AUT Journal of Mathematics and Computing

AUT J. Math. Comput., 4(2) (2023) 137-144
https://doi.org/10.22060/ajmc.2022.20659.1072

Original Article

Detecting semi-referees in the Plackett-Luce model

Omid Naghshineh Arjmand*[a], Nastaran Mirzaei[a]

[a]*Department of Mathematics and Computer Science Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran*
[b]*Shahrood University of Technology, Iran*

ABSTRACT: In this paper, we use the Plackett-Luce model for detecting some referees who judge arbitrariness with inaccuracy or without paying enough attention, which is called Semi-referees. We will investigate our method by simulation and sample of real data.

Review History:

Received:13 October 2021
Revised:03 December 2022
Accepted:07 December 2022
Available Online:20 March 2023

Keywords:

Ranking
Semi-referee
Plackett-Luce
Bradley-Terry

1. Introduction

The issue of referee and ranking is a subject that has many uses in stock, racing, reviewing articles, ranking web pages or social networks, and so on. In this paper, we first introduce the Bradley-Terry model that is used to rank a set of objects (articles, teams, individuals, or companies) that are pairwise compared. Then we introduce the generalization of this model, known as the Plackett-Luce model, which is used to rank several objects that are simultaneously compared. This model is expressed using simulation with an exponential random variable, and its parameters are estimated using the maximum likelihood method. Then we introduce a method for presenting semi-referees and survey our method by simulation and real data.

1.1. Bradley-Terry Model

In 1952 Bradley and Terry[1], introduced a probabilistic model for ranking objects that is known as the Bradley-Terry model. Of course, In 1920, Zeremlou also studied this model. This model is based on pairwise comparisons [4]. If i and j are two distinct objects of the community, the probability i is preferred to j :

$$\mathrm{P}(i \succ j) = \frac{\pi_i}{\pi_i + \pi_j}. \tag{1}$$

In this relation $\pi_i > 0$ and $\pi_j > 0$.

The Bradley-Terry model is used in statistical, sporting, machine learning, etc.

*Corresponding author.
E-mail addresses: naghshineh@aut.ac.ir, n.mirzaei@aut.ac.ir

1.1.1. Estimation of Bradley-Terry Model Parameters Using Maximum Likelihood Method

The maximum likelihood method answers the question, which is the maximum probability of occurrence of the observed sample for each vector of parameters. In this model, there is a sequence of independent experiments that in each experiment, the object i with the probability π_{ij} is preferred to the object j. In fact, in this model, the observed sample is the number of times object i is preferred to object j. (For all i, j).
Assume there are corresponding parameters $\lambda_1, \lambda_2, \ldots, \lambda_n$ for each of the objects $i, j = 1, \ldots, n$. The likelihood function is:

$$L(\Lambda) = \prod_i \prod_j \left(\frac{\lambda_i}{\lambda_i + \lambda_j}\right)^{w_{ij}} \tag{2}$$

w_{ij} is the number of times the object i is preferred to j and $w_{ii} = 0$.
If all the parameters are multiplied by a constant value, the likelihood function does not change; in other words, the values of λ_i are essential to each other. So we always solve this problem by setting λ_1 by 1.

The logarithm of the likelihood function is equal to

$$l(\Lambda) = \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} \log \frac{\lambda_i}{\lambda_i + \lambda_j} \tag{3}$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} (w_{ij} \log \lambda_i - w_{ij} \log(\lambda_i + \lambda_j)) \tag{4}$$

for $i \geq 2$, we derive the value of λ_i and put the resulting expression equal to zero.

$$\frac{\partial}{\partial \lambda_i} l(\Lambda) = \sum_j \frac{w_{ij}}{\lambda_i} - \sum_j \frac{N_{ij}}{\lambda_i + \lambda_j} = 0 \tag{5}$$

In this respect, the total number of comparisons between the object i and the object j is $N_{ij} = w_{ij} + w_{ji}$. This equation system can not be solved analytically. The use of a recursive algorithm is useful for solving this system; for $i \geq 2$

$$\lambda_i = \frac{\sum_{j=1}^{n} w_{ij}}{\sum_{j=1}^{n} \frac{N_{ij}}{\hat{\lambda}_i + \hat{\lambda}_j}} \tag{6}$$

We continue doing this until it converges numerically. In 1957, Ford [2] proved that this recursive algorithm is convergent under the following assumption. "In any addition of the set of objects to two categories, some objects from the first category must be preferred at least once over some objects from the second category.

