The Swiss Tournament Model
CIS 700-04: Machine Learning and Econometrics *

Chris Hua *Wharton School, University of Pennsylvania*

Contents

The Swiss tournament structure is well-known and commonly used, particularly in chess, because of its simple heuristic of preferring matchups with teams of similar win count. We consider the desirability of this tournament structure in different contexts, using simulation and empirical results. We find that while the Swiss structure performs well under partial ranking criterions, it underperforms a simple random pairing in full-ranking measures.

Keywords: pairwise comparisons, tournaments, ranking, matching

---

*Contact: chua@wharton.upenn.edu

Introduction

An important goal of tournaments is to find an overall winner, but it is often important to find the top-k contestants as well. A simple example of where this is useful is a preliminary tournament used for seeding purposes, prior to an elimination tournament. In some cases, knowing an exact ranking within this top subgroup is important, such as when a tournament will pay out monetary rewards based on finishing place; in other cases, knowing the exact ranking is not as important.

We present the case of American high school policy debate, in which teams compete in "regular-season" tournaments throughout the year in order to win 'bids' to the Tournament of Championships, the de facto culminating championship. Each round has two teams of two debaters, one "affirmative" (aff) and one "negative" (neg), and a judge. The affirmative side argues a policy-based plan which affirms that year's debate resolution, and the negative argues against the affirmative. For example, the 2012-13 resolution was "The United States federal government should substantially increase its transportation infrastructure investment in the United States."

All tournaments are structured in two parts, with a preliminary Swiss-system tournament and then a knockout/single-elimination tournament. Within the preliminary tournament, the first two rounds are randomly paired, and subsequent rounds are power-matched, which means teams are paired with teams that have similar records (i.e. similar number of wins). These are subject to the constraints that teams cannot debate teams from the same school, and they cannot debate teams who they have been paired with in earlier rounds.

The ultimate goal of "regular season" tournaments is to earn a bid to the "championship" tournament, the Tournament of Champions. These are allocated to tournaments roughly on the basis of tournament size and strength; the effect is that tournaments with more bids attract stronger teams. The bids are set up so that teams who make it to a given round of the tournament get the bid, e.g. octafinals means 16 bids, semifinals is 4 bids, etc. A perverse result of this bid system is that rounds after the bid round, containing the best teams, are treated as unimportant - teams routinely run less serious arguments or simply forfeit rounds- but the bid round and rounds before have enormous strategic investment.

This setup leads us to consider the efficacy of the Swiss-tournament design, in this instance.

Problem Statement

Does the Swiss-style tournament structure effectively find top-k teams?

Literature Review

Previous research has been done into creating tournament structures, and considering the various desirable aspects to optimize for.

Part of the simulation process involves finding pairings, which are directly analogous to a matching, which is a set of n disjoint pairs of participants.

In particular, the Swiss tournament structure is considered in Ólafsson (1990). Ólafsson considers the Swiss tournament structure and various chess-specific considerations; however, he focuses on an algorithm to create pairings which fulfill chess's requirements. He presents a

method using maximum weight perfect matching to perform the pairing matching. Under this method, we employ a graph structure to represent the teams (nodes) and the possible pairings (edges). The graph is initialized as a complete graph with equal weights; that is, all possible pairings are equally desirable, which fits the structure of a random initial pairing. The graph is complete because any team *could* play each other, and we represent desirability via edge weights. At the conclusion of each round, the edges are reweighted to fit the desirability of the pairing. These weights are functions of various competitive factors, including the difference in wins, how many rounds they have played on white/black, whether the pairing has already occured, and others. The general idea is that a higher weight represents a higher preference. Then, the maximum weight perfect matching algorithm finds a matching among the possible pairs.

The weighted perfect matching algorithm is a well-studied problem in computer science and graph theory. The first polynomial time algorithm for the problem was found by Edmonds (1965), known as Edmond's blossom algorithm, and improved on by numerous others, including Cook and Rohe (1999) and most recently by Kolmogorov (2009). The implementation used in this paper follows most directly the process given by Galil (1986). This implementation runs with time complexity $O(nm \log n)$, where n is the number of nodes and m is the number of edges in the graph. Exact details on the method can be found in any of these papers, or from examining the source code of the open-source programs used for simulation.

