A Geometric Least Squares Method for Peer Assessment

Mingyu Xiao
University of Electronic Science and Technology of China
Chengdu, China
myxiao@gmail.com

Yuqing Wang
University of Electronic Science and Technology of China
Chengdu, China
yoking0519@gmail.com

Binglin Tao
University of Electronic Science and Technology of China
Chengdu, China
tblmobi@163.com

ABSTRACT

In the peer assessment problem, a set of agents give evaluations to each other, and we are going to combine these peer assessments together to construct an overall evaluation. In this paper, we propose a geometric least squares method (GLS) to find an aggregate scoring overall agents for the peer assessment problem. Our method is based on the following observation. Since each agent has a missing score that should be given by itself, we consider the missing score as a variable and then each agent can be regarded as a line in an n-dimensional vector space. The final aggregate scores of the agents can be regarded as points on a line vector, called the *projection vector*. Thus, we treat the peer assessment problem as an optimization problem of selecting a projection vector with minimum total squared distance to all the lines representing the agents. We will see that this aggregate method has some advantages compared with the simple average method. One advantage is that, when the scores given by each agent (even ignoring the magnitude of the agent) are close to a groundtruth, the new method finds the groundtruth with the highest expectation.

KEYWORDS

Social Choice; Score Aggregation; Peer Assessment

ACM Reference Format:

Mingyu Xiao, Yuqing Wang, and Binglin Tao. 2018. A Geometric Least Squares Method for Peer Assessment. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 3 pages.

1 INTRODUCTION

The *peer assessment problem* has been well known for many years in social choice [1]. In the setting, each agent is both a candidate and a voter, and each agent is asked to assign numeric scores from its own point of view regarding the overall performance of other agents. The object is to get a combined overall evaluation of the agents. This problem has many important application scenarios [2, 7–12] and has been extensively studied in the literature [4, 5, 12, 13]. Peer assessments suffer from several fundamental problems, such as, how to set an incentive or define a mechanism to ensure that agents report their evaluations truthfully; after gaining the scores, how to aggregate the scores into a coherent assessment; and so on. In this paper, we focus on aggregating scores for the peer assessment problem.

A related problem is to obtain an aggregate rank of the agents based on the rankings given by the agents. For rank aggregation problems, there are some famous protocols, such as Kemeny [3] and Condorcet [16], which is to minimize the number of disagree pairs.

However, for score aggregation problems, there are few well-known protocols used to evaluate the aggregation results. Not all score aggregation problems in real life can be changed to rank aggregation problems, for example, to compute GPA of students. Most of references focus on the mechanism designs or how to normalize the scores. After doing these, they simply apply the sum-up or weight sum-up method to aggregate. In this paper, we study the aggregate models based on the given scores from agents and build a protocol for peer assessments inspired by the idea in [14].

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

2 THE PROBLEM AND DEFINITIONS

In the peer assessment problem, there is a set of agents $N = \{1, 2, \dots, n\}$. Each agent $j \in N$ has evaluated each other agent $i \in N \setminus \{j\}$ by giving a nonnegative score s_{ij}. We need to aggregate the n evaluations into a final overall vector of scores for the agents. The evaluation given by an agent i can be denoted by a column vector $\{s_{1i}, s_{2i}, \dots, s_{(i-1)i}, \text{-}, s_{(i+1)i}, \dots, s_{ni}\}^T$. Here we use "-" to denote the missing score at the ith site. All the evaluations form a matrix $S_{n\times n}$, called the *score matrix*. In the score matrix $S_{n\times n}$, all the elements on the diagonal are "-" denoting missing elements. We may also view each missing element as a variable and use t_i to denote them.

3 THE MODEL AND ALGORITHM

We consider the peer assessment problem from the geometric perspective. We look at the n-dimensional vector space $\mathcal{A}$ with base vectors, denoted by $\boldsymbol{e}_1, \boldsymbol{e}_2, \dots, \boldsymbol{e}_n$, representing the n agents. Each agent has a missing score that should be given by itself and we view it as a variable. Thus, each agent can be represented by a line l_i in the n-dimensional vector space $\mathcal{A}$ and this line l_i is parallel with the base vector $\boldsymbol{e}_i$ corresponding to itself. The final aggregate result is a vector, denoted by $\boldsymbol{e}$ and the final score for each agent i is the ith element of $\boldsymbol{e}$. The aggregation problem can be interpreted as a geometric problem to "project" n lines l_i in an n-dimensional vector space $\mathcal{A}$ to n points in a 1-dimensional unit vector space $\boldsymbol{e}/||\boldsymbol{e}||_2$. We still need to give the precise definition of "project". We say a line l_i projects onto a point c_i on a vector space $\boldsymbol{e}'$ if and only if c_i is a point on $\boldsymbol{e}'$ that minimizes the distance from c_i to the line l_i.

