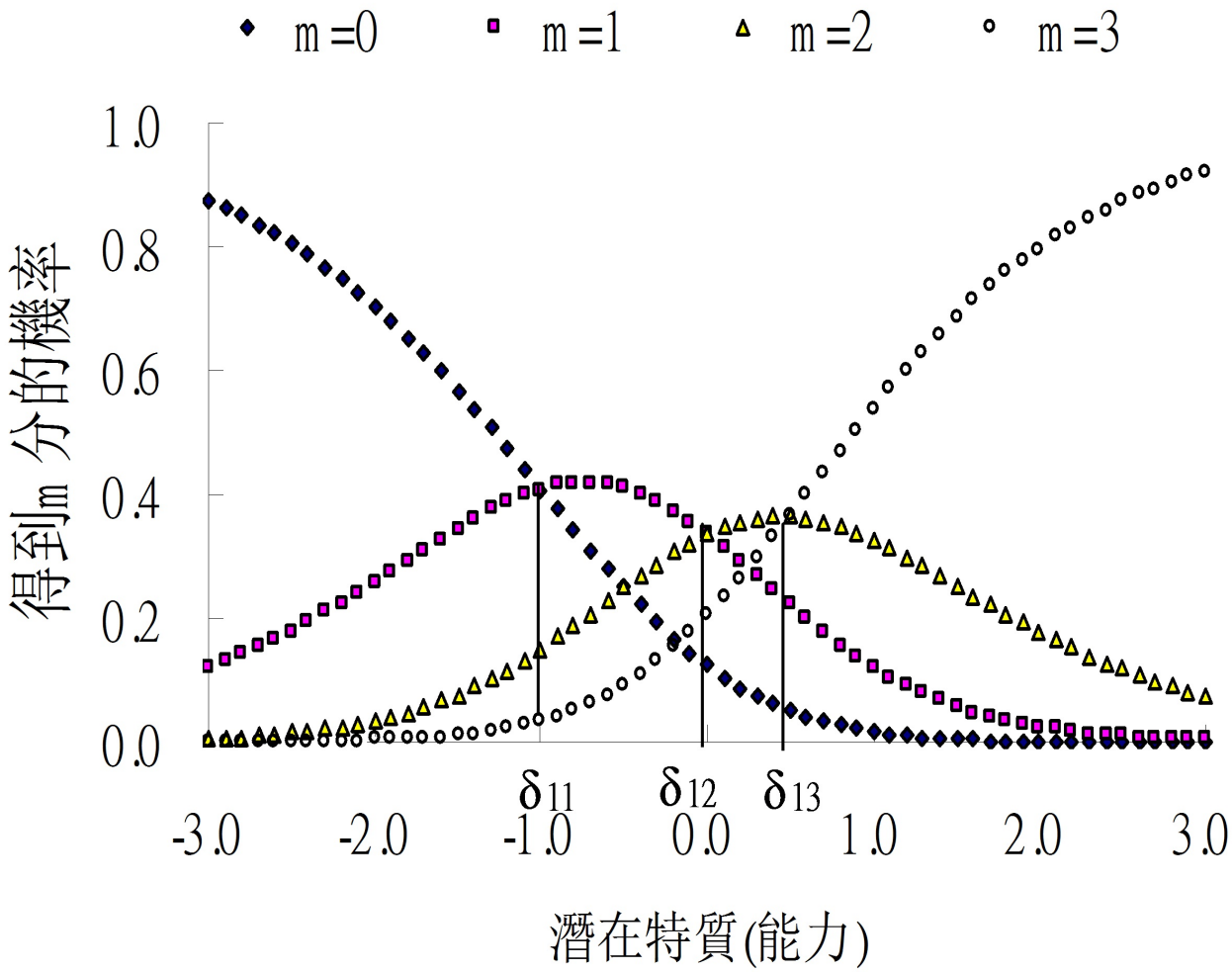
```
##            [,1]        [,2]        [,3]
## [1,]  1.1561704 -0.66185140 -0.01181249
## [2,] -0.6080602 -0.06367339  0.61043625
## [3,] -0.5453793  0.62041568 -0.12568176
## [4,] -1.4182366  0.47663008 -0.54304436
## [5,]  1.4824671 -0.37890977 -0.25734152
```

上面结果中，第一列是δ，第二、三列是τ_1和τ_2。如果补充τ_3的值，应该是这样：

```
##             [,1]        [,2]        [,3]        [,4]
## [1,]  1.1561704 -0.66185140 -0.01181249  0.67366388
## [2,] -0.6080602 -0.06367339  0.61043625 -0.54676286
## [3,] -0.5453793  0.62041568 -0.12568176 -0.49473392
## [4,] -1.4182366  0.47663008 -0.54304436  0.06641429
## [5,]  1.4824671 -0.37890977 -0.25734152  0.63625129
```

2.3 δ_i的值可以转换得到

当然，这还不是对PCM来说的δ，对四点计分的题目，PCM有四条ICC，会产生三个δ的值，即四条ICC的交叉点，如图：



公式如下：

$$p_0 = \Pr(X=0) = \frac{1}{1+\exp(\theta-\delta_1)+\exp(2\theta-(\delta_1+\delta_2))}$$

$$p_1 = \Pr(X=1) = \frac{\exp(\theta-\delta_1)}{1+\exp(\theta-\delta_1)+\exp(2\theta-(\delta_1+\delta_2))}$$

$$p_2 = \Pr(X=2) = \frac{\exp(2\theta-(\delta_1+\delta_2))}{1+\exp(\theta-\delta_1)+\exp(2\theta-(\delta_1+\delta_2))}$$

