Rank Aggregation Methods for Midwest Machine Learning 2017 Poster Competition

Amanda Bower, Lalit Jain, Laura Balzano

June 2017

This report explains the three rank aggregation methods (Kemeny aggregation, Borda count, and the Bradley-Terry-Luce model) used in the 2017 Midwest Machine Learning Symposium poster competition. The goal of rank aggregation is to take an initial set of rankings on a set of items and output a *single* ranking on all of the items that captures the initial rankings the "best."

In all that follows, n refers to the number of items being ranked and m refers to the number of (full or partial) rankings. We say σ is a full ranking if it is a permutation of the n items. For instance, if $n = 3$ and $\{1, 2, 3\}$ is the set of items, then one full ranking of $\{1, 2, 3\}$ is $(2, 3, 1)$ which means item 2 is preferred to item 3, which is preferred to item 1. Similarly, a partial ranking σ is a permutation of some subset I_σ of the items.

In the rank aggregation methods that follow, we will frequently use pairwise comparisons between two items. Note that any ranking on k items determines $\binom{k}{2}$ pairwise comparisons. For instance, considering the above example again, we get $\binom{3}{2} = 3$ pairwise comparisons: item 2 is preferred to item 3, item 2 is preferred to item 1, and item 3 is preferred to item 2. Given a full or partial ranking σ, we say $\sigma(i) > \sigma(j)$ if item i is preferred to item j under the ranking σ. Finally, note that in this setup, we do not allow ties between items.

1 Kemeny Aggregation

In short, Kemeny aggregation proposes that the best ranking is one that minimizes its Kendall tau distance with all the other rankings. See [1], [3].

1.1 Full Rankings

More formally, let $K(\cdot, \cdot)$ be the *Kendall tau distance* between two full rankings, σ and τ on n items, which is the number of pairwise disagreements (and also happens to be the number of adjacent pairwise inversions needed to get from σ to τ):

$$K(\sigma, \tau) := \sum_{i=1}^{n} \sum_{j=i+1}^{n} \mathbb{1}_{\sigma(i)<\sigma(j)} \mathbb{1}_{\tau(i)>\tau(j)} + \mathbb{1}_{\sigma(i)>\sigma(j)} \mathbb{1}_{\tau(i)<\tau(j)},$$

where $\mathbb{1}_A$ is 1 if A is true and 0 otherwise.

Given a list of m rankings, $\{\sigma_i\}_{i=1}^m$, on all n items, a (perhaps non-unique) *Kemeny optimal* ranking minimizes the following over rankings x:

$$\hat{K}(x, \{\sigma_i\}_{i=1}^m) := \sum_{i=1}^{m} K(x, \sigma_i). \tag{1}$$

In order to solve this minimization problem, we rewrite it as an integer linear program (ILP) to take advantage of ILP solvers. As a warm-up, note that every ranking σ on n items corresponds to a $n \times n$ $(0,1)$-matrix Σ where $\Sigma_{i,j} = 1$ means that item i is ranked higher than item j.

Similarly, consider any $n \times n$ $(0,1)$-matrix x such that $x_{ij} + x_{ji} = 1$ for $i, j \in [n]$ and $i \neq j$ and $x_{ij} + x_{jk} + x_{ki} \geq 1$ for all $i, j, k \in [n]$ and $i \neq j \neq k$ where $[n] := \{1, \ldots, n\}$. We take $x_{ij} = 1$ to mean that item i beats item j and similarly, $x_{ij} = 0$ means that item j beats item i.

The first type of constraints say that there are no ties and every item must be ranked, so exactly one of the following must happen: item i is ranked higher than item j or item j is ranked higher than item i. The second type of constraints ensures transitivity since if $x_{ij}, x_{jk}, x_{ki} = 0$, then we would have item i ranked lower than item j and item j ranked lower than item k but item k ranked lower than item i, a contradiction of transitivity (recall, there are no ties).

