4F1-4

定性的で主観的な比較結果から目的関数を推定する手法

A Mathematical Programming-based Approach to Determining Objective Functions from Qualitative and Subjective Comparisons

吉住貴幸
Takayuki Yoshizumi

IBM 東京基礎研究所
IBM Research - Tokyo

The solutions or states of optimization problems or simulations are evaluated by using objective functions. The weights for these objective functions usually have to be estimated from experts' evaluations, which are likely to be qualitative and somewhat subjective. Although such estimation tasks are normally regarded as quite suitable for machine learning, we propose a mathematical programming-based method for better estimation. The key idea of our method is to use an ordinal scale for measuring paired differences of the objective values as well as the paired objective values. By using an ordinal scale, experts' qualitative and subjective evaluations can be appropriately expressed with simultaneous linear inequalities, and which can be handled by a mathematical programming solver. We show that our method outperforms machine-learning-based algorithms in a test of finding appropriate weights for an objective function.

1. はじめに

最適化やシミュレーション、コンピュータ将棋などにおいて解や状態は当該ドメインの熟練者によって評価されるものである。熟練者の知識や直感をモデル化したものが目的関数（あるいは評価関数）であるといえ、多くの場合、この目的関数は様々な評価項目の重み付き和として表される。評価関数は熟練者の知識や直感を適切に反映したものとして設計される必要があり、通常、これは目的関数の重みを適切に設定することで行われる。一旦、適切な目的関数が得られると、熟練者なしでも解の評価が可能となり、それにより様々な最適化システムやシミュレーションシステムの構築が可能となる。本稿で扱う問題は、熟練者の評価結果から目的関数を決定するという問題である。

人間の熟練者にとっては、二つの解を比較して、「解 i と解 j は同じくらい良い」、「解 i は解 j より良い」、あるいは「解 i は解 j よりもはるかに良い」などの様に定性的な評価をするのが自然で直感的な評価方法であろう。また、単一の解を評価する場合であっても、暗黙の基準となる解を思い浮かべながら、例えば「この解は基準となる解よりも良い」と定性的な評価をしているとみることもできるだろう。つまり、測定理論 [Stevens 46] の観点からみると、熟練者による評価は間隔尺度や比率尺度ではなく、順序尺度によって測定するのが合理的であると言える。しかし、このような状況下であっても、“2 つの解の差” の大きさを定量的に評価することまではできない。例えば、「解 i は解 j よりも良い」と「解 s は解 t よりもはるかに良い」という 2 つの評価があったとしても、「解 s の解 t に対する良さの程度は、解 i の解 j に対する良さの程度の 2 倍である」といった結論を下すことはできない。また、ある熟練者が「解 i は解 j よりも良い」と評価している状況で、他の熟練者は「解 i は解 j よりもはるかに良い」と評価することもあるだろう。これは、熟練者の評価は主観的であることを意味している。

この問題には、機械学習の方法論の一つであるランキング学習 [Joachims 02, Liu 09] が適しているかもしれない。しかし、熟練者の評価は、標準的な機械学習の手法で対処できる以上の情報を含むこともある。例えば、「はるかに良い」の程度は、少なくとも「良い」の程度よりは大きいと言うことはできる。しかしながら、この種の情報は、ランキング学習を含めた従来の機械学習の手法では活用することができない。一般に、熟練者による評価は労力のかかる作業であるので、熟練者の評価結果からはできるだけ多くの情報を引き出すことが望ましい。

連絡先: 吉住貴幸，IBM 東京基礎研究所，yszm@jp.ibm.com

本稿では、熟練者の評価結果からより多くの情報を引き出し、目的関数を設計する手法を提案する。本質的なアイデアは、2 つの “目的関数” だけでなく、2 つの “目的関数の差” も順序尺度を使って測ることである。これにより、熟練者の定性的で主観的な評価結果を線形連立方程式で表現することが可能となる。その結果、線形計画法、あるいは混合整数計画法を用いることで、目的関数の適切な重みを求めることができる。提案手法は、機械学習の問題に対する、数理計画法に基づくアプローチともいえる。目的関数の重みを決定するタスクにおいて、従来の機械学習に基づく手法を上回る精度が実現できることを示す。