In a similar vein, Kujansuu, Lindberg, and Erkki (1999) present a method for pairing players, as an extension of the stable roommates problem. The canonical reference for the stable roommates problem is McVitie and Wilson (1971). In the stable roommates problem, each "roommate" creates a preference ranking of the others (full ranking is assumed). Then, the matching is stable if there are no potential roommates i and j who prefer each other to their matched roommate. In the tournament context, after each round, each team has a preference list constructed for them of the teams available to play. The weights can be assigned in a similar manner to Olafsson and have a relatively analogous meaning, representing how preferable a possible pairing between two teams is.

To my knowledge, very few other studies have considered the effectiveness of the Swiss tournament structure.

Research by Glickman and Jensen (2005) has considered alternative tournament formulations from a more theoretical basis. Specifically, Glickman and Jensen present a tournament structure where rounds are matched by maximizing expected Kullback-Leibler distance, a measure of difference between distributions. The pairings are picked such that they maximize the expected Kullback-Leibler distance between the prior and posterior distributions of θ, the distribution of player strengths. This model is heavily influenced by Bayesian optimal design. Notably, Swiss tournaments out-perform their model for small numbers of rounds.

Hanes (2015) researched the effect of power matching in policy debate tournaments, comparing the outcomes from the win rankings with the speaker points assigned to teams. He finds a disparity between the two rankings, and argues that we should prefer the results given by speaker point rankings instead, or at minimum a combination of wins and speaker points. It is worth noting that he considers the implications across a full season, while we focus on the effects on a tournament level.

Solution Approach/Main Results

Tournaments are repeated sets of paired comparisons, and we employ what is known as the Bradley-Terry model to understand the comparisons (Bradley and Terry 1952). The Bradley-Terry model belongs to a family of models known as linear paired comparison models, where win probabilities are only affected by player strengths in terms of the delta between the pairs. For several reasons, it is one of the most, if not the most, popular models for analyzing pairwise comparisons. It is given by:

$$\Pr(Y_{i,j} = 1) = \frac{\theta_i}{\theta_i + \theta_j}$$

Here, $Y_{i,j}$ is an indicator for the outcome of the pairwise comparison between competitors i and j, and θ_i and θ_j represent the underlying strength of competitors i and j. These θ values are relatively unconstrained, though under the traditional B-T assumptions they are positive numbers.

For simplicity, in our simulations, we drew the team strengths from probability distributions. The core distribution we use is the beta distribution. The PDF of the beta distribution is

$$\frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}$$

where: $\mathrm{B}(\alpha,\beta) = \dfrac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$

See the appendix for other distributions which our code and model supports.

Tournament design

As described above, teams compete in 6 or 7 rounds, with the first two rounds randomly paired and the following rounds power-matched. We implement this procedure using an adaption of the maximum weight perfect matching technique (Ólafsson 1990).

Our process is as follows:

1. Team strengths are generated according to the given distribution, parameters, and random seed. Teams are represented as nodes in a symmetric directed graph, and edges are possible pairings.
2. A first round is paired randomly.
3. Results for the round are simulated and recorded, following the Bradley-Terry model for pairwise comparisons.
4. All rounds after the second are paired using maximum weight perfect matching.
5. After the second round, we reweight the graph.

The maximum weight perfect matching procedure is an ingenious method to guarantee good pairings. We have several desirable characteristics in pairings: first, that teams which play each other should not meet in further rounds, and that teams should prefer teams which have the same win total, but if necessary, play teams with a difference of 1 win. We can represent these characteristics within our graph model of a tournament by assigning weights to edges which reflect the desirability of the pairing. Our exact formula for weighting a possible pairing between teams i and j is as follows:

$$W_{i,j} = \alpha - (\beta * |s_i - s_j|)^2$$

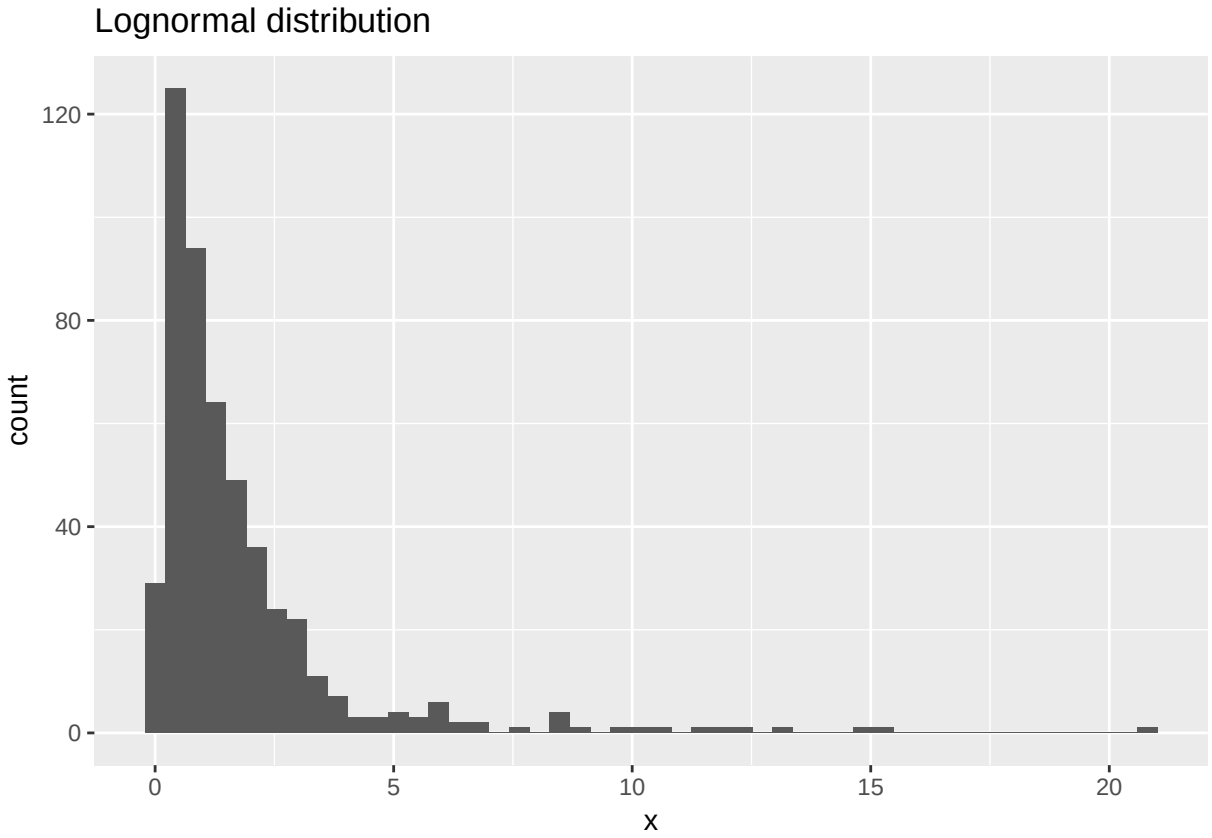
Here, α and β are constants which can be thought of as a location and scale parameter, respectively. We also present a delta value, $|s_i - s_j|$, which is the absolute value of the the difference between the two teams' wins. To make computation easier, we avoid negative weights by first checking the win delta and setting the pairing to a weight of 1 if the difference is greater than 1 win. When a particular pairing is done, we assign the pairing a weight of 0. This method lends itself to a maximum weight method because the larger a weight is on a particular pairing the more desirable it is in a pairing.

Weights are rebalanced at the end of each round, i.e. when all pairings are simulated. All edges that have not been picked are rebalanced, since even if a pairing is undesirable after k rounds, it could be desirable for the $k+1$ round. Picked edges are assigned fixed weights of 0 so that they are not picked. We then develop a pairing for the next round, which is represented as a maximum weight perfect matching. We use Edmond's blossom algorithm, as implemented in **Python** by NetworkX (Hagberg, Schult, and Swart 2008).