1.2. Plackett-Luce Model

The Plackett-Luce model is the extension of the Bradley-Terry model and is used to rank multiple objects together [3]. If a referee compares m of n object, the result of the Judgment can be shows:

$$D = (D_1 \succ D_2 \succ \cdots \succ D_m) \tag{7}$$

In this Model, the probability of superiority of objects relative to each other is calculated as follows:

$$\mathrm{P}(D_1 \succ D_2 \succ \cdots \succ D_m) = \prod_{j=1}^{m} \frac{\lambda_{D_j}}{\sum_{k=j}^{m} \lambda_{D_k}} \tag{8}$$

1.2.1. Simulation of Plakett-Luce Model by Exponential Random Variables

Let $X_1, X_2, \ldots, X_n$ be independent random variable, with X_i having an Exponential parameters λ_i distribution. Suppose Objects i is preferred to objest j if $X_i < X_j$ then the probability that $1 \succ 2 \succ \cdots \succ n$ is calculated as follows:

$$
\begin{aligned}
\mathrm{P}\left(X_1<\cdots<X_n\right) &= \int_0^{\infty} f_{X_1}(t)\mathrm{P}\left(t<X_2<\cdots<X_n\right)dt \\
&= \int_0^{\infty} f_{X_1}(t)\mathrm{P}\left(t<\min\left(X_2,\ldots,X_n\right), X_2<\cdots<X_n\right)dt \\
&= \int_0^{\infty} \lambda_1 e^{\lambda_1 t}\mathrm{P}\left(t<\min\left(X_2,\ldots,X_n\right)\right)\mathrm{P}\left(X_2<\cdots<X_n \mid t<\min\left(X_2,\ldots,X_n\right)\right)dt \\
&= \frac{\lambda_1}{\lambda_1+\lambda_2+\ldots+\lambda_n}\times \mathrm{P}\left(X_2<\cdots<X_n\right) \\
&= \frac{\lambda_1}{\lambda_1+\lambda_2+\ldots+\lambda_n}\times\frac{\lambda_2}{\lambda_2+\ldots+\lambda_n}\times \mathrm{P}\left(X_3<\cdots<X_n\right) \\
&= \ldots \\
&= \frac{\lambda_1}{\lambda_1+\lambda_2+\ldots+\lambda_n}\times\frac{\lambda_2}{\lambda_2+\ldots+\lambda_n}\times\cdots\times\frac{\lambda_n}{\lambda_n}.
\end{aligned}
$$

So we can simulate Plackett-Luce model by exponential random variables.

1.2.2. Estimation of Plakett-Luce Model Parameters Using Maximum Likelihood Method

Assume that the parameters $\lambda_1,\ldots,\lambda_n$ corresponding to objects $1,2,\ldots,n$ exist.
We calculate the probability value of the arbitrary vector as follows:

$$
\mathrm{P}(D)=\frac{\lambda_{D_1}}{\lambda_{D_1}+\cdots+\lambda_{D_m}}\times\frac{\lambda_{D_2}}{\lambda_{D_1}+\cdots+\lambda_{D_m}}\times\cdots\times\frac{\lambda_{D_m}}{\lambda_{D_m}} \tag{9}
$$

The likelihood function is written as:

$$
l(\Lambda)=\sum_j \log\mathrm{P}(D^j) \tag{10}
$$

The meaning of D is the j-th referees judgment.