基于*xs*的数据，可以得到δ_1、δ_2和δ_3的值，只要用$\delta+\tau_i$即可。

```
##              [,1]        [,2]      [,3]
## [1,]  0.49431904  1.14435796  1.829834
## [2,] -0.67173361  0.00237603 -1.154823
## [3,]  0.07503635 -0.67106108 -1.040113
## [4,] -0.94160655 -1.96128099 -1.351822
## [5,]  1.10355730  1.22512555  2.118718
```

2.3 $\Sigma\delta_i$值在*AXsi_*里

需要注意的是，结果中其实包含了δ_1、$\delta_1 + \delta_2$和$\delta_1 + \delta_2 + \delta_3$的值，即公式里$\theta$、$2\theta$、$3\theta$要减去的部分，注意下面结果中AXsi_.Cat1、AXsi_.Cat2和AXsi_.Cat3的值。

```
summary(mod6_1)
```

```
## ------------------------------------------------------------
## TAM 2.6-2 (2017-08-10 11:07:46)
## R version 3.4.1 (2017-06-30) x86_64, mingw32 | nodename = PC201508281246 | login = Administrator
##
## Date of Analysis: 2017-09-18 02:09:18
## Time difference of 0.2601829 secs
## Computation time: 0.2601829
##
## Multidimensional Item Response Model in TAM
##
## IRT Model PCM2
##
## Call:
## TAM::tam.mml(resp = data.sim.mfr, irtmodel = "PCM2")
##
## ------------------------------------------------------------
## Number of iterations = 56
## Numeric integration with 21 integration points
##
## Deviance =  1005.01
##    Log likelihood =  -502.5
## Number of persons =  100
## Number of persons used =  100
## Number of items =  5
## Number of estimated parameters =  16
##     Item threshold parameters  =  15
##     Item slope parameters      =  0
##     Regression parameters      =  0
##     (Co)Variance parameters    =  1
##
## AIC  =  1037  | penalty = 32    | AIC = -2*LL + 2*p
## AICc =  1044  | penalty = 38.55    | AICc = -2*LL + 2*p + 2*p*(p+1)/(n-p-1)  (bias corrected AIC)
## BIC  =  1079  | penalty = 73.68    | BIC = -2*LL + log(n)*p
## aBIC  =  1028  | penalty = 22.51    | aBIC = -2*LL + log((n-2)/24)*p  (adjusted BIC)
## CAIC =  1095  | penalty = 89.68   | CAIC = -2*LL + [log(n)+1]*p  (consistent AIC)
##
## ------------------------------------------------------------
## EAP Reliability
## [1] 0.723
## ------------------------------------------------------------
## Covariances and Variances
##       V1
## V1 0.862
## ------------------------------------------------------------
## Correlations and Standard Deviations (in the diagonal)
##       V1
## V1 0.929
## ------------------------------------------------------------
## Regression Coefficients
##      V1
## [1,]  0
## ------------------------------------------------------------
## Item Parameters -A*Xsi
##   item   N    M xsi.item AXsi_.Cat1 AXsi_.Cat2 AXsi_.Cat3 B.Cat1.Dim1
## 1   I1 100 0.74    1.156      0.494      1.639      3.469           1
## 2   I2 100 1.99   -0.608     -0.672     -0.669     -1.824           1
## 3   I3 100 1.99   -0.545      0.075     -0.596     -1.636           1
```

```
## 4   I4 100 2.53   -1.418     -0.942     -2.903     -4.255           1
## 5   I5 100 0.55    1.482      1.104      2.329      4.447           1
##   B.Cat2.Dim1 B.Cat3.Dim1
## 1           2           3
## 2           2           3
## 3           2           3
## 4           2           3
## 5           2           3
##
## Item Parameters Xsi
##              xsi se.xsi
## I1         1.156  0.135
## I2        -0.608  0.112
## I3        -0.545  0.109
## I4        -1.418  0.146
## I5         1.482  0.145
## I1_step1 -0.662  0.217
## I1_step2 -0.012  0.306
## I2_step1 -0.064  0.227
## I2_step2  0.610  0.300
## I3_step1  0.620  0.237
## I3_step2 -0.126  0.283
## I4_step1  0.477  0.246
## I4_step2 -0.543  0.261
## I5_step1 -0.379  0.231
## I5_step2 -0.257  0.341
```

这些参数出现在mod6_1$item里。

```
mod6_1$item
```

```
##   item   N    M   xsi.item  AXsi_.Cat1 AXsi_.Cat2 AXsi_.Cat3 B.Cat1.Dim1
## 1   I1 100 0.74  1.1561718  0.49431322  1.6386724   3.468515           1
## 2   I2 100 1.99 -0.6080646 -0.67174012 -0.6693670  -1.824194           1
## 3   I3 100 1.99 -0.5453840  0.07502925 -0.5960349  -1.636152           1
## 4   I4 100 2.53 -1.4182460 -0.94170536 -2.9029828  -4.254738           1
## 5   I5 100 0.55  1.4824688  1.10354934  2.3286702   4.447406           1
##   B.Cat2.Dim1 B.Cat3.Dim1
## 1           2           3
## 2           2           3
## 3           2           3
## 4           2           3
## 5           2           3
```