2. 問題設定

我々が対象とする問題は、熟練者の解の評価結果から、目的関数の重みを推定することである。解は K 項目の評価指標から構成されるとし、それを K 次元のベクトル $\boldsymbol{x} = (x_1, \ldots, x_K)$ として表す。目的関数は以下の様に K 項目の評価指標の重み付き和として定義されるとする。

$$f_{\boldsymbol{w}}(\boldsymbol{x}) \equiv \sum_{1 \le k \le K} w_k x_k \tag{1}$$

ここで $w_k (\ge 0)$ は目的関数項 x_k の重みである。したがって、目的関数を決定することは、重みベクトル $\boldsymbol{w} = (w_1, \ldots, w_K)$ を決定することと等価である。

熟練者の評価は、程度の大きさを伴った順序尺度であると仮定する。これはリカート尺度 [Likert 32] の一種とみることができる。例えば、熟練者は 2 つの解 i, j を比較して、定性的な

評価を下すとする。このときの評価は、例えば以下の 3 種類のどれかである。

- 解 i と解 j とは同じくらい良い。
- 解 i は解 j よりも良い。
- 解 i は解 j よりもはるかに良い。

（もし、解 j の方が良ければ、インデックスの i と j を入れ替えて考える。）通常の順序尺度では程度の大きさまでは考慮しないが、熟練者が程度を伴った定性評価を下すことは自然であるので、本稿では程度まで考慮する。

“はるかに良い” と “良い” の境界などは、主観に依存するものであり、個人によって異なりうることも考慮する必要がある。例えば、同じ 2 つの解 i, j の比較においても、ある熟練者は「解 i は解 j よりも良い」と感じるものが、別の熟練者にとっては「解 i は解 j よりもはるかに良い」と感じることもあるだろう。したがって、複数の熟練者の評価結果から目的関数、つまり $\boldsymbol{w}$ を精度良く推定するためには、個々の熟練者ごとに異なりうる境界を推定することも必要となる。

U を熟練者の集合とし、各熟練者 $u \in U$ について、$R_{\approx}^{(u)}$、$R_{>}^{(u)}$、$R_{\gg}^{(u)}$ をそれぞれ以下の様に定義する。

- $R_{\approx}^{(u)} \equiv \{(i,j)|$ 熟練者 $u \in U$ が「解 i と解 j は同じくらい良い」と評価した。$\}$
- $R_{>}^{(u)} \equiv \{(i,j)|$ 熟練者 $u \in U$ が「解 i は解 j よりも良い」と評価した。$\}$
- $R_{\gg}^{(u)} \equiv \{(i,j)|$ 熟練者 $u \in U$ が「解 i は解 j よりもはるかに良い」と評価した。$\}$

本稿で取り扱う問題は、解集合 $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ 上で定義される熟練者の評価結果 $R_{\approx}^{(u)}$、$R_{>}^{(u)}$、$R_{\gg}^{(u)}$ $(u \in U)$ から、目的関数の重み $\boldsymbol{w}$ を推定することである。

この問題は、機械学習の問題としてとらえることが自然ではあるが、我々は数理計画法の問題として定式化し求解する。定式化の準備として、次章において、機械学習の問題を数理計画法の問題として定式化する方法ついて説明する。

3. 機械学習問題の数理計画法としての定式化

機械学習の問題が最終的に数理計画問題に帰着されることはしばしば見受けられる [Bennett 06]。LP (Linear Programming: 線形計画) とは、数理計画問題の最も基本的な形式の一つであり、目的関数、及び制約が線形の式で記述され、決定変数が実数変数のものをいう。また、決定変数に整数変数を許す、MIP(Mixed Integer Programing: 混合整数計画) と呼ばれる問題のクラスもある。ある問題が LP あるいは MIP として記述できると、LP/MIP ソルバーを用いて効率よく求解することができる。LP や MIP では、目的関数と制約を線形の式で記述する必要があり、それが問題を記述する上での大きな制限に思えるかも知れない。しかしながら、LP や MIP であっても十分な問題記述能力があることを実例を用いて示していく。

ここでは、訓練集合 $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{n}$ から、入出力関係 $y = f_{\boldsymbol{w}}(\boldsymbol{x})$ を得る機械学習の問題を考える。ここで、$\boldsymbol{x}^{(i)}$ は K 次元ベクトルであり、$\boldsymbol{x}^{(i)} = (x_1^{(i)}, \ldots, x_K^{(i)})$ と表されるとする。また、求めたい関数関係の $f_{\boldsymbol{w}}$ は、以下の様に定義されるとする。

$$f_{\boldsymbol{w}}(\boldsymbol{x}^{(i)}) \equiv \sum_{1 \le k \le K} w_k x_k^{(i)} \tag{2}$$

