Metodologia de Estimação Clássica Implementada no Pacote ltm do R

Método de Estimação de Máxima Verossimilhança aplicado no pacote ltm do R

Modelos de Resposta ao Item

- Os problemas de estimação envolvem dois tipos de parâmetros, Itens e Habilidades.

- Divide-se em três situações:

1. Quando se deseja estimar as habilidades;
2. Quando se deseja estimar os parametros itens;
3. Quando se deseja estimar os parâmetros dos itens e as habilidades dos indivíduos simultaneamente.

- Geralmente a estimação é feita pelo Método da Máxima Verossimilhança através da aplicação de algum processo iterativo, como o algoritmo *Newton-Raphson* ou *Scoring"de Fisher* .

Método de Máxima Verossimilhança Marginal

- Criado por Bock&Lieberman – 1970
- Proposta desse método: Fazer a estimação em duas etapas:

1. Primeiro os parâmetros dos itens
2. Posteriormente, as habilidades.

- Como as habilidades não são conhecidas, precisaremos usar algum artifício de forma que a verossimilhança não seja mais função das habilidades.

- Artifício para eliminar as habilidades na verossimilhança: marginalizar a verossimilhança integrando-a com relação à distribuição da habilidade.

- De forma geral, consideremos que as habilidades θ_j, j=1,..., n são realizações de uma variável aleatória θ com distribuição contínua e função densidade de probabilidade (fdp) $g(\theta|\eta)$, duplamente diferenciável, com as componentes de η conhecidas e finitas.
- Para o caso em que θ tem distribuição Normal, temos $\eta = (\mu, \sigma^2)$, onde μ é a média e σ^2 a variância das habilidades. Portanto, se desejarmos que os itens sejam estimados na métrica (0,1), deverá adotar $\mu = 0 \ e \ \sigma^2 = 1$.

Abordagem de Bock e Aitkin (1981)

- Proposta: Adotar independência entre os itens de forma a possibilitar que os itens sejam estimados individualmente.
- Sugere-se que a estimativa da máxima verossimilhança seja feita através do Algoritmo EM (Dempster, Laird & Rubin – 1977).

- As equações de estimação em forma de quadratura para os parâmetros *ai*, *bi* e *ci* são respectivamente:

- $a_i = D(1-c_i)\sum_{k=1}^{q}(\overline{\theta_k} - b_i)[r_{ki} - P_{ki}f_{ki}]W_{ki} = 0$

- $b_i = -Da_i(1-c_i)\sum_{k=1}^{q}[r_{ki} - P_{ki}f_{ki}]W_{ki} = 0$

- $c_i = \sum_{k=1}^{q}[r_{ki} - P_{ki}f_{ki}]\frac{W_{ki}}{P_{ki}^{*}} = 0$

- Onde:

$r_{ki}(\theta) = \sum_{j=1}^{s} r_j u_{ji} g_{jk}^{*}(\theta)$,

$f_{ki}(\theta) = \sum_{j=1}^{s} r_j g_{jk}^{*}(\theta)$ e

$g_{jk}^{*}(\theta) = g_j^{*}(\overline{\theta_k})$.

Aplicação do Algoritmo EM

- Processo iterativo para determinar estimativas de máxima verossimilhança de parâmetros de modelos de probabilidade na presença de variáveis aleatórias não observadas.
- Objetivo: obter estimativas de ζ na presença de variáveis não observadas θ.

- Sendo $u_{..}$ um vetor incompleto, e $(u_{..}, \theta)$ os dados completos. Temos $f(u_{..}, \theta | \zeta)$ a densidade conjunta dos dados completos. Os passos EM para $\hat{\zeta}^{(k+1)}$:

1. **PASSO E (Esperança):** Calcular $E\left[\log f(u_{..}, \theta | \zeta) | u_{..}, \hat{\zeta}^{(k)}\right]$.
2. **PASSO M (Maximização):** Obter $\hat{\zeta}^{(k+1)}$ que maximiza a função do PASSO E.

- Há três formas do algoritmo EM, distinguidas pela relação entre a função (densidade) de probabilidade e a forma da família exponencial:

1. Aplicar quando a função é membro regular da família exponencial;
2. Aplicar quando a função não é membro regular da família exponencial, mas é membro da família exponencial curvada;
3. Aplicar quando a função não tem nenhuma relação com a família exponencial.

Descrevendo o Algoritmo

- Suponha que as habilidades são restritas a um conjunto de valores q, $\overline{\theta_k}$, k=1,... , q com probabilidades π_k.
- Seja f_{ki} o número de indivíduos com habilidade θ_k respondendo ao item i e r_{ki} o número de indivíduos com habilidade $\overline{\theta_k}$ respondendo corretamente ao item i.