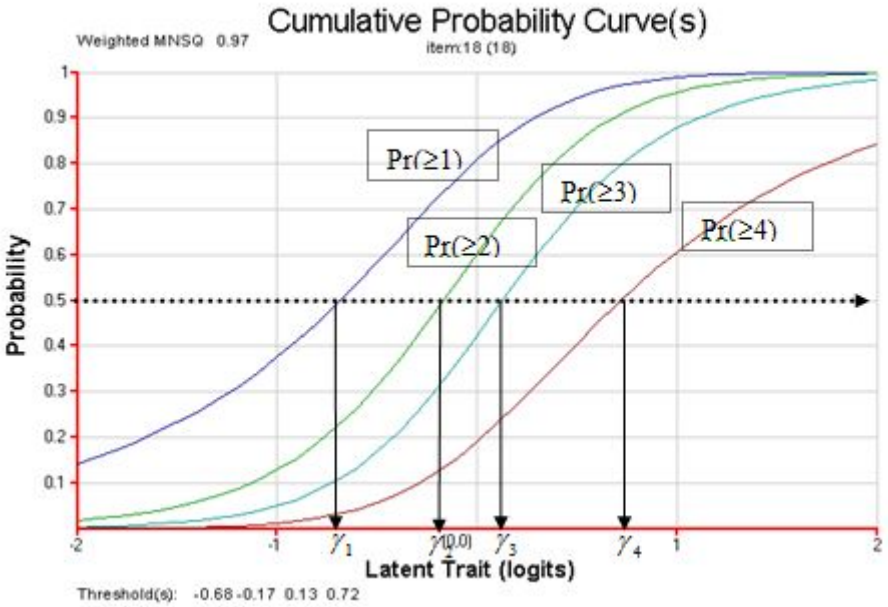
也出现在mod6_1$AXsi_里。

```
mod6_1$AXsi_
```

```
##      [,1]        [,2]       [,3]      [,4]
## [1,]    0  0.49431322  1.6386724  3.468515
## [2,]    0 -0.67174012 -0.6693670 -1.824194
## [3,]    0  0.07502925 -0.5960349 -1.636152
## [4,]    0 -0.94170536 -2.9029828 -4.254738
## [5,]    0  1.10354934  2.3286702  4.447406
```

2.4 thresholds的值需要计算一下

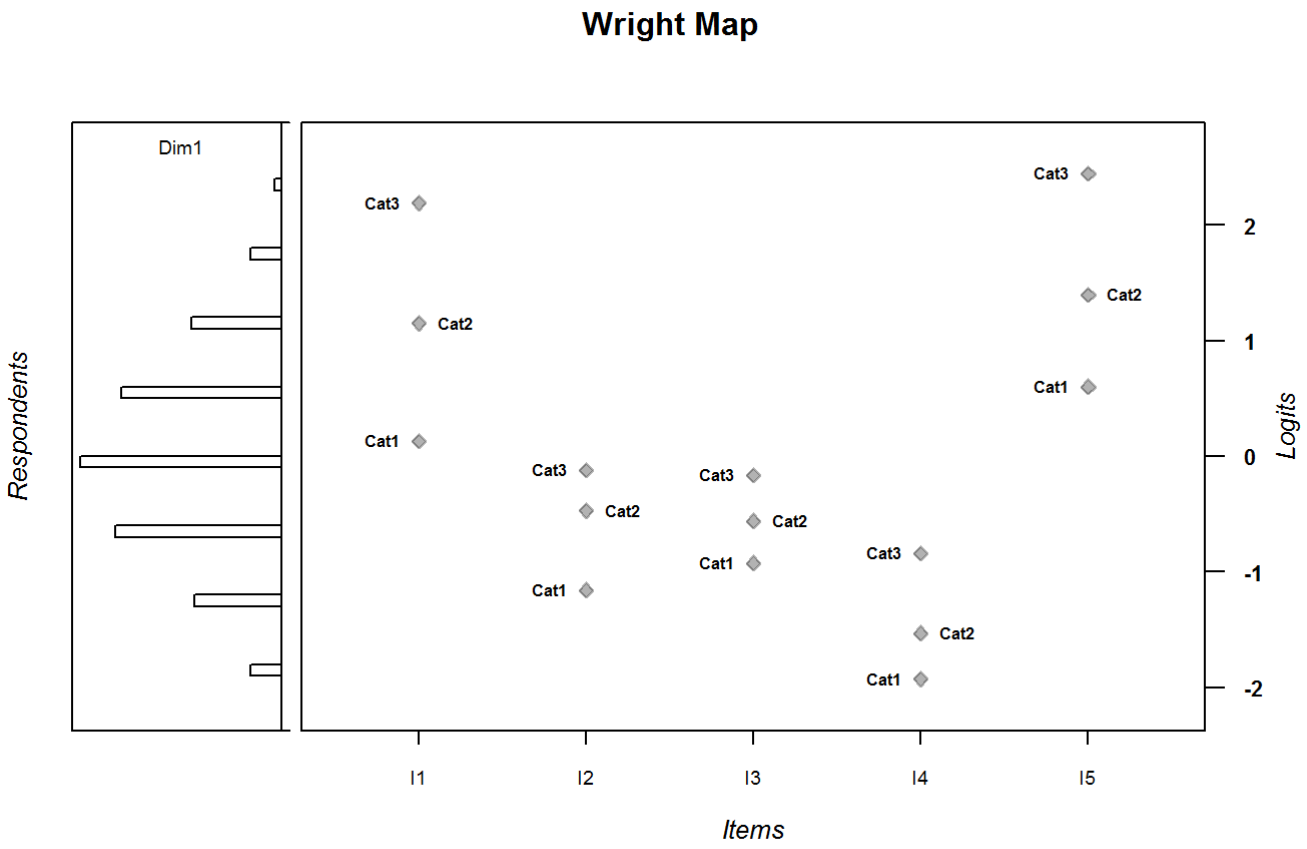
δ_i并不是p=0.5时候的阈值（Thurstonian thresholds），需要一个函数*irt.threshold*来计算Thurstonian thresholds，也即γ_i的值。




```
thresh1 <- TAM::IRT.threshold(mod6_1)
print(thresh1)
```

```
##          Cat1       Cat2       Cat3
## I1  0.1310742  1.1486090  2.1885924
## I2 -1.1596634 -0.4732507 -0.1241138
## I3 -0.9256880 -0.5611936 -0.1644446
## I4 -1.9284418 -1.5326427 -0.8435226
## I5  0.6008895  1.3954731  2.4416717
```

```
IRT.WrightMap(thresh1)
```