ここで、$\boldsymbol{w} = (w_1, \ldots, w_K)$ は重みベクトルである。つまり $f_{\boldsymbol{w}}$ を求めることは、$\boldsymbol{w}$を決定することと等価であるといえる。この問題を例とし、機械学習的アプローチによる定式化、及び数理計画法としての定式化を行い、それぞれの差異を議論する。

3.1 機械学習に基づくアプローチ

まず、この問題を機械学習の問題として捉えることを考える。損失関数を二乗誤差の最小化とするなら、以下の様に定式化するのが標準的であろう。

$$\min_{\boldsymbol{w}} \sum_{1 \le i \le N} \| f_{\boldsymbol{w}}(\boldsymbol{x}^{(i)}) - y^{(i)} \|^2 \tag{3}$$

最適な $\boldsymbol{w}$ は以下の計算により解析的に求めることができる。

$$\boldsymbol{w}^* = (X^t X)^{-1} X^t Y \tag{4}$$

ただし、X、及び Y は以下で定義されるとする。

$$X \equiv \begin{bmatrix} x_1^{(1)} & \cdots & x_K^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_K^{(n)} \end{bmatrix}, \quad Y \equiv \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}. \tag{5}$$

3.2 数理計画法に基づくアプローチ

次に、この問題を数理計画問題として捉えることを考える。ここでは、数理計画問題とは、ある制約 (通常は、等式、あるいは不等式で表現される) の元で、ある目的関数を最大化（あるいは最小化）するものを指す。この問題は、損失関数が二乗誤差を最小化するのではなく、絶対値の総和を最小化するものならば、以下の様に LP として記述可能である。

(MP1)

obj:

$$\min_{\boldsymbol{w}} \sum_{1 \le i \le N} z_i \tag{6}$$

sub. to:

$$-z_i \le f_{\boldsymbol{w}}(x^{(i)}) - y^{(i)} \le z_i, \ i = 1, \ldots, N \tag{7}$$

LP においては、全ての制約、目的関数は線形の式で表す必要があり、二乗誤差を扱うことは原理的に不可能である。しかし、LP の定式化における標準的なテクニック [Dantzig 98] を使うことで、以下の様に近似的に二乗誤差を扱うことが可能となる。

(MP2)

obj:

$$\min_{\boldsymbol{w}} \sum_{1 \le i \le N} z'_i \tag{8}$$

sub. to:

$$-z_i \le f_{\boldsymbol{w}}(x^{(i)}) - y^{(i)} \le z_i, \ i = 1, \ldots, N, \tag{9}$$

$$\max_j \{a_j z_i + c_j\} \le z'_i, \qquad i = 1, \ldots, N. \tag{10}$$

式 (10) は 2 次関数を近似する区分線形関数であり、係数の a_j と c_j は対応する 2 次関数に従って適切に設定する必要がある。なお、式 (10) の max 関数は、複数の線形の式の合成であるので、LP の範疇である。上記の例から分かるように、数理計画法の定式化のテクニックを使えば、原理的に扱うことが不可能に思える二乗損失も近似的に扱うことができる。これに加えて、ϵ 許容損失や、Huber 損失なども同様に定式化の工夫により扱うことができる。

4. 提案手法

本章では、熟練者の評価結果から目的関数の形を決定する問題を、数理計画法によるアプローチで解く手法を提案する。提案手法の本質的なアイデアは、順序尺度を 2 つの "目的関数値" を測ることに使うだけでなく、2 つの "目的関数値の差" を測ることにも使うことである。例えば、ある熟練者が解 i と解 j に関し「解 i は解 j よりも良い」と評価し、解 s と解 t に関し「解 s は解 t よりもはるかに良い」と評価したとする。この状況では、明らかに以下の式が成り立つ。

$$f\boldsymbol{w}(\boldsymbol{x}^{(i)}) > f\boldsymbol{w}(\boldsymbol{x}^{(j)}), \tag{11}$$

$$f\boldsymbol{w}(\boldsymbol{x}^{(s)}) > f\boldsymbol{w}(\boldsymbol{x}^{(t)}). \tag{12}$$

