数理解析研究所講究録
第1955巻 2015年 76-80

項目応答理論への数式処理の応用について

北本 卓也
TAKUYA KITAMOTO*
山口大学
YAMAGUCHI UNIVERSITY

Abstract

本論文では、項目応答理論の数式処理への応用について議論する。項目応答理論は、TOEFL や IT パスポート試験なので実際に活用されている新しい能力判定法であり、通常の正当加点方式より精緻が能力推定が行えると言われている。しかしながら、その方法は統計的な手法で受験者の能力値を推定しているため計算式が複雑であり、活用が難しいことや、準備が大変なことが問題点として挙げられる。そこで本稿では項目応答理論の計算について簡単に説明し、実際にその計算手順を述べる。さらに数式処理の項目応答理論への活用について述べる。具体的には、パラメータがシステムに含まれる場合にその計算結果をべき級数の形で計算する手法について解説する。

1 はじめに

様々な資格や現代社会においては、様々な資格試験が存在するが、これらの試験においては正確な能力判定が非常に重要である。通常の試験では正答した問いに対する点を加算して点数を計算する正当加算方式が受験者の能力推定値として使われることが多いが、この正当可算方式では各問の点を何点に設定するかなど対応しなければならない問題点もある。そこで TOEFL や IT パスポート等の試験では、項目応答理論という新しい能力評価方式で受験者の能力推定が行われている。この項目応答理論では統計的な手法で受験者の能力値を推定しているため、正当加算方式より精緻が能力推定が行えると言われている反面、計算式が複雑であり、活用が難しいことや、準備が大変なことが問題点として挙げられている。本稿では、この項目応答理論について概説し、その実際の計算法を述べる。また、数式処理を用いて、システムにパラメータが入った時の能力推定値の近似を計算する手法について述べる。

2 項目応答理論 (IRT) とは

項目応答理論 (以下、IRT と記す) は、TOEFL や IT パスポート試験などで実際に使われている能力判定方式である。通常の試験が正答加点方式であるのに対して、統計的な処理で受験者の能力を判定し、より精密な能力判定が可能であるとされている。

IRT では各問ごとに「難易度」と「識別力」というパラメータを設定する。難易度は問題の難しさを指している。識別力は受験者の能力を判定する力を示しており、識別力が高い問題というのは「能力が高い人はほとんど正解し、能力が低い人はほとんど不正解である」問題であるである。いわゆる良い問題は識別力

*kitamoto@yamaguchi-u.ac.jp

が高い問題という言い方ができる。IRT では、識別力が a_j、難易度が b_j である問題を能力値が θ である人が正解する確率 $p_j(\theta)$ は

$$p_j(\theta) = \frac{1}{1+e^{-Da_j(\theta-b_j)}} \quad (\text{ただし、} D = 1.7) \tag{1}$$

であると考える（これをロジスティックモデルという）。これより能力が θ である人が同じ問題を間違える確率は $(1-p_j(\theta))$ である事がすぐわかる。

今、問題が (1), (2) の 2 問あり、(1) の識別力と難易度の組 (識別力, 難易度) が (a_1, b_1) であり、(2) の (識別力, 難易度) が (a_2, b_2) である場合を考える。このとき、能力 θ の人が (1) を正解、(2) を不正解する確率は $p_1(\theta)\,(1-p_2(\theta))$ である。これより、実際の試験で (1) を正解、(2) を不正解した人がいた場合、この人の能力値は $p_1(\theta)\,(1-p_2(\theta))$ が最大値をとる θ の値とすることが最も合理的である。また同じ理由で、実際の試験で (1) を不正解、(2) を正解した人がいた場合、この人の能力値は $(1-p_1(\theta))\,p_2(\theta)$ が最大値をとる θ の値とすることが最も合理的である。$a_1 \neq a_2$ または $b_1 \neq b_2$ であれば、$p_1(\theta)\,(1-p_2(\theta))$ と $(1-p_1(\theta))\,p_2(\theta)$ の関数は異なるので当然これらの関数が最大値を取る θ の値も異なってくる。つまり、正答加点方式では等しく 1 点と数えられていた試験結果が IRT ではより精密に判定されることになる。

より一般的には、IRT では次のように能力判定を行う。今、n 個の問題があり、j 番目の問題の (識別力, 難易度) が (a_j, b_j) で与えられるとする。この試験を受けた受験番号 i の人が j 番目の問題が正解した場合、$u_{ij} = 1$ とし、不正解の場合 $u_{ij} = 0$ と u_{ij} $(j = 1, .., n)$ の値を定める。この時、受験番号 i の人の能力は関数

$$L_i(\theta) = \prod_{j=1}^{n} p_j(\theta)^{u_{ij}} \left(1-p_j(\theta)\right)^{1-u_{ij}} \tag{2}$$

が最大値を取る θ の値であると推定する。

実際には上の $L_i(\theta)$ の対数をとった

$$\log\left(L_i(\theta)\right) = \sum_{j=1}^{n} \left\{u_{ij} \log\left(p_j(\theta)\right) + \left(1-u_{ij}\right) \log\left(1-p_j(\theta)\right)\right\} \tag{3}$$