2.5 题目的拟合度检验

这个部分的结果就和CONQUEST的结果呈现比较接近了。

```
wle <- TAM::tam.wle(mod6_1)
```

```
## Iteration in WLE/MLE estimation  1   | Maximal change  1.4831
## Iteration in WLE/MLE estimation  2   | Maximal change  0.5121
## Iteration in WLE/MLE estimation  3   | Maximal change  0.1059
## Iteration in WLE/MLE estimation  4   | Maximal change  0.0043
## Iteration in WLE/MLE estimation  5   | Maximal change  9e-04
## Iteration in WLE/MLE estimation  6   | Maximal change  2e-04
## Iteration in WLE/MLE estimation  7   | Maximal change  0
## ----
##  WLE Reliability = 0.681
```

```
# extract item parameters
b4<- - mod6_1$AXsi[ , -1 ]
data.sim.mfr1<-data.frame(data.sim.mfr)
resp <- data.sim.mfr1[ , c("X1","X2","X3","X4","X5") ]
fitla <- sirt::pcm.fit(b=b4 , theta=wle$theta , resp)
fitla$itemfit
```

```
##    item    outfit   outfit.t     infit    infit.t
## X1   X1 0.6292863 -2.1248805 0.6658028 -2.4697790
## X2   X2 0.5423456 -1.9496920 0.6897081 -2.5015375
## X3   X3 0.6761667 -1.1970944 0.8367440 -1.1842274
## X4   X4 0.8192256 -0.5033304 0.6773597 -1.8279278
## X5   X5 0.9309634 -0.1923768 0.8376612 -0.9712908
```

3. 把数据作为评价数据来看

EXAMPLE 9这个例子里介绍了两个模型：

- two way interaction item and rater的，其formulaA是 ~item+item:step + item*rater
- three way interaction item, female and rater，其fromulaA是 ~item+item:step + female*rater* + *female*item*step

比较来看，后者关注评价者在性别上是否有差异。

3.1 选择one way interaction的模型

如果我们不关心item和rater的交互作用，我们可以修改formular，去掉里面的item*rater，保留rater。

```
formulaA <- ~item+item:step +rater
```

```
mod9a <- TAM::tam.mml.mfr( resp=data.sim.mfr, facets=data.sim.facets, formulaA=formulaA)
```

对参数进行估计后，会出现如下结果：

```
##  [1] "xsi"                "xsi.facets"         "beta"
##  [4] "variance"           "item"               "person"
##  [7] "pid"                "EAP.rel"            "post"
## [10] "rprobs"             "itemweight"         "theta"
## [13] "n.ik"               "pi.k"               "Y"
## [16] "resp"               "resp.ind"           "group"
## [19] "G"                  "groups"             "formulaY"
## [22] "dataY"              "pweights"           "time"
## [25] "A"                  "B"                  "se.B"
## [28] "nitems"             "maxK"               "AXsi"
## [31] "AXsi_"              "se.AXsi"            "nstud"
## [34] "resp.ind.list"      "hwt"                "like"
## [37] "ndim"               "xsi.fixed"          "xsi.fixed.estimated"
## [40] "B.fixed.estimated"  "beta.fixed"         "Q"
## [43] "formulaA"           "facets"             "xsi.constr"
## [46] "variance.fixed"     "nnodes"             "deviance"
## [49] "ic"                 "deviance.history"   "control"
## [52] "irtmodel"           "iter"               "resp_orig"
## [55] "printxsi"           "YSD"                "PSF"
## [58] "CALL"               "prior_list_xsi"     "penalty_xsi"
```

3.2 δ和τ的值仍在*xsi*里

和一般测验数据分析比较，结果的第一项都是*xsi*，但是多面的模型里多了rater的参数。**其δ和τ的值和不考虑侧面时基本一致。**

```
mod9a$xsi
```

```
##                  xsi     se.xsi
## I1         1.16008438 0.13579358
## I2        -0.61201293 0.11150281
## I3        -0.54967380 0.10912308
## I4        -1.41688612 0.14584335
## I5         1.48913339 0.14558727
## rater1    -0.05991844 0.08780488
## rater2    -0.31322609 0.08917051
## rater3     0.34287233 0.08683269
## rater4    -0.28033790 0.08836054
## I1:step1  -0.67053164 0.21684766
## I2:step1  -0.06049310 0.22703725
## I3:step1   0.62323487 0.23685245
## I4:step1   0.48580970 0.24560616
## I5:step1  -0.38956322 0.23155711
## I1:step2  -0.01485097 0.30610362
## I2:step2   0.60730862 0.30001565
## I3:step2  -0.12862680 0.28326077
## I4:step2  -0.54684035 0.26109880
## I5:step2  -0.25965492 0.34190486
```

