Rasch Model and its Application in Word Memorization Software

Zhao Feng

Institute of Mathematics
Tsinghua University

February 10, 2017

在心理测试中，IRT(Item Response Theory) 对某个 person 的能力值 θ(latent variable) 的估计值, 是通过该 person 在若干个 item 的测试结果给出的, 以下的讨论局限于每个 item 的测试结果是 0 或 1，分别代表答案错误与正确。假设各个 item 彼此独立，被试者对某个 item 的回答正确的概率用 IRF(item response function) 建模. 一般能力 θ 会做一个归一化, 使得其均值为 0 标准差为 1，这样 $\hat{\theta}$ 作为 θ 的估计值一般在-3 到 3 之前，非常接近 0 表示水平中等。这种归一化给不同测试集之间相互比较提供了方便。

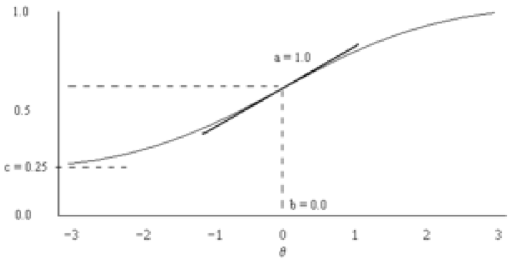
IRF 函数有多种不同的建模方式，一般常用 Logistic function:

IRF 函数有多种不同的建模方式，一般常用 Logistic function:

$$p_i(\theta) = c_i + \frac{1 - c_i}{1 + exp(-a_i(\theta - b_i))} \tag{1}$$

上式中 i 表示被试者的编号，a,b,c 是 item 的参数，分别表征 discrimination,difficulty 和 pseudo guessing, 可以从下图 (ICC 曲线,item characteristic curve) 形象地说明这三个参数

Figure: 三个参数的 IRF



三个参数的 IRF 虽然精确，但实际中估计参数比较繁琐，一般常用的是 1 个参数 (b) 的 Rasch Model, 其可以简化表述为第 k 个 person 在第 i 个 item 上答对的概率为

三个参数的 IRF 虽然精确，但实际中估计参数比较繁琐，一般常用的是 1 个参数 (b) 的 Rasch Model, 其可以简化表述为第 k 个 person 在第 i 个 item 上答对的概率为

$$P(X_{ki}=1)=\frac{exp(\beta_k-\delta_i)}{1+exp(\beta_k-\delta_i)} \tag{2}$$

上式中 β_k 表示 ability,δ_i 表示 difficulty. 在获得 person $\times$ item 的二维表格数据后，要先根据数据估计 Rasch Model 的参数 $\vec{\delta}=(\delta_1,..\delta_I)$, 常用的方法有极大似然法，CML,EM 等

首先讨论 JML(joint maximum likelihood) 的方法,observed data matrix 联合概率似然函数为

首先讨论 JML(joint maximum likelihood) 的方法,observed data matrix 联合概率似然函数为

$$\log(\Lambda) = \sum_{k=1}^{N} \beta_k r_k - \sum_{i=1}^{I} \delta_i s_i + \sum_{k=1}^{N} \sum_{i=1}^{I} \log(1 + exp(\beta_k - \delta_i)) \quad (3)$$

其中 $r_k = \sum_{i=1}^{I} x_{ki}$, 表示第 k 个 person 的总分，$s_i = \sum_{k=1}^{N} x_{ki}$, 表示第 i 个 item 的总分,$p_{ki}$ 为 (2)。对对数似然函数关于 δ_i 和 β_k 求偏导，得到含 β_k 和 δ_i 的非线性方程组为

首先讨论 JML(joint maximum likelihood) 的方法,observed data matrix 联合概率似然函数为

$$\log(\Lambda) = \sum_{k=1}^{N} \beta_k r_k - \sum_{i=1}^{I} \delta_i s_i + \sum_{k=1}^{N} \sum_{i=1}^{I} \log(1 + exp(\beta_k - \delta_i)) \quad (3)$$

其中 $r_k = \sum_{i=1}^{I} x_{ki}$, 表示第 k 个 person 的总分，$s_i = \sum_{k=1}^{N} x_{ki}$, 表示第 i 个 item 的总分,$p_{ki}$ 为 (2)。对对数似然函数关于 δ_i 和 β_k 求偏导，得到含 β_k 和 δ_i 的非线性方程组为

$$\begin{aligned} s_i &= \sum_{k=1}^{N} p_{ki}, i = 1, ..I \\ r_k &= \sum_{i=1}^{I} p_{ki}, k = 1, ..N \end{aligned} \quad (4)$$