- Desenvolvendo a distribuição conjunta de $\boldsymbol{f}$ e $\boldsymbol{r}$, aplicando o logaritmo e aplicando a Esperança temos:

$$E[\log \mathrm{L}(\zeta)] = \bar{C} + \sum_{i=1}^{I}\sum_{k=1}^{q}\{r_{ki} log P_{ki} + (\bar{f}_{ki} - \bar{r}_{ki}) log Q_{ki}\}$$

Assim, os Passos E e M são:

- **PASSO E:** Usar os pontos de quadratura $\overline{\theta_k}$, os pesos Ak, k = 1, ..., q e estimativas iniciais dos parâmetros dos itens, $\hat{\zeta}_i$; i = 1, ...,I; para gerar $g_j^*(\overline{\theta_k})$ e, posteriormente, $\bar{f}_{ki}$ e $\bar{r}_{ki}$.

- **PASSO M:** Com $\boldsymbol{r}$ e $\boldsymbol{f}$ obtidos no Passo E, resolver as equações de estimação para ζ_i i = 1,..., I; usando o algoritmo Newton-Raphson ou "Scoring" de Fisher.

Pacote LTM para TRI – Software R

- Desenvolvido para análise dicotômica multivariada e para dados politômicos utilizando modelos de variáveis latentes.

- Estimativas são obtidas sobre máxima verossimilhança marginal usando quadratura de Gauss-Hermite.

Modelo de Variável Latente:

- Modelo de regressão multivariada que liga a categoria resposta à co-variável não observada.
- É necessária independência condicional, pois simplifica o processo de estimação.
- Nesse modelo as variáveis não observadas podem ser quantificadas pelas variáveis latentes.

- Ideia básica: encontrar, para um dado conjunto de variáveis resposta x1,. . . , XP, um conjunto de variáveis latentes Z1,. . . , ZQ (Q<P) com:

$E(x_i|z) = g(\lambda_{i0} + \lambda_{i1}z_1 + \cdots + \lambda_{iq}z_q),$ i=1,...p

Dados Dicotômicos

- Modelo para a probabilidade de resposta correta em cada item, dado o nível de habilidade z. Modelo geral:

$$P(x_{im} = 1|z_m) = c_i + (1 - c_i)g\{ a_i (z_m - b_i)\}$$

Onde:

- x_{im} =é a variável dicotômica
- z_m = nível do examinador sobre a latente
- c_i = parâmetro relacionado à probabilidade do individuo com habilidade baixa responder o item corretamente
- a_i = parâmetro de discriminação
- b_i = parâmetro de dificuldade

Dados Politômicos

- Análise de variáveis politômicas está atualmente tratada por ltm usando o Modelo de Resposta Gradual.

$$P(x_{im} = k|z_m) = \; g(\eta_{ik}) - g(\eta_{ik+1})$$

onde $\eta_{ik} = \; a_i \, (z_m - \; b_{ik})$

Implementação do Pacote LTM

- Os parâmetros são estimados pela maximização da log-verossimilhança dos dados observados integrando as variáveis latentes.

- LTM tem quatro funções para ajustar o modelo:

1. *rash( )* - para modelo Rash
2. *ltm( )* - para modelos de Traço Latente
3. *tpm( )* - para modelos de três parâmetros
4. *grm( )* – para modelos de respostas graduais

- A maximização é conseguida utilizando *optim( )* para *rash*, *tpm* e *grm*. Para *ltm* um algoritmo híbrido é adotado.

- Os valores iniciais são obtidos através do comando *start.val( )*.

- Com o retorno dos objetos através das quatro funções anteriores de ajuste do modelo torna-se disponível os comandos: *print( )*, *coef( )*, *summary( )*, *plot ( )*, *fitted()*, *loglik( )*, *margins( )*,*anova( )*, *fator.scores( )*.

Uso do Pacote LTM no R

- **Dados binários:** comando *descript( )* para uma análise descritiva dos dados TRI, utiliza-se um dos quatro comandos anteriores para o ajuste do modelo, sendo o mais comum o *rash( ), ltm( )*. Utiliza-se o comando *information( )* para verificar a área sob o teste da curva do item em um intervalo especificado.
- **Dados ordinais (politômicos):** uso do *descript()*. O comando *vcor.teste( )* como um método alternativo para explorar o grau de associação entre os pares com coeficiente de correlação não paramétrica. Para o ajuste do modelo é comum a utilização do comando *grm( )*.

Referências

- Andrade, D.F.; Tavares, H.R.; Valle, R.C. (2000) Teoria de Resposta ao Item: Conceitos e Aplicações.

- Rizopoulos, D. (2006) ltm: An R Package for Latent Variable Modeling and Item Response Theory Analyses