

Teorias de Avaliação - CE095

Adilson dos Anjos¹

¹Departamento de Estatística
Universidade Federal do Paraná
aanjos@ufpr.br

Curitiba, PR
2 de outubro de 2014

Teoria da Resposta ao Item –Introdução ao pacote MIRT–

Principais recursos

- Análise de itens dicotômicos e politômicos;
- Modelos da TRI: Unidimensional (UIRT) e Multidimensional (MIRT);
- Métodos Bayesianos de estimação (priors);
- Recursos gráficos;
- Matenedor: Phil Chalmers .

Principais recursos

- Artigo:
mirt: A Multidimensional Item Response Theory Package for the R Environment, 2012
(<http://www.jstatsoft.org/v48/i06/paper>)
- <http://philchalmers.github.io/mirt/mirt-vignettes.html>
- <https://github.com/philchalmers/mirt>

Principais recursos

- Rasch (Rasch/partial credit),
- 1PL, 2PL, 3PL e 4PL,
- graded (graded response model) ,
- grsm (rating scale graded response model) ,
- gpcm (generalized partial credit model) ,
- rsm (Rasch rating scale) ,
- nominal (nominal model) ,
- mcm (multiple choice model) ,
- PC2PL (2PL partially compensatory model) ,
- PC3PL (2PL partially compensatory model) .

Exemplo:
análise de um modelo de 3 parâmetros com
pacote MIRT–

Pacote mirt

Exemplo: Simulado ENEM

```
> require(mirt)  
> require(ltm)
```

Pacote mirt

Exemplo: Simulado ENEM

```
> link<-'http://people.ufpr.br/~aanjos/CE095/dados/simenem.dat'
> prova<-read.fwf(link,w=c(7,-1,rep(1,45)))
> colnames(prova)<-c('id',paste('i',1:45,sep=""))
> head(prova,n=3)
```

	id	i1	i2	i3	i4	i5	i6	i7	i8	i9	i10	i11	i12	i13	i14	i15	i16
1	1001382	2	4	2	5	5	1	5	5	3	2	4	3	4	5	3	3
2	1005404	4	3	2	5	4	1	2	5	3	5	4	3	4	5	3	3
3	1013800	2	3	2	5	2	1	2	5	3	1	4	1	4	5	3	3

	i17	i18	i19	i20	i21	i22	i23	i24	i25	i26	i27	i28	i29	i30	i31
1	5	1	2	5	5	3	2	5	2	4	1	3	2	5	5
2	4	1	2	5	5	1	2	5	1	3	4	3	4	5	4
3	4	1	2	5	5	5	2	5	3	2	4	3	4	5	2

	i32	i33	i34	i35	i36	i37	i38	i39	i40	i41	i42	i43	i44	i45
1	1	4	2	4	2	1	1	3	5	5	5	3	2	2
2	3	3	1	4	2	4	2	3	1	5	5	3	1	2
3	3	3	1	1	2	4	2	3	1	5	4	1	4	2

```
> #prova<-prova[1:1000,2:46]
```

Exemplo: Simulado ENEM

```
> prova.desc<-descript(prova) # coluna 1 é id  
> prova.desc
```

Pacote mirt

Exemplo: Simulado ENEM

```
> gab<-c(4,3,2,5,2,1,2,5,3,1,4,3,4,5,3,3,4,1,2,5,5,
+        1,2,5,1,3,4,3,4,5,4,3,3,4,1,2,4,2,3,1,5,5,5,1,2)
> prova.c<-mult.choice(prova[,2:46],gab) # prova corrigida
> head(prova.c,n=3)
```

	i1	i2	i3	i4	i5	i6	i7	i8	i9	i10	i11	i12	i13	i14	i15	i16	i17
[1,]	0	0	1	1	0	1	0	1	1	0	1	1	1	1	1	1	0
[2,]	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1
[3,]	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
	i18	i19	i20	i21	i22	i23	i24	i25	i26	i27	i28	i29	i30	i31	i32		
[1,]	1	1	1	1	0	1	1	0	0	0	1	0	1	0	0		
[2,]	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
[3,]	1	1	1	1	0	1	1	0	0	1	1	1	1	0	1		
	i33	i34	i35	i36	i37	i38	i39	i40	i41	i42	i43	i44	i45				
[1,]	0	0	0	1	0	0	1	0	1	1	0	0	1				
[2,]	1	0	0	1	1	1	1	1	1	1	0	1	1				
[3,]	1	0	1	1	1	1	1	1	1	0	0	0	1				

Pacote mirt

A calibração pode ser feita a partir de uma amostra de respondentes.