对实际应用来说，一般 N 很大，直接求解 (4) 计算量太大。故一般先求只含 item 的边缘概率分布，在 item 的参数 δ_i 求出的情况下，由于各个 person 之间相互独立，只需分别对只含一维参数 β_k 的函数求极大值点即可。对第 k 个 person, 其各 item 得分的 joint distribution 为

对实际应用来说，一般 N 很大，直接求解 (4) 计算量太大。故一般先求只含 item 的边缘概率分布，在 item 的参数 δ_i 求出的情况下，由于各个 person 之间相互独立，只需分别对只含一维参数 β_k 的函数求极大值点即可。对第 k 个 person, 其各 item 得分的 joint distribution 为

$$
\begin{aligned}
P(\vec{x_k}|\beta_k, \vec{\delta}) &= \prod_{i=1}^{I} \frac{exp(x_{ki}(\beta_k - \delta_i))}{1 + exp(\beta_k - \delta_i)} \\
&= \frac{exp(r_k\beta_k)exp(-\sum_{i=1}^{I} x_{ki}\delta_i)}{\prod_{i=1}^{I}(1 + exp(\beta_k - \delta_i))}
\end{aligned} \tag{5}
$$

Conditional Maximum Likelihood Method

定义 $\gamma_{r|\vec{\delta}} = \sum_{||\vec{y}||_1=r} exp(-\sum_{i=1}^{I} y_i\delta_i)$, 为 elementary symmetric function, 则条件似然函数 $P(x_k|r_k, \vec{\delta})$ 为

Conditional Maximum Likelihood Method

定义 $\gamma_{r|\vec{\delta}} = \sum_{||\vec{y}||_1=r} exp(-\sum_{i=1}^{I} y_i\delta_i)$, 为 elementary symmetric function, 则条件似然函数 $P(x_k|r_k,\vec{\delta})$ 为

$$
\begin{aligned}
P(x_k|r_k,\vec{\delta}) &= \frac{P(x_k|\beta_k,\vec{\delta})}{P(r_k|\beta_k,\vec{\delta})} \\
&= \frac{exp(-x_{ki}\delta_i)}{\gamma_{r_k|\vec{\delta}}}
\end{aligned} \tag{6}
$$

上式不含 β_k，说明 r_k 是参数 β_k 的充分统计量。由于各 person 得分相互独立, 只需把 N 个对数似然函数相加即可。

Conditional Maximum Likelihood Method

定义 $\gamma_{r|\vec{\delta}} = \sum_{||\vec{y}||_1 = r} exp(-\sum_{i=1}^{I} y_i \delta_i)$, 为 elementary symmetric function, 则条件似然函数 $P(x_k|r_k, \vec{\delta})$ 为

$$
\begin{aligned}
P(x_k|r_k, \vec{\delta}) &= \frac{P(x_k|\beta_k, \vec{\delta})}{P(r_k|\beta_k, \vec{\delta})} \\
&= \frac{exp(-x_{ki}\delta_i)}{\gamma_{r_k|\vec{\delta}}}
\end{aligned} \tag{6}
$$

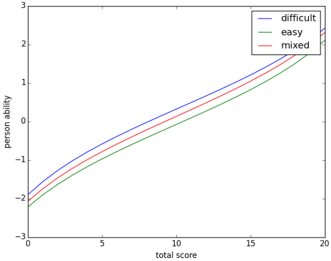
上式不含 β_k，说明 r_k 是参数 β_k 的充分统计量。由于各 person 得分相互独立, 只需把 N 个对数似然函数相加即可。

$$
\log(\Lambda(\vec{x}|\vec{r}, \vec{\delta})) = \sum_{k=1}^{N} \frac{exp(-x_{ki}\delta_i)}{\gamma_{r_k|\vec{\delta}}} \tag{7}
$$

下面来讨论 Rasch Model 在背单词软件的具体应用，item difficulty 的参数 b 即为单词难度，具体实施中可能要考虑到:

1. 根据课本内容统计词频，归一化后作为每个单词难度的近似替代量
2. 每一个用户初始化背单词能力为 0，每一次背单词后保留其该次背单词能力的估计值，在下一次背单词时采用之前能力值的加权平均值，对于该平均值-单词难度 >3 的单词则不予考虑，在其他单词中按单词难度进行重要度抽样，样本数量为 N 个, 作为该次背单词的测试集。每次用户的有效测试（没有中途退出和缺失值）保存到服务器的数据库用来更新单词难度。
3. 定期更新单词难度之前集齐一定数量的测试结果, 应考虑到用户的能力变化曲线，有选择地剔除某一部分数据再用 CML 全局计算单词难度，将计算值与原有的频率值做平均。