Although the algorithm which we use runs in $O(nm \log n)$ time, since our graph is fully connected, we have $m = n(n-1)/2$, which means that the algorithm runs in $O(n^3)$ time, where n is the number of teams competing. This becomes computationally intensive for relatively large tournaments; in our computations, simulating 500 occurences of a 256 team tournament takes over half an hour using a 2013 Macbook Pro.

Note that the algorithm is used to find pairings for round 2, since the round is intended to be randomly paired. At this point the graph is initialized with equal weights for every pairing except those which have occured, which have a 0 weighting. Then, since we have no other constraints, the maximum weight perfect matching returns an acceptable pairing which conveniently guarantees no repeat matches.

We drew our model from the lognormal distribution, with a mean $\mu = 0$, and a standard deviation $\sigma = 1$. We implemented support for other distributions, but empirical results showed little difference in the results, so long as the shape was generally similar. We infer a distribution of skill that is roughly displayed below.



Simulation procedure

We consider several different tournament configurations, and run 500 simulated tournaments for each of them.

Size	Teams	Rounds	K
Small	32	5	8
Medium	64	6	16
Large	128	6	32
Very large	256	7	64

These tournaments are, in order, modeled after a local tournament, the NDCA tournament, the Blake tournament, and the Berkeley tournament. There are infinite numbers of setups to test, but using these offers a realistic test of real world conditions. Under these specifications, we observed the following results.

We present several metrics of success, described below.

- "Top-1" indicates if the top-rated player went undefeated throughout the tournament. This is known as the Copeland winning condition which is any player with a maximum score, (Saari and Merlin 1996).
- Each of these tournaments has a particular K associated with them. These hark back to the goal of finding the K teams who will earn a bid for the tournament; here, we show the percent of teams in the top-K by strength who also place that highly by win rank. This can also be thought of as a partial ranking measure, because we test for group membership of the top-k teams but not the actual placement among those teams.

- Squared loss is defined here as $L = \sum_i (R_i^{\text{Strength}} - R_i^{\text{Wins}})^2$\$. We use R to denote the team's percent rank in terms of their underlying strengths and in terms of their observed wins.

- Finally, we report the Kendall τ and Spearman ρ measures of correlation. These correlation coefficients measure the discrepancy between our reported (win) ranking and the actual underlying (strength) ranking.
  - Kendall's tau-b is given by $\tau_B = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)(n_0 - n_2)}}$ (Kendall 1945). We use the tau-b implementation as it is robust to ties, which happen quite often in this dataset, with a discrete number of rounds played.
  - Spearman's rho is given by $r_s = \rho_{\text{rg}_X, \text{rg}_Y} = \frac{\text{cov}(\text{rg}_X, \text{rg}_Y)}{\sigma_{\text{rg}_X} \sigma_{\text{rg}_Y}}$ (Zwillinger and Kokoska 2001).

We do not report the p-values of the Kendall or Spearman coefficients because these values are all 0.001 or lower, and very highly significant.

Experimental results

For our main Swiss-style tournament simulation, we observed the following results:

Size	Top-1	Top-K	Squared Loss	Kendall's tau	Spearman's rho
Small	0.326	0.926	1.767	0.512	0.661
Medium	0.646	0.928	3.328	0.519	0.684
Large	0.696	0.982	7.432	0.497	0.647
Extra Large	0.660	0.915	17.759	0.437	0.579
Perfect	1.000	1.000	0.000	1.000	1.000

Under a random pairing framework, we observed the following results:

Size	Top-1	Top-K	Squared Loss	Kendall's tau	Spearman's rho
Small	0.088	0.621	2.532	0.378	0.513
Medium	0.512	0.827	5.128	0.384	0.511
Large	0.640	0.978	7.077	0.510	0.664
Extra Large	0.790	0.754	15.251	0.484	0.639
Perfect	1.000	1.000	0.000	1.000	1.000

Each configuration that we test under the Swiss system performs relatively similarly, according to the metrics which we used. This is with the exception of the small tournament, where the top-ranked team finishes undefeated (and thus a Copeland winner) only about a third of time, versus 2/3 of the time in the other configurations. The measures of correlation, given by the Kendall τ and Spearman ρ seem to decrease with larger tournament sizes, but this is likely due to the larger number of teams to rank. This probably also explains the increase in squared loss, an admittedly invented metric. However, all of the Swiss tournament structures perform well under the top-k metric, with over 90% of top-k teams properly identified.