As previously explained, we solve the $\lambda_1=1$ problem with the addition of the kernel.
We derive λ_i for $i\geq 2$ to maximize of the relation (9) and put the resulting expression equal to zero.

$$
\begin{aligned}
\frac{\partial\log\mathrm{P}(D)}{\partial\lambda_i} &= \frac{\partial}{\partial\lambda_i}\begin{pmatrix}\log\lambda_i-\log\left(\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_k}\right)-\log\left(\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_{k+1}}\right)\\ +\cdots+\log\left(\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_m}\right)\end{pmatrix} \\
&= \frac{1}{\lambda_i}-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_k}}-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_{k+1}}}-\cdots-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_m}} \\
\frac{\partial l}{\partial\lambda_i} &= \sum_{\substack{D\\ i\in D\\ i=D_k}}\left(\frac{1}{\lambda_i}-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_k}}-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_{k+1}}}-\cdots-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_m}}\right) \\
&= \frac{n_i}{\lambda_i}-\sum_{\substack{D\\ i\in D\\ i=D_k}}\left(\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_k}}-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_{k+1}}}-\cdots-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_m}}\right).
\end{aligned}
$$

This equation system can not be calculated analytically. So, we use the recursive algorithm to solve this device and, for $i\geq 2$, we set λ_i equal to

$$
n_i\Bigg/\sum_{\substack{D\\ i\in D\\ i=D_k}}\left(\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_k}}-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_{(k+1)}}}-\cdots-\frac{1}{\lambda_{D_1}+\lambda_{D_2}+\cdots+\lambda_{D_m}}\right). \tag{11}
$$

We continue doing this until it converges numerically.

2. Statement of the Problem of Identifying Semi-referees

In modeling and solving a ranking problem, it is implicitly assumed that all referees have similar tastes and accuracy. The random nature of things somehow causes the difference in the judgment of the two referees. In many real issues, some referees judge things absolutely randomly. We call this group semi-referees and want to identify the semi-referee group.

2.1. Semi-referees Detection Algorithm

For detecting semi-referees, assume that each referee has m object for judgment, which at first, all the referees have judged really. Then we estimate the merit parameters with the maximum likelihood method. By using these parameters, we calculate the likelihood estimator of each referee.

For each referee, the null hypothesis is that he/she has really refereed, and the alternative hypothesis is the semi-referee. We compare the estimated likelihood value with $\frac{1}{m!}$ (The probability of an arbitrary permutation from all m permutations).

This loop ends when the null hypothesis or the alternative hypothesis does not change for any of the referees. The steps described in Figure 1 are shown.