下面假定各个 item 难度值已知，给定 person 在每一个 item 上的得分, 由 Rasche Model 可以估计 person 的 ability，在测试题目给定的情况下和总分具有非线性的一一对应关系, 通过极大似然的方法可以推导出 person 的 ability 满足的代数方程 [13], 用牛顿法求解方程即得到 ability 参数。实际实现时发现牛顿算法在分数接近满分和接近零分时误差较大，改用优化的方法求 β 在 [-3,3] 区间的极大值则无此问题，下图是利用仿真数据得到的分数-能力曲线:



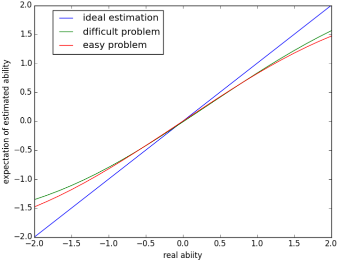
上图中三组数据分别是 20 个难题，20 个容易题，和 10 个难题和 10 个容易题的混合组，由上图可看出每一条曲线有如下特点:

1. 能力和分数的关系在高分和低分段斜率比较大，非线性性比较明显，而在中部接近线性。
2. 平均水平能力为 0 对应答出来一半的题目，曲线具有对称性。

• 比较不同的曲线也符合直观, 答出同样数量的题，对于难题组能力高，混合组能力次之，最末为简单题组。通过用 Ability 而不是 score 来衡量从而消除了某一次 Test 题目的影响而在一个统一的 scale 上比较。

精度分析

对于单次测试而言，可以用总的 Fisher 信息量 $I(\beta)$ 衡量精度，一般而言，对于某一个估计的值 $\hat{\beta}$,$I(\hat{\beta})$ 越大则表明估计统计量的方差越小，能力估计的精度越高。[15] 基于 MLE 的方法一般都是有偏的，即估计统计量 $\hat{\beta}$ 的均值不等于 β, 由于引入了先验的分布，这种偏差在两级状态会非常明显。[17]，下图是比较两组题目 bias 的结果:



如果 person 的能力接近平均水平 (ability ≈ 0)，则几乎没有 bias, 但两级的 bias 会比较大。由于采用了先验的正态总体假设，对于高分，会低估 person 的能力而对于低分则会高估 person 的能力。如果题目较难，高分段的 bias 减小而低分段的 bias 增大。

用 β 表示某人的能力，β 的先验分布记成 $p(\beta)$, 一般是正态分布，表示在没有考试成绩的时候对其能力的估计，设此人参加了有 $i = 1, 2, ..I$ 个 item 组成的测试，得分为 x_i, 每道题的难度为 δ_i, 由贝叶斯公式，对其成绩的后验估计为:

用 β 表示某人的能力，β 的先验分布记成 $p(\beta)$, 一般是正态分布，表示在没有考试成绩的时候对其能力的估计，设此人参加了有 $i = 1, 2, ..I$ 个 item 组成的测试，得分为 x_i, 每道题的难度为 δ_i, 由贝叶斯公式，对其成绩的后验估计为:

$$p(\beta|\vec{x}) = \frac{p(\beta)p(\vec{x}|\beta)}{p(\vec{x})} \propto p(\beta)p(\vec{x}|\beta) \tag{8}$$

β 最有可能的取值为:

用 β 表示某人的能力，β 的先验分布记成 $p(\beta)$, 一般是正态分布，表示在没有考试成绩的时候对其能力的估计，设此人参加了有 $i = 1, 2, ..I$ 个 item 组成的测试，得分为 x_i, 每道题的难度为 δ_i, 由贝叶斯公式，对其成绩的后验估计为:

$$p(\beta|\vec{x}) = \frac{p(\beta)p(\vec{x}|\beta)}{p(\vec{x})} \propto p(\beta)p(\vec{x}|\beta) \tag{8}$$

β 最有可能的取值为:

$$\underset{\beta}{\operatorname{argmax}}\, p(\beta)p(\vec{x}|\beta) \tag{9}$$

其中 $p(\vec{x}|\beta)$ 由 Rasch Model 给出:

用 β 表示某人的能力，β 的先验分布记成 $p(\beta)$, 一般是正态分布，表示在没有考试成绩的时候对其能力的估计，设此人参加了有 $i = 1, 2, ..I$ 个 item 组成的测试，得分为 x_i, 每道题的难度为 δ_i, 由贝叶斯公式，对其成绩的后验估计为:

$$p(\beta|\vec{x}) = \frac{p(\beta)p(\vec{x}|\beta)}{p(\vec{x})} \propto p(\beta)p(\vec{x}|\beta) \tag{8}$$