Exemplo: Simulado ENEM

```
> amostra<-3000
> set.seed(60) # semente
> prova.s<-prova.c[sample(1:dim(prova.c)[1],amostra), ] # amostra de 3000
> head(prova.s)
```

	i1	i2	i3	i4	i5	i6	i7	i8	i9	i10	i11	i12	i13	i14	i15	i16	i17
[1,]	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
[2,]	0	0	1	0	0	1	1	1	1	1	1	0	0	1	1	0	0
[3,]	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1
[4,]	1	1	0	1	0	1	1	1	1	0	1	0	1	1	1	1	1
[5,]	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
[6,]	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1
	i18	i19	i20	i21	i22	i23	i24	i25	i26	i27	i28	i29	i30	i31	i32		
[1,]	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
[2,]	1	1	1	1	0	1	0	1	0	1	1	0	1	1	1		
[3,]	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
[4,]	0	1	0	0	1	1	0	1	0	1	1	1	0	1	1		
[5,]	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		

Pacote mirt

O modelo de 3 parâmetros precisa ser especificado na função mirt:

Exemplo: Simulado ENEM

```
> prova.par<-mirt(prova.s,1,itemtype='3PL')
```

```
Iteration: 1, Log-Lik: -63625.003, Max-Change: 3.42705  
Iteration: 2, Log-Lik: -62459.272, Max-Change: 0.54481  
Iteration: 3, Log-Lik: -62212.797, Max-Change: 0.48782  
Iteration: 4, Log-Lik: -62118.263, Max-Change: 0.92965  
Iteration: 5, Log-Lik: -62064.801, Max-Change: 0.29504  
Iteration: 6, Log-Lik: -62028.052, Max-Change: 0.25673  
Iteration: 7, Log-Lik: -62003.372, Max-Change: 0.26730  
Iteration: 8, Log-Lik: -61985.387, Max-Change: 0.19429  
Iteration: 9, Log-Lik: -61972.211, Max-Change: 0.27178  
Iteration: 10, Log-Lik: -61961.578, Max-Change: 0.12838  
Iteration: 11, Log-Lik: -61952.769, Max-Change: 0.44500  
Iteration: 12, Log-Lik: -61946.055, Max-Change: 0.45688  
Iteration: 13, Log-Lik: -61940.878, Max-Change: 0.37279  
Iteration: 14, Log-Lik: -61936.955, Max-Change: 0.40433  
Iteration: 15, Log-Lik: -61933.752, Max-Change: 0.09607  
Iteration: 16, Log-Lik: -61932.122, Max-Change: 0.27628
```

Exemplo: Simulado ENEM

Observe que o parâmetro d está na escala Normal. Para obter o valor de b na escala logística faça:

$$b = \frac{-d}{a}$$

Nessa função, $d = -ab$ pois,

$$a(\theta - b) = a\theta \underbrace{- ab}_d = a\theta + d$$

Exemplo:

```
> par(mfrow=c(1,1),ask=FALSE)
> itemplot(prova.par,item=1)
> par(mfrow=c(1,1))
```

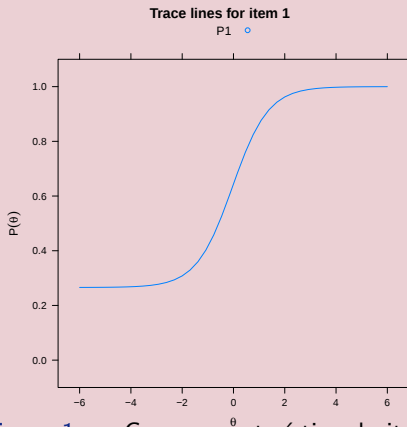


Figura 1 : Curva característica do item 1.