It is not difficult to see (with induction on n) that such an x corresponds to a full ranking since the constraints guarantee exactly one item will beat $n-1$ items (which we will take as the item that is ranked best), one item will beat $n-2$ items (which we will take as ranked second best), and so on.

For example, the ranking item 1 > item 2 > item 3 corresponds to $\begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$.

Furthermore, notice that for a $n \times n$ $(0,1)$-matrix x subject to the above constraints, we can rewrite Equation 1 as

$$\begin{aligned}
\hat{K}(x, \{\sigma_i\}_{i=1}^m) &= \sum_{k=1}^{m} K(x, \sigma_i) \\
&= \sum_{k=1}^{m} \sum_{i,j=1}^{n} x_{ji} \mathbb{1}_{\sigma_k(i) > \sigma_k(j)} + x_{ij} \mathbb{1}_{\sigma_k(i) < \sigma_k(j)} \\
&= \sum_{i,j=1}^{n} x_{ji} \left(\sum_{k=1}^{m} \mathbb{1}_{\sigma_k(i) > \sigma_k(j)} \right) + x_{ij} \left(\sum_{k=1}^{m} \mathbb{1}_{\sigma_k(i) < \sigma_k(j)} \right) \\
&= \sum_{i,j=1}^{n} x_{ji} N_{ij} + x_{ij} N_{ji}, \qquad (2)
\end{aligned}$$

where for $i, j \in [n]$, $N_{ij} = |\{k \in [m] : \sigma_k(i) > \sigma_k(j)\}|$ is the number of times item i beat item j over all m rankings, so we can take $N_{ii} = 0$.

Then finding a ranking that minimizes Equation 1 is equivalent to solving the following ILP:

$$
\begin{array}{lrl}
\underset{x \in \{0,1\}^{n \times n}}{\text{minimize}} & \sum_{i,j=1}^{m} & x_{ji}N_{ij} + x_{ij}N_{ji} \\
\text{subject to} & x_{ij} + x_{ji} = 1, & \forall i,j \in [n]\ i \neq j, \\
& x_{ij} + x_{jk} + x_{ki} \geq 1, & \forall i,j,k \in [n], i \neq j \neq k, \\
& x_i \in \{0,1\}, & \forall i \in [n].
\end{array}
$$

1.2 Partial Rankings

Given a partial ranking σ, recall I_σ gives the subset of items that σ is a ranking of. In the partial ranking case, we modify the Kendall tau distance from above so that only pairs of items that have been compared in both rankings contribute:

$$K_{\text{partial}}(\sigma,\tau) := \sum_{i,j \in I_\sigma \cap I_\tau}^{n} \mathbb{1}_{\sigma(i)<\sigma(j)}\mathbb{1}_{\tau(i)>\tau(j)} + \mathbb{1}_{\sigma(i)>\sigma(j)}\mathbb{1}_{\tau(i)<\tau(j)}.$$

(Note that K_{partial} is no longer a metric since $K_{\text{partial}}(\sigma,\tau) = 0$ does not imply $\tau = \sigma$.)

We then solve the corresponding integer linear program from the full ranking case by substituting K for K_{partial}.

2 Borda Count

In short, Borda count methods assign points to items depending on what rank they have in each ranking and ranks the items by their final points. See [5].

Given m (full or partial) rankings, $\{\sigma_i\}_{i=1}^m$, we give $n-(i-1)$ points to item j every time item j is ranked ith in any of the m rankings. We then rank the items according to their accumulated points, where more points indicates a better ranking.

Notice that in the case of full rankings, this scheme of assigning points is equivalent to giving item i $W_i = |\{(\sigma_k, j) : \sigma_k(i) > \sigma_k(j), k \in [m]\}|$ points where W_i is the number of times i wins a pairwise comparison in any ranking.

3 Bradley-Terry-Luce Model

In short, the Bradley-Terry-Luce (BTL) model is a parametric model on the probability that item i beats item j. The items are then ranked by the probability that an item beats all the others. See [2], [4].