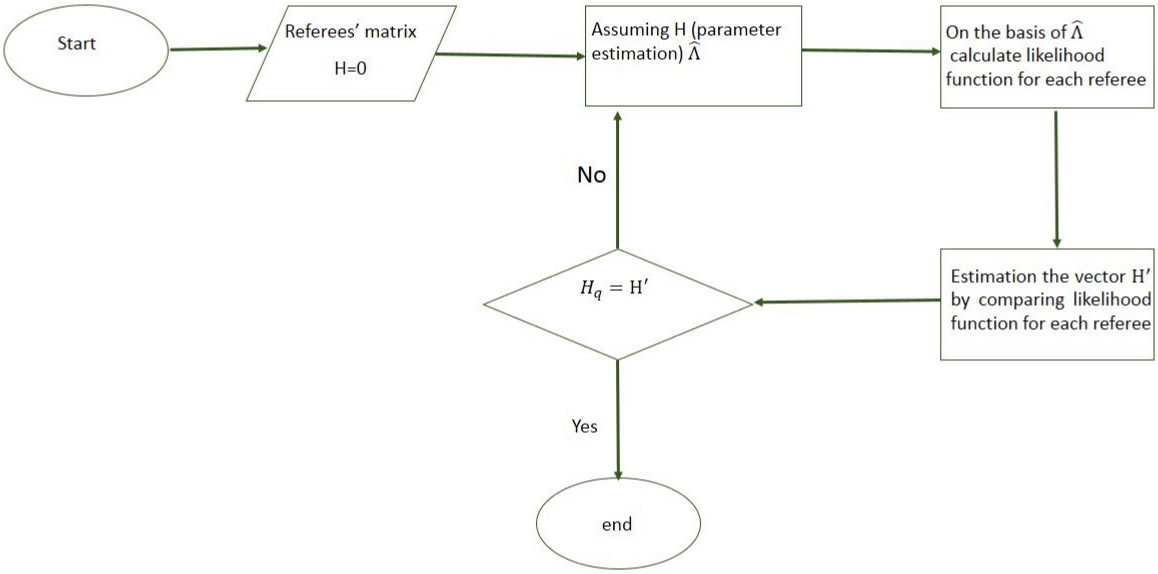

Figure 1: Referee and Semi-referee Identification Algorithm

3. Simulation

We simulate a matrix with 100 rows for the number of referees and 8 columns for the number of objects. Each referee has ranked 5 objects. We estimate an exponential random variable for each object with the merit parameter (λ) of that object. We consider 4 cases for simulation:
The first case $\lambda_i = 4^{i-1}, i = 1, \ldots, n$.
The second case $\lambda_i = 2^{i-1}, i = 1, \ldots, n$.
The third case $\lambda_i = 1.2^{i-1}, i = 1, \ldots, n$.
The fourth case $\lambda_i = i, i = 1, \ldots, n$.

3.1. The First Case

When put $\lambda_i = 4^{i-1}, i = 1, \ldots, 8$, comparing two consecutive objects equals the probability of 0.2 in favor of the first and 0.8 in favor of the second one. In each row, we attribute the smallest value of the exponential distribution to the number 1 and the largest value to 5.
First, we start when the number of semi-referees is 20, and the result is shown in the following table:

As we can see, in this implementation, we have 20 semi-referees. The algorithm correctly detects 20 semi-referees, but mistakenly considers 1 referee as a semi-referee if the number of semi-referees is equal to 10, 20, 30, 40, and 50. After 1000 repetitions, the following results are obtained:

Table 1: Semi-referee

Result \ Fact	Referee	Semi-referee
Referee	79	0
Semi-referee	1	20

Table 2: Referee and Semi-referee Identification table

$\lambda_i = 4^{i-1}$ $d = 100$ n=8 m=5 times=100					
Number of Semi-referees	10	20	30	40	50
The Mean of Number of Semi-referees the Algorithm Correctly Detects	9.999	19.999	29.995	38.674	14.931
The Mean of Number of Referees the Algorithm Correctly Detects	89.585	79.825	69.812	58.906	24.498

The above table shows the mean of referees and semi-referees that the algorithm detects after 1000 performances. For example, when semi-referees are 30, the mean of semi-referees that the algorithm detects is 29.995, and the mean of referees that detects is 69.812.

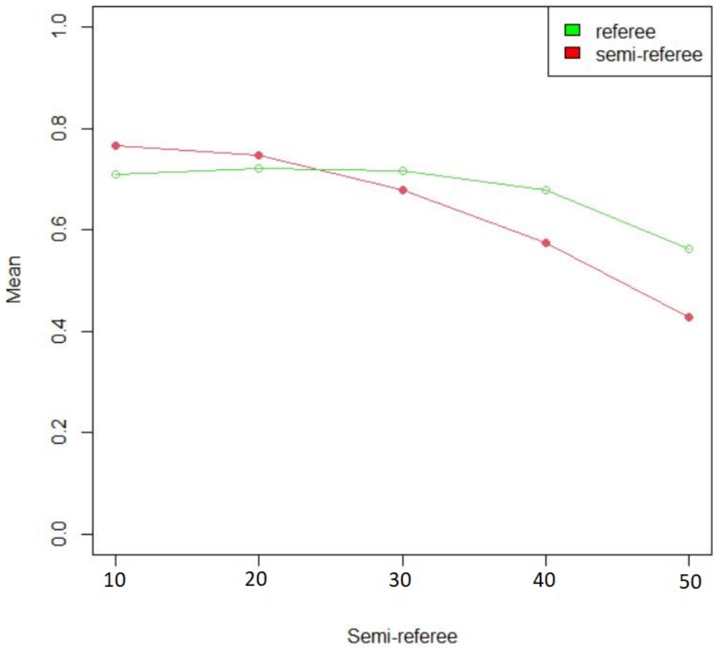

Figure 2: The First Case

In figure 2, the number of semi-referees is the horizontal axis, and the vertical axis represents the ratio of the number of semi-referees identified by the algorithm.