この状況であっても、"はるかに良い" の程度の大きさは、"良い" の 2 倍などと結論づけることはできない。しかし、"はるかに良い" の程度は "良い" の程度よりも少なくとも大きいということはできる。このことから以下の等式を得ることができる。

$$f\boldsymbol{w}(\boldsymbol{x}^{(s)}) - f\boldsymbol{w}(\boldsymbol{x}^{(t)}) > f\boldsymbol{w}(\boldsymbol{x}^{(i)}) - f\boldsymbol{w}(\boldsymbol{x}^{(j)}). \tag{13}$$

つまりこれは、2 つの "目的関数値の差" も順序尺度を用いて測ることができることを意味している。これにより、2 つの "目的関数値" に対してのみ順序尺度を用いるよりも多くの情報を、熟練者の評価結果から抽出することができるようになる。

次に、問題全体を数理計画問題として定式化する。式 (11), (12), 及び (13) は、決定変数である $\boldsymbol{w}$ に関して線形であるので、LP の制約式としてそのまま記述することが可能である。"同じ程度" と "良い" の境界、あるいは "良い" と "はるかに良い" の境界を表す決定変数を熟練者ごとに導入する。各熟練者 $u \in U$ に関し、"同じ程度" と "良い" の境界を $b_{u,0}$ "良い" と "はるかに良い" の境界を $b_{u,1}$ とする。これらの表記法を用い、式 (11), (12), 及び (13) で表される熟練者の定性的で主観的な評価結果は以下の連立線形不等式として表現できる。

$$\begin{aligned} -b_{u,0} < f\boldsymbol{w}(\boldsymbol{x}^{(i)}) - f\boldsymbol{w}(\boldsymbol{x}^{(j)}) + \sigma_{ij}^{(u)} &< b_{u,0}, \\ \forall (i,j) \in R_{\approx}^{(u)}, \forall u \in U&, \\ b_{u,0} < f\boldsymbol{w}(\boldsymbol{x}^{(i)}) - f\boldsymbol{w}(\boldsymbol{x}^{(j)}) + \sigma_{ij}^{(u)} &< b_{u,1}, \\ \forall (i,j) \in R_{>}^{(u)}, \forall u \in U&, \\ b_{u,1} < f\boldsymbol{w}(\boldsymbol{x}^{(i)}) - f\boldsymbol{w}(\boldsymbol{x}^{(j)}) + \sigma_{ij}^{(u)}&, \\ \forall (i,j) \in R_{\gg}^{(u)}, \forall u \in U&, \end{aligned} \tag{14}$$

ここで、$\sigma_{ij}^{(u)}$ はエラー項である。$b_{u,0}$ や $b_{u,1}$ は熟練者ごとに定義される。つまり、これは主観的な境界値を適切に定式化できていることを意味している。また、$\boldsymbol{w}$ を正規化するために以下の制約の追加が必要である。

$$\sum_{1 \le k \le K} w_k = 1 \tag{15}$$

これらの不等式（制約）の元で、以下の関数を最小化することで最適な $\boldsymbol{w}$ を得ることができる。

$$\min_{\boldsymbol{w}} \sum_{u \in U} \sum_{(i,j) \in R^{(u)}} |\sigma_{ij}^{(u)}| \tag{16}$$

ここで、$R^{(u)}$ は $R_{\approx}^{(u)}$、$R_{>}^{(u)}$、$R_{\gg}^{(u)}$ の和集合を表す。また、(MP1) の様に、絶対値の最小化は LP として記述可能である。全ての制約式、及び目的関数は線形の式で表されるためこれは LP であり、LP ソルバーを用いることで簡単に最適な $\boldsymbol{w}$ を得ることができる。なお、この定式化において、程度の数は問題ではなく、新たな程度である $R_{\ggg}^{(u)}$ や $R_{⋙}^{(u)}$ などを導入することは容易である。

5. 関連研究

我々の提案手法は機械学習の問題に対する数理計画法的アプローチであると言える。それと同時に、扱っている問題は、人間の主観的な評価をどう取り扱い、客観性のある合理的な知識を抽出するかにも関わっている。ここでは、この観点から既存研究との関連を議論する。

ランキング学習 ランキング学習 [Joachims 02, Liu 09] は、順位付けするためのアルゴリズム群であり、典型的な応用の一つは、Web search エンジンの出力の調整である。ランキング学習には想定する入力形式によりいくつかの流儀があるが、そのひとつは一対比較であり、我々の想定する問題と同じである。しかしながら、複数の程度を扱うことはできず、この部分が我々が取り扱う問題との本質的な差であると言える。