Let P be the $n \times n$ matrix such that P_{ij} is the probability that item i beats item j. (P is known as the *preference matrix*.) In the BTL model, the assumption is there exists $w \in \mathbb{R}^n$ such that

$$P_{i,j} = \frac{e^{w_i}}{e^{w_i} + e^{w_j}} = \frac{1}{1 + e^{w_j - w_i}}$$

.

We can use logistic regression (without an intercept) to find the maximum likelihood estimate, $\hat{w}$. In particular, if item i beats item j for some ranking σ_k, then consider $x_{i,j,\sigma_k} \in \mathbb{R}^n$ such that there is a 1 in the coordinate corresponding to item i, a -1 in the coordinate corresponding to item j, and zeros elsewhere. Then in the case of full or partial rankings, we can train a logistic regression model with x_{i,j,σ_k} with label 1 if $i < j$ (as numbers) or label 0 $j < i$. (We could have also put label 1 if $j < i$ and label 0 if $i < j$. The choice is arbitrary but we must be consistent).

After obtaining the maximum likelihood estimate $\hat{w}$ of w, we rank the items by sorting $\hat{w}$.

4 Poster Competition Results

The first day of the poster competition was used to narrow down the 40 student posters to a select group of finalists. Judging the first day was open to anyone who wanted to participate, and we asked participants to provide a ranking on as many posters as they wished. We received 39 ballots of partial rankings. The average number of posters ranked on each ballot was 5 posters and the median was 4 posters.

Originally, we wanted to use the partial ranking version of all three rank aggregation methods on the first day as described above. However, the partial ranking data had two issues: (1) the number of posters ranked from person to person varied greatly (the largest ranking had size 14 whereas the smallest had size 2) and (2) missing data since each person ranked a small number of posters relative to the total number of posters. Related to the issue of missing data, see Figure 2 which depicts the overlap (or lack thereof) between the partial rankings.

Because of these issues and since Kemeny rank aggregation and the BTL model utilize all pairwise comparisons from the ranking data, these algorithms seemed to favor the posters that were ranked highest by the people who ranked a large number of posters. In particular, a poster ranked as first on a list of length k corresponds to $k - 1$ pairwise comparisons in which that poster won.

Therefore, we decided not to use the results of Kemeny and BTL and instead only use the Borda count algorithm. See Figure 1 for a histogram of number of partial rankings a poster appeared in. The posters ranked higher by the Borda count algorithm also tend to appear in a higher number of ballots (since Borda gives points every time a poster was ranked), which seems like a reasonable way to rank the posters in light of the two issues of varying ballot lengths and missing data.