β 最有可能的取值为:

$$\underset{\beta}{\operatorname{argmax}}\, p(\beta)p(\vec{x}|\beta) \tag{9}$$

其中 $p(\vec{x}|\beta)$ 由 Rasch Model 给出:

$$p(\vec{x}|\beta) = \prod_{i=1}^{I} \frac{exp(x_i(\beta - \delta_i))}{1 + exp(\beta - \delta_i)} \tag{10}$$

对含先验分布的对数似然函数 $\log(p(\vec{x}|\beta))$ 关于 β 求导得:

对含先验分布的对数似然函数 $\log(p(\vec{x}|\beta))$ 关于 β 求导得:

$$\frac{p'(\beta)}{p(\beta)} + \sum_{i=1}^{I} x_i = \sum_{i=1}^{I} \frac{exp(\beta - \delta_i)}{1 + exp(\beta - \delta_i)} \tag{11}$$

上面方程的解 β 对各项得分的依赖仅仅通过总分 $\sum_{i=1}^{I} x_i$ 的形式，因此总分是参数 β 的充分统计量。由于 Rasch dichotomous Model 对 person 的能力只有一个维度的假定, 在 items 一定的情况下，相同能力与相同总分一一对应。如果不加先验分布，在全对和全错两种极端情况下方程 (11) 无解，因此适当的先验分布是必要的，有用户 ability 数据的情况下可以拟合正态分布的参数，在缺少用户数据的初始化阶段可以用标准正态分布代替，此时上式第一项化为 $-\beta$。

Rasch 模型 Fisher 信息量

$I(\beta)$ 的计算公式为:

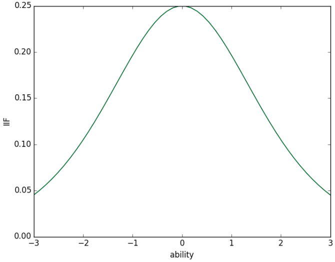
$$I(\beta) = -E(\frac{\partial^2 \log p(\vec{x}|\beta)}{\partial \beta^2}) \tag{12}$$

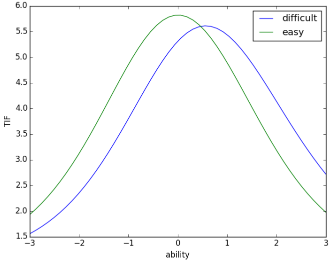
对于 Rasch Model, 代入似然函数表达式，取先验分布为正态分布，则有

$$I(\beta) = 1 + \sum_{i=1}^{I} \frac{exp(\beta - \delta_i)}{(1 + exp(\beta - \delta_i))^2} \tag{13}$$

上式中第一项是先验分布的信息量，后面分别是每一个 item 的信息量 (item infromation function)，它们彼此独立因而可以相加。上式的和也被称为 Test Information Function(TIF).

对于每一个 item, 其 IIF 有左图所示的形式:





左图是假定 item difficulty 为 0，当 $\beta - \delta_i > 3$ 时 (题目过难或过易)，IIF 已经小于 0.05, 在这种情况下能力估计的误差为比较大。由于先验分布对 TIF 的贡献是常数，可以在一定程度上减轻能力的极端情况造成的信息量过少的问题。右图是 20 道题的 TIF 结果，对于难题组，全对的信息量要比简单题大，而全错的信息量比简单题小，这符合一般经验。

Rasch 模型 Bias 数值计算

计算 bias 需要计算给定真实能力后计算关于估计量的期望值，在对模型没有任何了解的情况下可以采用 Monte Carlo 模拟，但对于 Rasch Model 由 (11) 式得 $r_I = \sum_{i=1}^{I} x_i$ 是 β 的充分统计量，于是首先计算总分 r_I 的分布，再由

$$E(\hat{\beta}(r_I)) = \sum_{i=1}^{I} \hat{\beta}(i) P(r_I = i) \tag{14}$$

计算出期望值。在上式中 $\hat{\beta}(i)$ 可以通过对 (9) 求解极大值得到，而 r_I 的分布可以类比组合数的计算方法递推得到。记 $A_n^k = \sum_{k=1}^{n} x_k$, 则有如下递推公式:

$$A_n^k = P(x_n = 1)A_{n-1}^{k-1} + P(x_n = 0)A_{n-1}^k \tag{15}$$

上式中 $P(x_n = 1)$ 为第 n 道题的答对概率。该方法计算规模为 $O(n^3)$, 其中 n 为 item 数量，相比 Monte Carlo 模拟需要大量重复才能得到比较精确的结果在 n 不大时效率比较高。

Further Reading I

Rasch model estimation.
https://en.wikipedia.org/wiki/Rasch_model_estimation

Patrick Mair,Reinhold Hatzinger,Marco J. Maier
Extended Rasch Modeling: The R Package eRm.