の関数の方が簡単なので (3) を考える（(2) の最大値を取る θ と (3) の最大値を取る θ は同じである）。関数の最大化を考えるには、(3) の導関数 $\frac{d\log(L_i(\theta))}{d\theta}$ の零点を求めれば良い。

以上をまとめると、次の能力判定アルゴリズムを得る。

(i) 問題の識別力、難易度を求め、j 番目の問題の (識別力, 難易度) を (a_j, b_j) とする。

(ii) 試験結果を u_{ij} とし、(2) より θ の関数 $L_i(\theta)$ を定義する。ただし、受験番号 i の人が j 番目の問題に正解したときは $u_{ij} = 1$、不正解の時は $u_{ij} = 0$ とする。

(iii) $\frac{d\log(L_i(\theta))}{d\theta} = 0$ の θ に関する根を求め、それを i 番目の受験者の能力値とする。

上のアルゴリズムのステップ (i) の (a_j, b_j) の計算は次のように行う。今、n 問の問題がある試験を受けた m 人の受験者がいて、u_{ij} $(i = 1, ..., m,\ j = 1, ..., n)$ のような結果を得たとする（ただし、受験番号 i の人が j 番目の問題に正解したときは $u_{ij} = 1$、不正解の時は $u_{ij} = 0$ とする）。

受験番号 i の人の能力値を θ_i とするとき、このような事が起こる確率は $\prod_{i=1}^{m} L_i(\theta_i)$ である。これが実際に起こった事ならば、この確率 $\prod_{i=1}^{m} L_i(\theta_i)$ を変数 θ_i $(i = 1, ..., m)$, a_j, b_j $(j = 1, ..., n)$ の関数と考え、この確率を最大化するように変数 θ_i, a_j, b_j ををを定めることが最も合理的である。すなわち、

$$\frac{d}{d\theta_i}\prod_{i=1}^{m} L_i(\theta_i) = 0 \quad (i = 1, ..., m), \quad \frac{d}{d\theta_i}\prod_{i=1}^{m} L_i(a_j) = 0 \quad (i = j, ..., n), \quad \frac{d}{d\theta_i}\prod_{i=1}^{m} L_i(b_j) = 0 \quad (i = j, ..., n) \tag{4}$$

の連立方程式の解を求めればよいが、この方程式は悪条件であることが知られており、特に m,n が大きくなると解くことは現実的でない。そこで実際には「変数 θ の周辺化」というテクニックを用いて、変数 θ を方程式より消去する。

今、「受験者の能力値 θ が平均値 0、分散 1 の正規分布に従う」と仮定すると「$\prod_{i=1}^{m} L_i(\theta_i)$ を $\theta_1,...,\theta_m$, $a_1,...,a_n$, $b_1,...,b_n$ で最大化する」ということは「$\prod_{i=1}^{m}\int_{-\infty}^{\infty} g(\theta)L_i(\theta)d\theta$ を $a_1,...,a_n$, $b_1,...,b_n$ で最大化する」と同値となる（ただし、$g(\theta)=\frac{1}{\sqrt{2\pi}}e^{-\frac{\theta^2}{2}}$)。そこで、関数 $\prod_{i=1}^{m}\int_{-\infty}^{\infty} g(\theta)L_i(\theta)d\theta$ の最大化を「EM アルゴリズム」と呼ばれるアルゴリズムを用いて行う。この「EM アルゴリズム」は、この計算に特化した特殊なアルゴリズムであるが、Easy Estimation というフリーソフトがあり、それを用いて行うことができる。

3 IRT 活用の数値例

今、試験の結果 u_{ij} が次のように得られたとする（問題数 6、受験者数 18 であり、受験番号 i の受験者が j 番目の問題を正解した時に $u_{ij}=1$、不正解の時に $u_{ij}=0$ である)。

$$u_{ij}=\begin{pmatrix}
1&1&1&0&1&1\\
1&0&1&0&1&1\\
0&1&1&0&0&1\\
1&1&1&0&1&1\\
1&1&1&0&1&1\\
0&0&0&0&1&0\\
1&1&1&0&1&1\\
1&0&1&1&1&1\\
1&1&0&0&0&0\\
1&1&1&0&0&0\\
1&1&1&1&1&0\\
0&1&0&0&1&1\\
1&0&1&0&1&1\\
0&1&0&0&0&1\\
0&1&1&1&1&1\\
1&1&0&0&0&1\\
1&1&0&1&1&1\\
0&1&1&1&0&1
\end{pmatrix} \tag{5}$$

このとき、このデータを用いて EasyEstimation で各問題の識別力 a_j と難易度 b_j を求めると、

$$\begin{pmatrix} a_1 & b_1\\ a_2 & b_2\\ a_3 & b_3\\ a_4 & b_4\\ a_5 & b_5\\ a_6 & b_6 \end{pmatrix}=\begin{pmatrix} 0.55590 & -1.06248\\ 0.08773 & -6.02801\\ 1.09860 & -0.72077\\ 0.55488 & 0.77856\\ 1.02614 & -0.74215\\ 0.45489 & -2.00578 \end{pmatrix} \tag{6}$$