Exemplo:

```
> par(mfrow=c(1,1),ask=FALSE)
> itemplot(prova.par,item=41) # c=0
> par(mfrow=c(1,1))
```

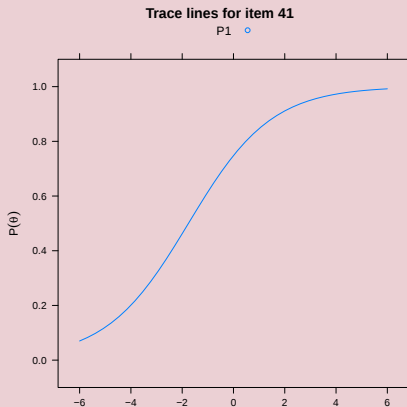


Figura 2 : Curva característica do item 41.

Exemplo:

```
> par(mfrow=c(1,1),ask=FALSE)
> itemplot(prova.par,item=1,type='info')
> par(mfrow=c(1,1))
```

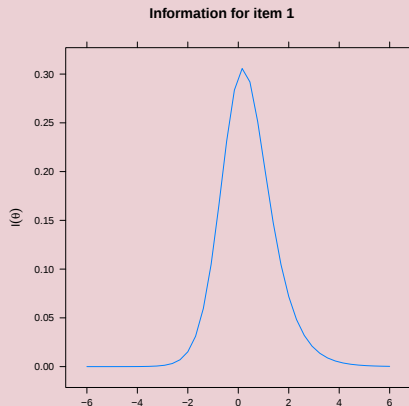


Figura 3 : Curva de informação do item 1.

Exemplo:

```
> par(mfrow=c(1,1),ask=FALSE)
> itemplot(prova.par,item=41,type='info')
> par(mfrow=c(1,1))
```

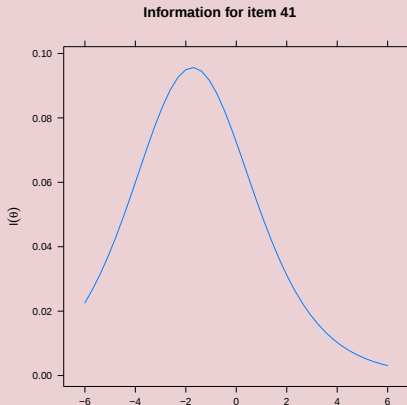


Figura 4 : Curva de informação do item 41.

Exemplo:

```
> print(plot(prova.par))
```

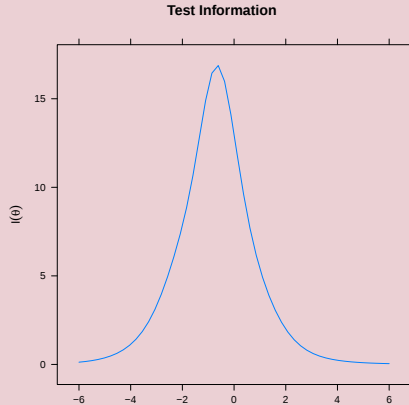


Figura 5 : Curva de informação do teste.

Proficiência

Exemplo:

```
> prova.sco<-fscores(prova.par)
```

Method: EAP

Empirical Reliability:

F1

0.89

Para obter a proficiência para algum padrão de resposta que não esteja no conjunto original de dados deve-se definir um valor para o argumento `response.vector` da função `fscores()`. Esse vetor precisa ter o mesmo tamanho do conjunto original de itens. No exemplo a prova foi simulada com 45 itens. Com a função `sample()` pode-se gerar um padrão de respostas qualquer:

Exemplo:

```
> set.seed(29)
> resp.nova<-sample(0:1,45,rep=T)
> resp.nova

[1] 0 0 0 0 1 0 1 1 0 0 1 0 0 1 0 1 1 0 1 0 1 1 1 1 1 0 0 1 1 0
[31] 0 0 0 1 0 0 1 0 0 0 1 0 1 1 0
```

Obtém-se a proficiência da seguinte maneira:

Exemplo:

```
> prova.sco.nova<-fscores(prova.par,response.vector=resp.nova)
```

Method: EAP

Empirical Reliability:

F1

0.89

Inserindo respostas:

Exemplo:

```
> novo<-c(1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,  
+         0,1,0,1,1,0,1,0,0,1,1,1,0,0,0,  
+         1,1,0,0,1,1,1,0,1,1,0,1,0,0,1)  
> prova.sco.nova2<-fscores(prova.par,response.vector=novo)
```

Method: EAP

Empirical Reliability:

F1

0.89

Exercício:

exercício

- Analise os dados de 'futebol' com as funções do pacote `mirt`;
- Compare os resultados do pacote `mirt` com os resultados do pacote `ltm`.