其δ和τ的值和不考虑侧面时基本一致，考虑侧面模型的δ和τ的值：

```
##            [,1]       [,2]        [,3]
## [1,]  1.1600844 -0.6705316 -0.01485097
## [2,] -0.6120129 -0.0604931  0.60730862
## [3,] -0.5496738  0.6232349 -0.12862680
## [4,] -1.4168861  0.4858097 -0.54684035
## [5,]  1.4891334 -0.3895632 -0.25965492
```

比较不考虑侧面模型时δ和τ的值：

```
##            [,1]        [,2]        [,3]
## [1,]  1.1561704 -0.66185140 -0.01181249
## [2,] -0.6080602 -0.06367339  0.61043625
## [3,] -0.5453793  0.62041568 -0.12568176
## [4,] -1.4182366  0.47663008 -0.54304436
## [5,]  1.4824671 -0.37890977 -0.25734152
```

最后一个τ_i的值不在*xsi*里，而是出现在*xsi.facets*中（请注意，有侧面的模型多了个*xsi.facets*），参数个数比*xsi*多了五个，多的这五个，就是补充完整的五个题目的τ_3的值。即完整的δ、τ_1、τ_2和τ_3的值是在*xsi.facets*中，当然，也包括rater的参数。

```
mod9a$xsi.facets
```

```
##    parameter     facet         xsi     se.xsi
## 1         I1      item  1.16008438 0.13579358
## 2         I2      item -0.61201293 0.11150281
## 3         I3      item -0.54967380 0.10912308
## 4         I4      item -1.41688612 0.14584335
## 5         I5      item  1.48913339 0.14558727
## 6     rater1     rater -0.05991844 0.08780488
## 7     rater2     rater -0.31322609 0.08917051
## 8     rater3     rater  0.34287233 0.08683269
## 9     rater4     rater -0.28033790 0.08836054
## 10    rater5     rater  0.31061010 0.17609253
## 11  I1:step1 item:step -0.67053164 0.21684766
## 12  I2:step1 item:step -0.06049310 0.22703725
## 13  I3:step1 item:step  0.62323487 0.23685245
## 14  I4:step1 item:step  0.48580970 0.24560616
## 15  I5:step1 item:step -0.38956322 0.23155711
## 16  I1:step2 item:step -0.01485097 0.30610362
## 17  I2:step2 item:step  0.60730862 0.30001565
## 18  I3:step2 item:step -0.12862680 0.28326077
## 19  I4:step2 item:step -0.54684035 0.26109880
## 20  I5:step2 item:step -0.25965492 0.34190486
## 21  I1:step3 item:step  0.68538261 0.37512975
## 22  I2:step3 item:step -0.54681552 0.37623836
## 23  I3:step3 item:step -0.49460807 0.36923670
## 24  I4:step3 item:step  0.06103064 0.35846195
## 25  I5:step3 item:step  0.64921814 0.41293780
```

不考虑rater的话，*xsi*的参数如下：

```
##            [,1]       [,2]        [,3]
## [1,]  1.1600844 -0.6705316 -0.01485097
## [2,] -0.6120129 -0.0604931  0.60730862
## [3,] -0.5496738  0.6232349 -0.12862680
## [4,] -1.4168861  0.4858097 -0.54684035
## [5,]  1.4891334 -0.3895632 -0.25965492
```

*xsi.facets*的参数如下：

```
##            [,1]       [,2]        [,3]        [,4]
## [1,]  1.1600844 -0.6705316 -0.01485097  0.68538261
## [2,] -0.6120129 -0.0604931  0.60730862 -0.54681552
## [3,] -0.5496738  0.6232349 -0.12862680 -0.49460807
## [4,] -1.4168861  0.4858097 -0.54684035  0.06103064
## [5,]  1.4891334 -0.3895632 -0.25965492  0.64921814
```

比较可知，*xsi.facets*的τ_3，表中第四列，和第二、三列相加等于0，符合GPCM中对τ的约束条件。

3.3 和不考虑rater侧面的模型类似，也可以得到δ_i的值

根据*xsi.facets*里的参数，用$\delta + \tau_i$同样可以得到δ_i的值。

```
##              [,1]         [,2]      [,3]
## [1,]  0.48955274  1.145233411  1.845467
## [2,] -0.67250604 -0.004704312 -1.158828
## [3,]  0.07356106 -0.678300597 -1.044282
## [4,] -0.93107641 -1.963726464 -1.355855
## [5,]  1.09957016  1.229478468  2.138352
```

比较下不考虑侧面时δ_i的值：

```
##              [,1]        [,2]      [,3]
## [1,]  0.49431904  1.14435796  1.829834
## [2,] -0.67173361  0.00237603 -1.154823
## [3,]  0.07503635 -0.67106108 -1.040113
## [4,] -0.94160655 -1.96128099 -1.351822
## [5,]  1.10355730  1.22512555  2.118718
```