The average method is, for each agent, to use the average of its $n-1$ scores given by other agents as its aggregate score. In the geometric model, the average method is exactly to project the n lines onto n points on the 1-dimensional vector space $\boldsymbol{e}' = (1/\sqrt{n}, 1/\sqrt{n}, \ldots, 1/\sqrt{n})$. See Figure 1 for an illustration. However, why should we always select this diagonal vector space to be projected onto? Our idea is to select an "overall optimal" vector space to be projected onto. Thus, we consider the peer assessment problem as an optimization problem of selecting a projection vector $\boldsymbol{e}'$ such that the total squared distance from $\boldsymbol{e}'$ to the n lines l_i is minimized. We will call this method the *geometric least squares method* (GLS). Next, we describe the detailed steps.

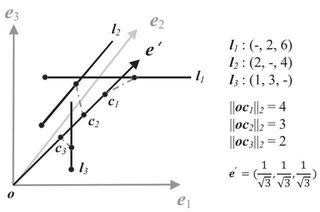

Figure 1: An illustration for the projection

A line in an n-dimensional vector space $\mathcal{A}$ can be represented by a parameter function $l = \boldsymbol{e}t + \boldsymbol{p}$, where t is the parameter, $\boldsymbol{p}$ is the original point (the point with $t = 0$ on the line) and $\boldsymbol{e}$ is a vector deciding the direction of the line.

Recall that $\boldsymbol{e}_i$ is the base vector with the ith element being 1 and all other elements being 0. Let $\boldsymbol{p}_i$ denote the point $(s_{i1}, s_{i2}, \ldots, s_{i(i-1)}, 0, s_{i(i+1)}, \ldots, s_{in})$. The n lines l_i representing the n agents will be denoted by

$$l_i = \boldsymbol{e}_i t_i + \boldsymbol{p}_i, i \in \{1, 2, \ldots, n\},$$

where t_i are the variables. We will denote the vector $\boldsymbol{e}$ to be projected onto as a line l_0 going through the original point $\mathbf{0} = (0, 0, \ldots, 0)$: $\boldsymbol{e} = l_0 = \boldsymbol{e}_0 t_0 + \mathbf{0}$, where t_0 is the variable.

The distance $dist(l'_i, l'_j)$ between two lines $l'_i = \boldsymbol{v}_i t'_i + \boldsymbol{q}_i$ and $l'_j = \boldsymbol{v}_j t'_j + \boldsymbol{q}_j$ in a high-dimensional space is defined to be the minimum distance between any two points lying on the lines, which can be computed by

$$dist(l'_i, l'_j) = \min_{t'_i, t'_j} ||l'_i - l'_j||_2 = \min_{t'_i, t'_j} ||\boldsymbol{v}_i t'_i - \boldsymbol{v}_j t'_j + \boldsymbol{q}_i - \boldsymbol{q}_j||_2.$$

Directed computation shows that

$$dist(l_0, l_i) = ||\frac{\boldsymbol{p}_i \boldsymbol{e}_0^T \boldsymbol{e}_0 - \boldsymbol{e}_i \boldsymbol{e}_0^T \boldsymbol{e}_0 \boldsymbol{e}_i^T \boldsymbol{p}_i}{1 - (\boldsymbol{e}_0 \boldsymbol{e}_i^T)^2} - \boldsymbol{p}_i||_2.$$

The object of GLS is to find a line l_0 minimizing the total squared distance between l_0 and the n lines l_i

$$E = \sum_{i=1}^{n} dist^2(l_0, l_i) = \sum_{i=1}^{n} ||\frac{\boldsymbol{p}_i \boldsymbol{e}_0^T \boldsymbol{e}_0 - \boldsymbol{e}_i \boldsymbol{e}_0^T \boldsymbol{e}_0 \boldsymbol{e}_i^T \boldsymbol{p}_i}{1 - (\boldsymbol{e}_0 \boldsymbol{e}_i^T)^2} - \boldsymbol{p}_i||_2^2.$$

Thus, it is to find a unit vector $\boldsymbol{e}_0$ satisfying

$$\boldsymbol{e}_0 = \arg\min_{l_0} E = \arg\min_{l_0} \sum_{i=1}^{n} dist^2(l_0, l_i).$$

We only need to solve the above optimization problem. Let $\boldsymbol{e}_0 = (e_{01}, e_{02}, \ldots, e_{0n})$. When E achieves the minimum value, the derivatives of E with respect to $\boldsymbol{e}_0$ is zero. Thus, the partial derivatives of E with respect to each component of $\boldsymbol{e}_0$ is zero, i.e., $\frac{\partial E}{\partial e_{0i}} = 0, i \in \{1, 2, \ldots, n\}$.