3.2. The Second Case

We consider the $\lambda_i = 2^{i-1}, i = 1, \ldots, n$ and semi-referees is equal to 10, 20, 30, 40, 50 after 1000 iterations. The results are illustrated in the following chart:

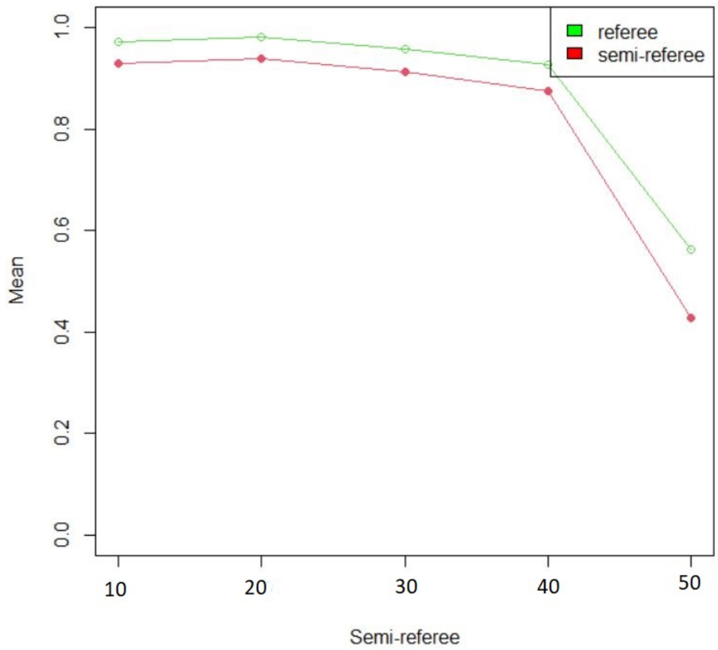

Figure 3: The Second Case

3.3. The Third Case

We consider $\lambda_i = 1.2^{i-1}, i = 1, \ldots, n$ and semi-referees is equal to 10, 20, 30, 40, 50 after 1000 iterations. The results are illustrated in the following chart:

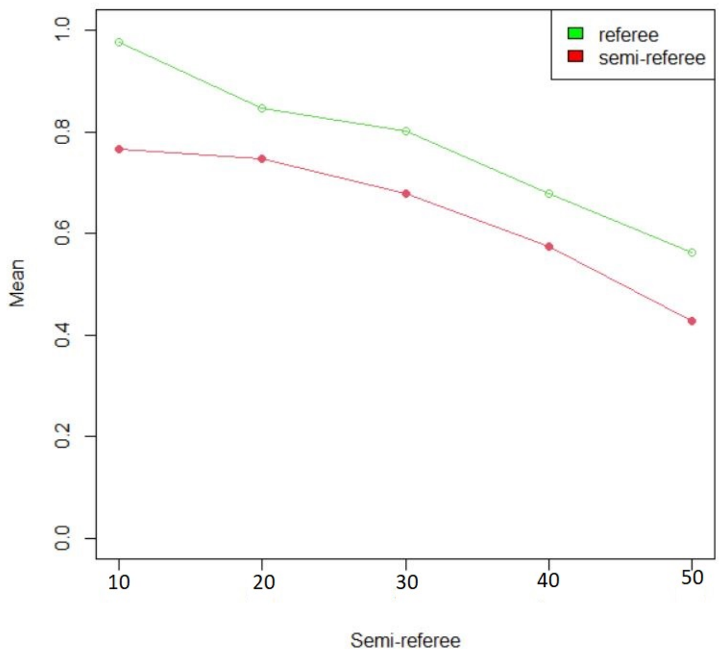

Figure 4: The Third Case

3.4. The Fourth Case

The $\lambda_i = i, i = 1, \ldots, n$ and semi-referees is equal to 10, 20, 30, 40, 50 after 1000 iterations.
The results are illustrated in the following chart:

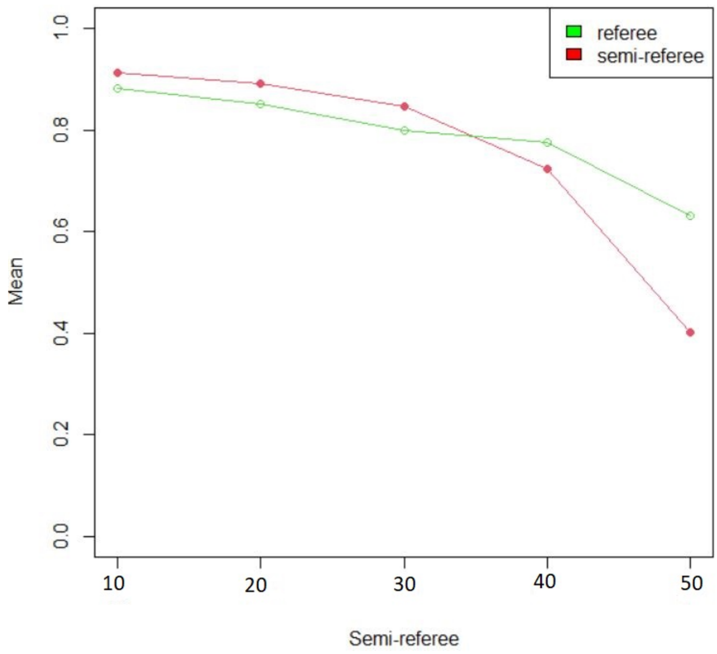

Figure 5: The Fourth Case

3.5. Referee Matrix by Real Data

In this section, 124 articles were sent for the conference of the National Elite Foundation (selected students of the country's universities). 136 people who were among the participants judged the articles. The referees viewed and rated the articles based on the order we specified. To identify the best articles by the reviewers, each reviewer compared 5 articles. The following results were obtained after calculations.

```
1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 1 0 1 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0
0 0 0 0 0 0 1 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
```

After calculating, we find the algorithm considers 22 out of 136 people to be semi-referees.
(0 means referee, and 1 means semi-referee).

Conclusion

We introduced the Plackett-Luce Model, which is used for multiple comparisons, and estimated the parameters of this model using the maximum likelihood estimator. The simulation, which was repeated and performed 1000, checked and plotted each case, and the following results were obtained: When we consider the merit parameter equal to 4 raise to the power of column number, as long as we increase the semi-referee to 40%, the algorithm has high accuracy in identifying the clauses. However, It drops sharply from 40% on the chart, and semi-referees are not identified correctly. When we set the parameter λ equal to 2, the power of the number corresponding to the object (column number), as long as we increase the semi-referees to 40% , the algorithm has high accuracy in identifying semi-referees. When we set the parameter λ equal to 1.2, the power of the number corresponding to the object (column number), as long as we increase the semi-referees by 30%, the algorithm has high accuracy in identifying semi-referees. When we set the parameter λ equal to the number corresponding to the object (column number), the algorithm has high accuracy in identifying the semi-referees as long as we increase the semi-referees to 30%.

According to the simulation results, the algorithm was very accurate, and then we used real data to identify the semi-referees. It was observed that 22 out of 136 are semi-referees.

References

[1] R. A. Bradley and M. E. Terry, *Rank analysis of incomplete block designs. I. The method of paired comparisons*, Biometrika, 39 (1952), pp. 324–345.

[2] L. R. Ford, Jr., *Solution of a ranking problem from binary comparisons*, Amer. Math. Monthly, 64 (1957), pp. 28–33.

[3] D. R. Hunter, *MM algorithms for generalized Bradley-Terry models*, Ann. Statist., 32 (2004), pp. 384–406.

[4] E. Zermelo, *Die Berechnung der Turnier-Ergebnisse als ein Maximumproblem der Wahrscheinlichkeitsrechnung*, Math. Z., 29 (1929), pp. 436–460.

Please cite this article using:

Omid Naghshineh Arjmand, Nastaran Mirzaei, Detecting semi-referees in the Plackett-Luce model, AUT J. Math. Comput., 4(2) (2023) 137-144
https://doi.org/10.22060/ajmc.2022.20659.1072