3.4 $\Sigma\delta_i$值仍在*AXsi_*里，但是里面包含了评分者效应

这些参数出现在mod9a$item里：

```
mod9a$item
```

```
##         item  N    M   xsi.item  AXsi_.Cat1   AXsi_.Cat2 AXsi_.Cat3
## 1  I1-rater1 20 0.80  1.1001610  0.42964483  1.514961593  3.3004831
## 2  I1-rater2 20 1.15  0.8468580  0.17634181  1.008355564  2.5405741
## 3  I1-rater3 20 0.60  1.5029461  0.83242986  2.320531653  4.5088382
## 4  I1-rater4 20 0.80  0.8797466  0.20923038  1.074132711  2.6392398
## 5  I1-rater5 20 0.35  1.4706872  0.80017100  2.256013947  4.4120616
## 6  I2-rater1 20 2.15 -0.6719251 -0.73241287 -0.797030403 -2.0157753
## 7  I2-rater2 20 2.05 -0.9252281 -0.98571588 -1.303636431 -2.7756843
## 8  I2-rater3 20 1.95 -0.2691401 -0.32962784  0.008539658 -0.8074202
## 9  I2-rater4 20 2.30 -0.8923395 -0.95282731 -1.237859284 -2.6770186
## 10 I2-rater5 20 1.50 -0.3013989 -0.36188669 -0.055978048 -0.9041968
## 11 I3-rater1 20 2.00 -0.6095857  0.01365541 -0.724558640 -1.8287572
## 12 I3-rater2 20 2.10 -0.8628888 -0.23964761 -1.231164669 -2.5886663
## 13 I3-rater3 20 1.70 -0.2068007  0.41644044  0.081011420 -0.6204021
## 14 I3-rater4 20 2.25 -0.8300002 -0.20675903 -1.165387522 -2.4900006
## 15 I3-rater5 20 1.90 -0.2390596  0.38418158  0.016493714 -0.7171787
## 16 I4-rater1 20 2.40 -1.4767905 -0.99088226 -3.014524637 -4.4303715
## 17 I4-rater2 20 2.60 -1.7300935 -1.24418527 -3.521130666 -5.1902805
## 18 I4-rater3 20 2.30 -1.0740055 -0.58809723 -2.208954577 -3.2220164
## 19 I4-rater4 20 2.70 -1.6972049 -1.21129670 -3.455353519 -5.0916148
## 20 I4-rater5 20 2.65 -1.1062643 -0.62035608 -2.273472283 -3.3187929
## 21 I5-rater1 20 0.65  1.4292075  1.03966401  2.209226936  4.2876224
## 22 I5-rater2 20 0.85  1.1759045  0.78636100  1.702620907  3.5277134
## 23 I5-rater3 20 0.20  1.8319925  1.44244904  3.014796997  5.4959775
## 24 I5-rater4 20 0.60  1.2087930  0.81924957  1.768398055  3.6263791
## 25 I5-rater5 20 0.45  1.7997336  1.41019019  2.950279290  5.3992009
##    B.Cat1.Diml B.Cat2.Diml B.Cat3.Diml
## 1            1           2           3
## 2            1           2           3
## 3            1           2           3
## 4            1           2           3
## 5            1           2           3
## 6            1           2           3
## 7            1           2           3
## 8            1           2           3
## 9            1           2           3
## 10           1           2           3
## 11           1           2           3
## 12           1           2           3
## 13           1           2           3
## 14           1           2           3
## 15           1           2           3
## 16           1           2           3
## 17           1           2           3
## 18           1           2           3
## 19           1           2           3
## 20           1           2           3
## 21           1           2           3
## 22           1           2           3
## 23           1           2           3
## 24           1           2           3
## 25           1           2           3
```

也出现在mod9a$Axsi_里：

```
mod9a$AXsi_
```

```
##       [,1]        [,2]         [,3]       [,4]
##  [1,]    0  0.42964483  1.514961593  3.3004831
##  [2,]    0  0.17634181  1.008355564  2.5405741
##  [3,]    0  0.83242986  2.320531653  4.5088382
##  [4,]    0  0.20923038  1.074132711  2.6392398
##  [5,]    0  0.80017100  2.256013947  4.4120616
##  [6,]    0 -0.73241287 -0.797030403 -2.0157753
##  [7,]    0 -0.98571588 -1.303636431 -2.7756843
##  [8,]    0 -0.32962784  0.008539658 -0.8074202
##  [9,]    0 -0.95282731 -1.237859284 -2.6770186
## [10,]    0 -0.36188669 -0.055978048 -0.9041968
## [11,]    0  0.01365541 -0.724558640 -1.8287572
## [12,]    0 -0.23964761 -1.231164669 -2.5886663
## [13,]    0  0.41644044  0.081011420 -0.6204021
## [14,]    0 -0.20675903 -1.165387522 -2.4900006
## [15,]    0  0.38418158  0.016493714 -0.7171787
## [16,]    0 -0.99088226 -3.014524637 -4.4303715
## [17,]    0 -1.24418527 -3.521130666 -5.1902805
## [18,]    0 -0.58809723 -2.208954577 -3.2220164
## [19,]    0 -1.21129670 -3.455353519 -5.0916148
## [20,]    0 -0.62035608 -2.273472283 -3.3187929
## [21,]    0  1.03966401  2.209226936  4.2876224
## [22,]    0  0.78636100  1.702620907  3.5277134
## [23,]    0  1.44244904  3.014796997  5.4959775
## [24,]    0  0.81924957  1.768398055  3.6263791
## [25,]    0  1.41019019  2.950279290  5.3992009
```