Comparing the results from the Swiss tournament to a tournament where the rounds are randomly paired, with the only constraint being that teams cannot repeat pairings, yields some surprising results.

The larger a tournament is, with random pairings, the likelier that the top team is undefeated. This is reasonable because the top team should get easier placements in the random framework than in the power-matched framework, which is the goal of the Swiss-style tournament. While this finding has little effect on the debate tournaments, which place relatively little emphasis on preliminary winners, it is important for other contexts in which Swiss tournaments are used to pick winners, such as chess.

With the random pairings framework, for the large and extra large tournaments, the three ranking measures defined over all of the teams actually show better results than for the Swiss-style pairings. This is a pretty surprising result, especially because the Swiss-style tournament performs noticeably better than the random pairing tournament in picking the top-k teams. This can be thought of as a discrepancy between partial ranking and full-ranking measures. Understanding why this discrepancy exists would be a worthwhile research direction.

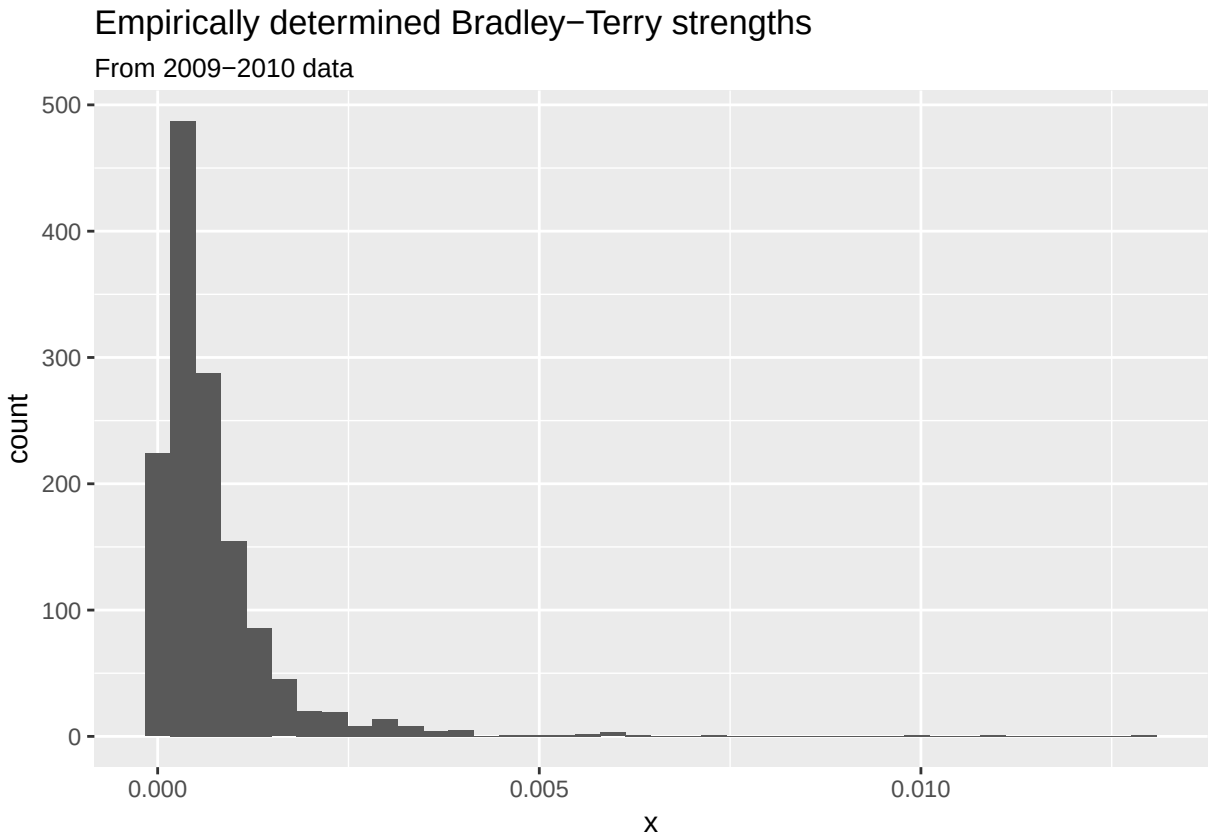
Another interesting trend that we see in the full ranking correlation metrics is that the random pairing tournament performs generally better with larger tournaments, while the Swiss tournament performs generally better with smaller tournaments. This same characteristic was noted by Glickman and Jensen (2005), where their tournament model underperformed Swiss tournaments in sum of squared deviations in ranks (SSDR) for tournaments of 4 and 8 rounds, but was better in 16 round specifications. One possible explanation is that the random pairing rounds employed by Swiss tournaments in the first 2 rounds actually contribute the most to the full rankings, and the power-matched rounds reduce the accuracy. Further research could test the effect of varying the number of random rounds used in the Swiss tournament, to balance the goals of picking a top-k group of teams as well as yielding fair rankings to all participants.