SVM ベースのアルゴリズム 仮に "良い" を 2〜4, "はるかに良い" を 4〜6 などの定量値へ対応付けできるとすると、ϵ 許容損失を用いた SVM をこの問題に適用できるかもしれない。しかしながら、事前に適切な定量値を割り当てるのは難しく、また、同じ "良い" であっても熟練者によってその定量値は異なりうるので、その点からも、SVM ベースの手法の適用は難しい。一方で提案手法は、"良い" と "はるかに良い" との境界値を熟練者ごとに最適化の結果として求めることができる。この点が提案手法の本質的なアドバンテージといえる。

6. 評価実験

ランダムに生成したデータを使って、提案手法の妥当性を検証した。熟練者は 3 人いるとし、定性評価の程度は 5 つあるとした。正しい境界値は $(b_{1,0}^*, \ldots, b_{1,3}^*) = (2,4,6,8)$, $(b_{2,0}^*, \ldots, b_{2,3}^*) = (1,2,4,8)$, $(b_{3,0}^*, \ldots, b_{3,3}^*) = (4,6,7,8)$ とした。推定する目的関数は 10 の項からなるとし、正しい重みは全て 0.1 とした。サンプルとなる解 ($\boldsymbol{x} = (x_1, \ldots, x_K)$) の各項は $\mathcal{N}(10, 8^2)$ に従う正規分布から生成した。熟練者の評価結果 ($R_{\approx}^{(u)}$, $R_{>}^{(u)}$ など) を生成するために、ランダムに 2 つの解を選択し、目的関数値の差に $\mathcal{N}(0, 0.1^2)$ に従うランダムノイズを加え、その値に応じて $R_{\approx}^{(u)}, R_{>}^{(u)}, R_{\gg}^{(u)}, R_{\ggg}^{(u)}$, $R_{⋙}^{(u)}$ のいずれかに分類した。

提案手法を比較検討するために、以下の様にベースラインとなる 2 つのアルゴリズムを作った。

Baseline 1. これは提案手法において、分類する程度を $R_{\approx}$ と $R_{>}$ の 2 つに絞ったアルゴリズムである。(他の程度である $R_{\gg}, R_{\ggg}$, $R_{⋙}$ は、すべて $R_{>}$ に分類する。)。また、熟練者を区別しないものとする。このアルゴリズムは、一対比較に基づくランキング学習のアルゴリズムとみることができる。

Baseline 2. 一般に、事前に定性評価の境界値や各程度の幅を知ることはできない。Baseline 2 アルゴリズムにおいては、定性評価の幅は全ての熟練者で同一とする。また、

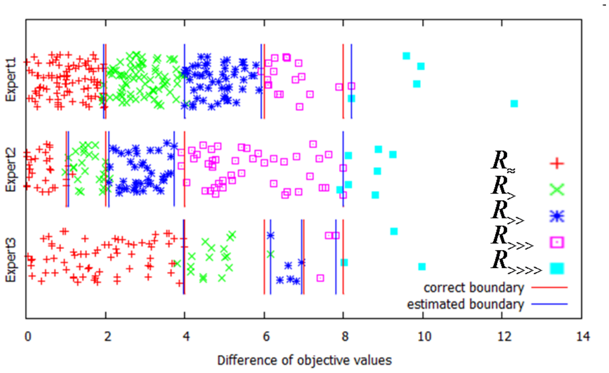

図 1: 提案手法による境界値の推定結果

熟練者による差もないものとする。これは提案手法のおいて、熟練者の区別をせず、また程度の境界値を事前に与える事に相当する。したがって Baseline 2 アルゴリズムは、ϵ 許容損失 SVM と等価とみることができる。

まず最初に、熟練者ごとに異なる境界値 ($b^*_{1,0}$ など) を個別に正しく推定できるかを調べた。先に述べた方法で、解のペアを 500 作成し、それぞれの熟練者に、20%, 30%, 50%の割合で割り当てた。なお、割合を変えたのは、ヘテロな状況でも正しく推定できるかを検証するためである。図 1 は、それぞれの熟練者ごとに、解のペアの目的関数値の差、及びそれがどの程度 ($R_{\approx}$,$R_{>}$ など) に属しているかを区別してプロットしたものである。横軸は正しい重みを使って計算した解のペアの目的関数値の差である。縦軸はそれぞれの熟練者内ではランダムに座標値を与えた。また、正しい境界線と、提案手法によって推定した境界線も示している。この図から分かるように、提案手法においては、熟練者ごとに異なる境界線を精度良く推定できていることが分かる。