需要注意的是，5个评分者，5个题目，每个题目有3个δ，所以，其实是一个5X5X3的表，即25X3的表。比如对第一个评分者而言，其宽严值为-0.0599，则：

- 第一个题目的δ为1.1600，其第一个计分点的τ_1为-0.6705，则δ_1为(-0.0599)+1.1600+(-0.6705)，即为0.4296，是公式里θ要减去的δ_1。只是这里增加了评分者的效应，换言之，评分者也被当作一个δ看。
- 第二个计分点的τ_2为-0.01485，则δ_2为(-0.0599)+1.1600+(-0.01485)，即为1.0853。而考虑了评分者效应的δ_1+δ_2，即0.4296+1.0853=1.5149，是$\Sigma\delta_i$的值，即公式里2θ要减去的部分。
- 第三个计分点的τ_3为0.6853，则δ_3为(-0.0599)+1.1600+0.6853，即为1.7855。而考虑了评分者效应的δ_1+δ_2+δ_3，即0.4296+1.0853+1.7855=3.3004，是$\Sigma\delta_i$的值，即公式里3θ要减去的部分。

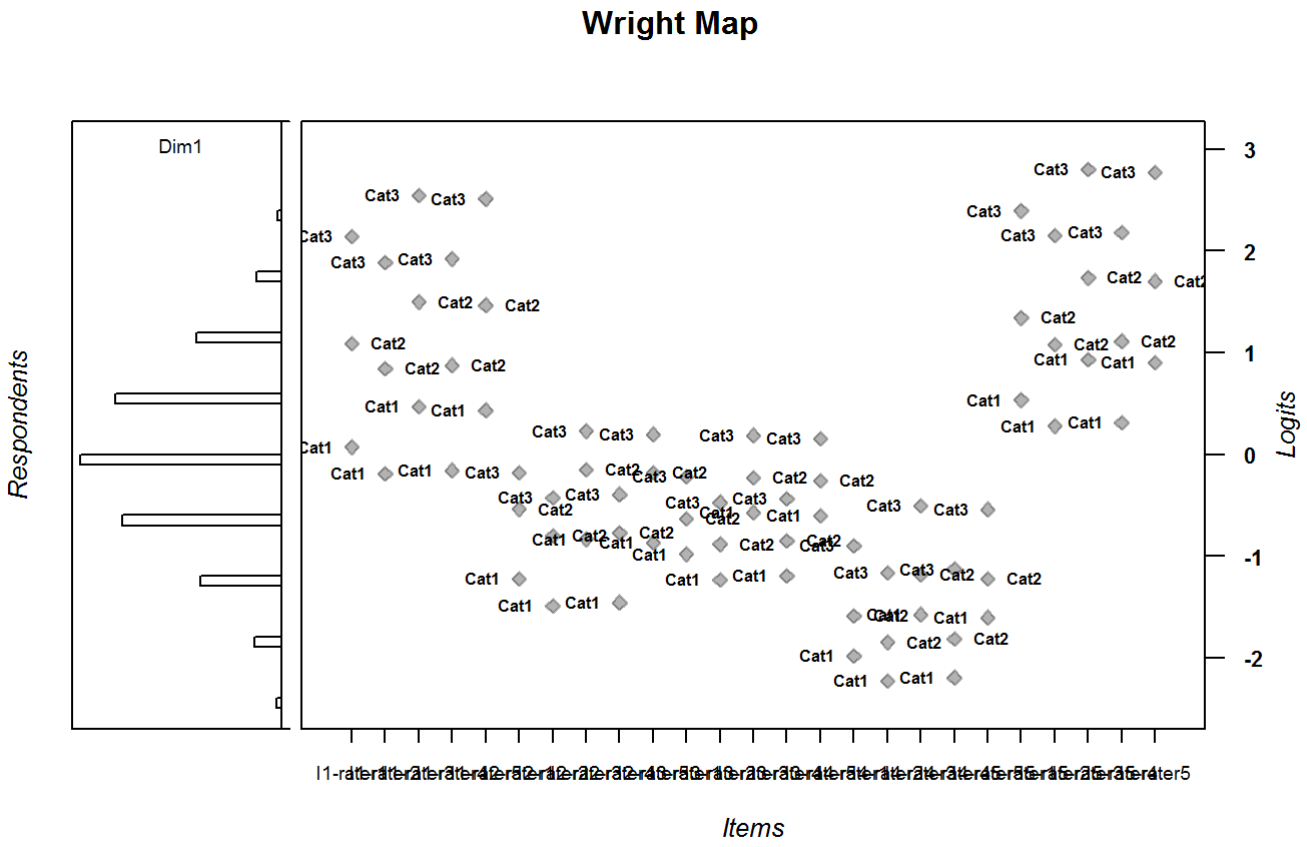
3.5 也可以得到*thresholds*的值

多面RASCH的thresholds的值不是以题目为单位，而是以raterXstep为单位的。

```
thresh2 <- TAM::IRT.threshold(mod9a)
print(thresh2)
```

```
##                  Cat1       Cat2       Cat3
## I1-rater1  0.07080288  1.0899429  2.1419344
## I1-rater2 -0.19157730  0.8432041  1.8862551
## I1-rater3  0.46608161  1.4972248  2.5451465
## I1-rater4 -0.15794442  0.8749092  1.9209853
## I1-rater5  0.43274285  1.4661868  2.5114762
## I2-rater1 -1.22389041 -0.5402646 -0.1847839
## I2-rater2 -1.49166580 -0.8044101 -0.4283356
## I2-rater3 -0.83614337 -0.1500137  0.2280771
## I2-rater4 -1.45931139 -0.7727230 -0.3949891
## I2-rater5 -0.86827839 -0.1815056  0.1959050
## I3-rater1 -0.98480249 -0.6341462 -0.2240908
## I3-rater2 -1.23375998 -0.8849118 -0.4733202
## I3-rater3 -0.57170910 -0.2327009  0.1857284
## I3-rater4 -1.19520413 -0.8543846 -0.4383308
## I3-rater5 -0.60532330 -0.2628127  0.1521383
## I4-rater1 -1.98609464 -1.5880887 -0.9029369
## I4-rater2 -2.22876245 -1.8512266 -1.1694553
## I4-rater3 -1.57636497 -1.1891718 -0.5062473
## I4-rater4 -2.19803085 -1.8152200 -1.1320490
## I4-rater5 -1.60648978 -1.2252684 -0.5422537
## I5-rater1  0.53378570  1.3414576  2.3934605
## I5-rater2  0.27597135  1.0782385  2.1498369
## I5-rater3  0.93063724  1.7353270  2.8039233
## I5-rater4  0.30791178  1.1114850  2.1816076
## I5-rater5  0.89908351  1.7022659  2.7728737
```

```
IRT.WrightMap(thresh2)
```



3.6 题目的拟合度检验

```
wle <- TAM::tam.wle(mod9a)
```

```
## Iteration in WLE/MLE estimation  1   | Maximal change  1.3002
## Iteration in WLE/MLE estimation  2   | Maximal change  0.5626
## Iteration in WLE/MLE estimation  3   | Maximal change  0.0563
## Iteration in WLE/MLE estimation  4   | Maximal change  0.0099
## Iteration in WLE/MLE estimation  5   | Maximal change  0.002
## Iteration in WLE/MLE estimation  6   | Maximal change  4e-04
## Iteration in WLE/MLE estimation  7   | Maximal change  1e-04
## ----
##  WLE Reliability = 0.657
```