By solving the above function set, we can get the projection vector $\boldsymbol{e}_0$. However, the solution is too complex to obtain an explicit expression directly. There exists some numerical methods to solve these kinds of partial derivatives and implements of them are available in most famous mathematical softwares. We implemented the algorithm in Matlab and the code is available online [15].

4 PROPERTY ANALYSIS

Relations With Maximum Likelihood Estimate

The object of GLS is to optimize the total squared distance. This criteria has a good meaning in probabilistic semantics.

Assume that there is a groundtruth score vector $\hat{\boldsymbol{c}}_i$ for each agent i. The actual score of agent i given by agent j is an error estimation of the jth element of $\hat{\boldsymbol{c}}_i$. The actually score vector $\boldsymbol{c}_i$ of agent i is

$$\boldsymbol{c}_i = \hat{\boldsymbol{c}}_i + \boldsymbol{\varepsilon}_i,$$

in which $\boldsymbol{\varepsilon}_i$ is the error vector. For groundtruth, we assume that all the agents agree with each other consistently ignoring the magnitude, i.e., the groundtruth score vectors for all agents satisfy the linear relationship.

Property 1. *Under the above assumption, if the error obeys the Gaussian distribution, then to minimize the total squared distance is equal to maximize the likelihood function.*

It is reasonable to assume the error obeys the Gaussian distribution, because the Center Limit Theorem shows if an error is the sum of many small, independent error sources, the total error will be about Normally distributed (Gaussian distribution) [6].

A Stable Property

Property 2. *Given a consistent score matrix. If any agent changes its given scores by multiplying them by a positive constant γ, then the new final aggregate score vector obtained by our method is a constant multiple of the old one.*

This property says that the relationship of the agents in the final aggregate score vector will not change if one agent amplifies or shrinks its magnitude without changing the relationship of the agents in its given scores. This provides an incentive for agents to avoid grading all others low.

ACKNOWLEDGEMENT

The work is supported by the National Natural Science Foundation of China, under grants 61772115 and 61370071.

REFERENCES

[1] Kenneth J Arrow. 2012. *Social choice and individual values.* Vol. 12. Yale university press.
[2] Nicolas Bousquet, Sergey Norin, and Adrian Vetta. 2014. A near-optimal mechanism for impartial selection. In *International Conference on Web and Internet Economics(WINE)*. Springer, 133–146.
[3] Vincent Conitzer, Andrew Davenport, and Jayant Kalagnanam. 2006. Improved bounds for computing Kemeny rankings. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference (AAAI)*, Vol. 6. 620–626.
[4] Luca De Alfaro and Michael Shavlovsky. 2014. CrowdGrader: A tool for crowdsourcing the evaluation of homework assignments. In *Proceedings of the 45th ACM technical symposium on Computer science education(SIGCSE)*. ACM, 415–420.
[5] Luca De Alfaro, Michael Shavlovsky, and Vassilis Polychronopoulos. 2016. Incentives for Truthful Peer Grading. *arXiv preprint arXiv:1604.03178* (2016).
[6] William Feller. 2015. The fundamental limit theorems in probability. In *Selected Papers I*. Springer, 667–699.
[7] Ron Holzman and Hervé Moulin. 2013. Impartial nominations for a prize. *Econometrica* 81, 1 (2013), 173–196.
[8] Thorsten Joachims and Karthik Raman. 2016. Bayesian ordinal aggregation of peer assessments: A case study on KDD 2015. In *Solving Large Scale Learning Tasks. Challenges and Algorithms.* Springer, 286–299.
[9] David Kurokawa, Omer Lev, Jamie Morgenstern, and Ariel D Procaccia. 2015. Impartial peer review. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI)*. 582–588.
[10] Karim R Lakhani, David A Garvin, and Eric Lonstein. 2010. Topcoder (a): Developing software through crowdsourcing. *Harvard Business School General Management Unit Case No. 610-032* (2010).
[11] Michael R Merrifield and Donald G Saari. 2009. Telescope time without tears: a distributed approach to peer review. *Astronomy & Geophysics* 50, 4 (2009), 4–16.
[12] Toby Walsh. 2014. The peerrank method for peer assessment. In *Proceedings of the Twenty-first European Conference on Artificial Intelligence(ECAI)*. IOS Press, 909–914.
[13] James R Wright, Chris Thornton, and Kevin Leyton-Brown. 2015. Mechanical ta: Partially automated high-stakes peer grading. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE)*. ACM, 96–101.
[14] Mingyu Xiao and Yuqing Wang. 2017. Score Aggregation via Spectral Method. In *Proceedings of The Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI)*. 451–457.
[15] Mingyu Xiao and Yuqing Wang. 2018. GLS. https://github.com/echo23333/GLS. (2018).
[16] H. P. Young. 1988. Condorcet's Theory of Voting. *American Political Science Review* 82, 4 (1988), 1231–1244.