次に、以下で定義する $\boldsymbol{w}$ に関する相対誤差を用いて、提案手法と 2 つのベースラインアルゴリズムとを比較評価した。

$$RelativeError(\boldsymbol{w}) \equiv \frac{1}{K}\sum_{1\leq k\leq K}\frac{|w^*_k - w_k|}{w^*_k} \qquad (17)$$

ここで w^*_k は目的関数項 k の正しい重みを表している。いくつかのサイズの訓練データを使い、各アルゴリズムを用いて計算した $\boldsymbol{w}$ の相対誤差を図 2 に示す。縦軸は相対誤差で、横軸は訓練データのサイズを表している。なお、各サイズの訓練データに関して、異なる乱数の種を使い 100 回実行しその平均値をプロットした。提案手法を Baseline1 と比較すると、全てのデータサイズにおいて、大幅な精度向上が達成されていることが分かる。提案手法と Baseline1 との本質的な差は、$\boldsymbol{w}$ を推定するために複数の境界の情報を提案手法のみが利用できることである。つまり、提案手法は同じ訓練データからより多くの情報を抽出し利用することができるので、より良い性能が達成できているとみることができる。

データサイズが 50 の場合を除いて、提案手法は Baseline2 も上回っている。データサイズが 500 の場合、提案手法の相対誤差 (1.76%) は、Baseline2 のそれ (5.20%) の概ね 3 分の 1 であり、大幅な性能向上といえる。データサイズが 50 の場合は、提案手法は Baseline2 を下回っている。提案手法においては、訓練データを 3 つのグループ (各熟練者) に分ける必要があり、各境界線を個別に推定する必要がある。結果として、実質的な訓練データのサイズが他の手法と比較して小さくなってしまう。つまり、12 の境界線 (4 つの境界線 × 3 人の熟練者) を推定するのに、50 という訓練データは少なすぎると言うことである。したがって、これはアルゴリズムの欠点と言うよりも、データの問題と言うべきものである。

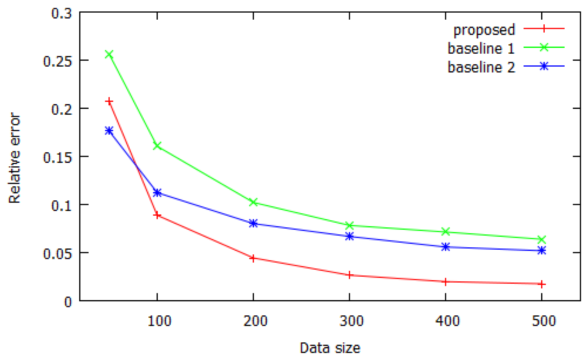

図 2: 各アルゴリズムの相対誤差

7. おわりに

本稿では熟練者の定性的で主観的な評価結果から目的関数を推定する方法を提案した。提案手法は機械学習の問題を数理計画法の問題として捉える方法論であるとも言える。この方法論を用いることで、目的関数の形を熟練者の評価結果から推定する問題を、機械学習に基づく手法よりも精度良く推定できることを示した。本手法は、整数変数を導入することで目的関数項の基底関数を自動的に選択することも可能である。この基底関数に関する拡張や、より詳細な実験結果については [Yoshizumi 15] を参照されたい。

参考文献

[Bennett 06] Bennett, K. P. and Parrado-Hernández, E.: The interplay of optimization and machine learning research, *The Journal of Machine Learning Research*, Vol. 7, pp. 1265–1281 (2006)

[Dantzig 98] Dantzig, G. B.: *Linear programming and extensions*, Princeton university press (1998)

[Joachims 02] Joachims, T.: Optimizing search engines using clickthrough data, in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 133–142ACM (2002)

[Likert 32] Likert, R.: A technique for the measurement of attitudes., *Archives of psychology* (1932)

[Liu 09] Liu, T.-Y.: Learning to rank for information retrieval, *Foundations and Trends in Information Retrieval*, Vol. 3, No. 3, pp. 225–331 (2009)

[Stevens 46] Stevens, S. S.: On the theory of scales of measurement, *Science*, Vol. 103, No. 2684, pp. 667–680 (1946)

[Yoshizumi 15] Yoshizumi, T.: A Mathematical Programming based Approach to Determining Objective Functions from Qualitative and Subjective Comparisons, in *AAAI*, p. in Press (2015)