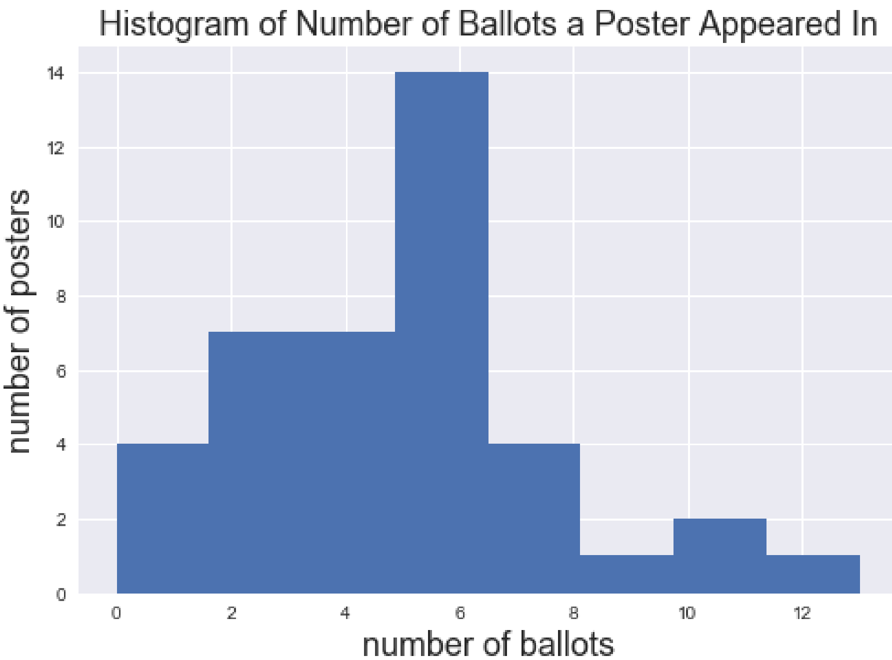

Figure 1: Histogram of Number of Times a Poster Appeared on a Ballot

The second day of the poster competition was limited to 8 selected judges and 5 posters. Each judge gave a ranking on all 5 posters, so that there was no missing data and so that the length of the rankings were uniform.

All three rank aggregation algorithms agreed on the top three posters. However, there were six Kemeny optimal rankings where each Kemeny optimal ranking corresponded to one of the six permutations of the three aforementioned posters. We declared these top three as winners since each algorithm output a different winner.

References

[1] Alnur Alia and Marina Meilab. Experiments with kemeny ranking: What works when? *Mathematical Social Sciences*, 64(1):28–40, 2004.

[2] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[3] Cynthia Dwork, Ravi Kumary, Moni Naorz, and D. Sivakumarx. Rank aggregation revisited. 2002.

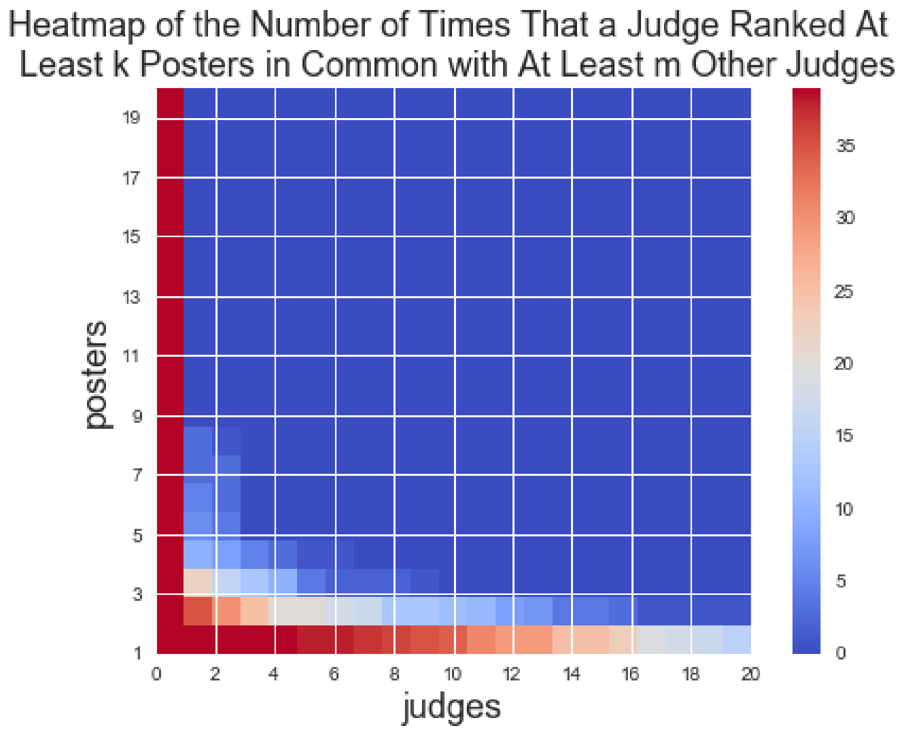

Figure 2: Heatmap of Number of Times That a Judge Ranked At Least k Posters in Common with at least m Other Judges

[4] R.D. Luce. *Individual Choice Behavior: A Theoretical Analysis.* Wiley, 1959.

[5] Arun Rajkumar and Shivani Agarwal. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages I–118–I–126. JMLR.org, 2014.