を得る。この a_j,b_j $(j=1,...,6)$ を用いて各受験者の能力値を推定していく。例えば、受験番号 1 の人は 4 番以外の問題全てに正解しているので、この人の能力が θ である確率は $p_1(\theta)p_2(\theta)p_3(\theta)(1-p_4(\theta))p_5(\theta)p_6(\theta)$

受験番号	正答パターン	正答数	能力推定値
1	111011	5	1.77846
2	101011	4	1.30327
3	011001	3	-0.740402
4	111011	5	1.77846
5	111011	5	1.77846
6	000010	1	-1.58112
7	111011	5	1.77846
8	101111	5	4.66106
9	110000	2	-2.08719
10	111000	3	-0.60450
11	111110	5	1.338884
12	010011	3	-0.978627
13	101011	4	1.303268
14	010001	2	-2.46506
15	011111	5	0.876418
16	110001	3	-1.405324
17	110111	5	-0.114244
18	011101	4	-0.433130

表 1: 能力推定値

である。この関数が最大値を取る θ の値を求めると $\theta = 1.77846$ を得るので、この人の能力の推定値は 1.77846 である。以下同様に、受験番号 1 から 18 までの受験者の能力値を推定すると表 1 の結果を得る。この表からわかるように、受験番号 1 と受験者番号 8 の受験者は共に正答数が 5 であるが、間違っている問題が異なるので能力推定値が異なっており、受験者番号 8 の受験者の方が能力値が高いと推定されている。これは IRT が正答加点方式より細かい能力推定が行えることを示している。

4 数式処理の応用

IRT の能力推定において、識別力 a_j または難易度 b_j にパラメータ k が含まれる場合を考える。能力推定を行うには $\frac{d\log(L_i(\theta))}{d\theta} = 0$ の θ に関する根を求める必要がある。この方程式は対数や指数関数を含む複雑な式なのでパラメータ k が含まれる場合は解くことができない。そこでこの式のべき級数根を求めることを考える。べき級数根の計算は記号的ニュートン法を用いれば、以下のように行える。

(i) $f(\theta) = \frac{d}{d\theta}L_i(\theta)$ と置く。

(ii) $f(0) = 0$ の根を数値的に計算し、$\alpha^{(0)}(\theta)$ と置く。$p = 1$ と置く。

(iii) 次の式で $\alpha^{(2^p)}$ を求める（$\alpha^{(2^p)}$ は $(2^p - 1)$ 次の θ のべき級数である）。ただし、下の式の除算は θ のべき級数の除算を行う。

$$\alpha^{(2^p)} \equiv \alpha^{(2^{p-1})} - \frac{f(\alpha^{(2^{p-1})})}{f'(\alpha^{(2^{p-1})})} \pmod{\theta^{2^p}} \tag{7}$$

(iv) k が十分多ければ $\alpha^{(p)}$ を返す。そうでなければ、p の値を 1 つ増やし、ステップ (iii) へ行く。

実際に識別力 a_1 にパラメータを k を入れて $a_1 = 0.49569 + k$ と置き、上の計算法で受験番号 1 の受験者の能力値を k のべき級として計算すると、

$$1.77486 - 0.667559k + 0.0637216k^2 + 1.81175k^3 - 4.83251k^4 + 8.68428k^5 - 11.806k^6 + 9.60001k^7 \quad (8)$$

を得る。これは識別力 a_1 の値が k ほど変化した時の受験者 1 の推定能力値の近似式となっている。もちろん、$k = 0$ の時はパラメータがない場合と同じなので、上のべき級数の定数項は表 1 の結果と同じである。

5 結論

項目応答理論について概説し、その実際の計算法を述べた。計算例を示し、項目応答理論を用いれば実際に正当加算方式より精緻な能力推定が行えることを示した。また、記号的ニュートン法を用いれば、システムにパラメータ k が入った場合の能力推定値をべき級数の形で計算可能であることを示した。

項目応答理論は非常に強力は能力推定法であるが、学校などの実際の現場で活用されるには解決されるべき課題も少なくない。その 1 つは部分点をいかに組み入れるかということだと思われる。今の項目応答理論では正解 (1) か不正解 (0) しかデータとして取り入れることができないが、どのようにすれば部分点 (0.5 など) をデータとして取り入れることができるかを今後は考えていきたい。

参 考 文 献

[1] Maple T.A. 公式ホームページ：http://www.cybernet.co.jp/maple/product/maple_ta/

[2] Web Mathematica　公式ホームページ：http://www.wolfram.com/products/webmathematica/

[3] STACK 日本公式ホームページ：http://ja-stack.org/

[4] manavee 公式ホームページ：http://manavee.com/

[5] 岩ヶ谷、山口, “Maple/MapleSim/Maple T.A. に見る数式処理の応用技術” 数式処理 Bulletin of JSSAC, Vol. 18, No. 2, pp. 117-125, 2012.

[6] 中村、中原、秋山, “STACK と Moodle で実践する数学 e ラーニング” 数理解析研究所講究録, Vol. 1674, pp. 40-46, 2010.

[7] 中村, “数学 e ラーニング” 東京電機大学出版局, 2010.