```
fit2<-TAM::tam.fit(mod9a)
```

```
## Item fit calculation based on 100 simulations
## |**********|
## |----------|
```

```
summary(fit2)
```

```
##    parameter Outfit Outfit_t Outfit_p Infit Infit_t Infit_p
## 1         I1  0.879   -0.636    0.525 0.912  -0.584   0.559
## 2         I2  0.904   -0.383    0.702 0.976  -0.170   0.865
## 3         I3  1.119    0.463    0.643 1.099   0.683   0.495
## 4         I4  1.041    0.184    0.854 0.972  -0.109   0.913
## 5         I5  1.151    0.621    0.535 1.066   0.415   0.678
## 6     rater1  0.971    0.219    0.827 0.957  -0.504   0.614
## 7     rater2  1.005    0.282    0.778 1.002  -0.069   0.945
## 8     rater3  0.939    0.163    0.871 0.933  -0.744   0.457
## 9     rater4  0.993    0.258    0.797 0.963  -0.466   0.641
## 10    rater5  1.002    0.178    0.859 0.930  -0.696   0.486
## 11  I1:step1  0.971   -0.268    0.789 0.967  -0.454   0.650
## 12  I2:step1  0.886   -0.730    0.465 0.957  -0.479   0.632
## 13  I3:step1  1.019    0.138    0.890 1.012   0.127   0.899
## 14  I4:step1  1.094    0.468    0.640 1.004   0.062   0.950
## 15  I5:step1  1.005    0.057    0.955 1.004   0.057   0.955
## 16  I1:step2  0.756   -0.340    0.734 0.987  -0.030   0.976
## 17  I2:step2  0.957   -0.110    0.913 0.992   0.032   0.975
## 18  I3:step2  1.022    0.143    0.886 1.011   0.115   0.909
## 19  I4:step2  1.107    0.501    0.616 1.013   0.124   0.901
## 20  I5:step2  1.325    0.447    0.655 0.958  -0.145   0.885
## 21  I1:step3  0.929   -0.413    0.680 0.960  -0.331   0.741
## 22  I2:step3  0.922   -0.363    0.717 0.966  -0.179   0.858
## 23  I3:step3  1.008    0.082    0.934 1.006   0.085   0.932
## 24  I4:step3  1.102    0.491    0.623 1.011   0.112   0.911
## 25  I5:step3  1.027    0.144    0.886 0.979  -0.118   0.906
```

4. 结语

TAM的结果比较多，容易看糊涂，尤其是对多面RASCH来说。

- *xsi*里是δ和τ_i的值

- *xsi.facets*里是δ_i的值，即δ+τ_i
- *AXsi*里是$\Sigma\delta_i$的值，对多面RASCH来说，还包括评分者的宽严度的估计值，而且是rater X step的
- 可以得到*thresholds*的值，但是对多面RASCH来说，也是rater X step的
- 可以计算拟合值，拟合值是δ和τ_i分别计算的，还包括rater的，对非多面RASCH来说，则可以每个题目得到一个拟合值