Validation

In addition to individually scraped tournaments, we also have 2 year-long datasets, covering multiple tournaments in the 2009-2010 and 2010-2011 seasons. Using these datasets and their final results, we can estimate Bradley-Terry parameters for the teams participating in those tournaments. We can then rerun our experiments using these empirically determined parameters instead. The 2009-2010 dataset consists of 13310 debated rounds by 1424 teams, in 67 tournaments.

Our MLE estimation is done using the **R** language (R Core Team 2016). In particular, we use the **BradleyTerryScalable** package (Kaye and Firth 2017). This package follows the procedure laid out in (Caron and Doucet 2010) for maximum likelihood estimation of Bradley-Terry parameters when Ford's assumption does not hold. Ford's assumption is: in every possible partition of players into two non-empty subsets, some individual in the second set beats some individual in the first set at least once (Ford 1957). Our datasets are very sparse and cover a wide range of teams, meaning that Ford's assumption does not hold; in particular, this means that the more traditional MLE estimation methods of minorization-maximization (Hunter 2004) and Iterative-Luce Spectral Ranking (Maystre and Grossglauser 2015) cannot be used.

Due to computational considerations we do not include the results of using these empirically determined Bradley-Terry coefficients; however, we do include here a graph of the coefficients. These are reasonably similar to the lognormal distribution which we assumed, and it is thus reasonable that we use the lognormal when running our simulations.



Discussion and Conclusion

A common argument against Swiss-style tournaments is that they provide poor results for top-teams, because of possible disparities in schedule strength. We find empirical proof that the Swiss-style tournament performs well when picking the top-k teams from a given pool, although the Swiss tournament underperforms a random pairing in creating a full ranking. This implies that tournament organizers should place greater emphasis on their goals for the tournament: seeding an elimination tournament, picking an outright winner, or creating a full ranking. For policy debate, the power-matching of the Swiss tournament creates an effective partial ranking which can be used to seed an elimination tournament. However, for chess and other games, the Swiss tournament underperforms random pairings for the tested configurations.

An example of the practical considerations would be a round-robin tournament. We do not test it here, but a round robin tournament would very likely perform well on all of our metrics in a simulation. This is, however, not feasible in real life, since in a large tournament, we would not expect teams to debate hundreds of rounds, each round taking around 2 hours.

In this paper, we have developed an environmental framework for working with tournaments and understanding the implications of their results. Many extensions are can be further investigated.

A particularly curious question is why the Swiss tournament performs well for small tournaments (both in terms of rounds and teams involved) but does not perform well in larger settings, when compared to a random pairing. Further work could investigate the differences in strengths that are created in the Swiss model vs a random model.

Additionally, work has been done to create alternative tournament pairing methods, such as

in Glickman and Jensen (2005). Given our particular question of finding top-k teams, it would be useful to test these other models. One note is that while the Glickman model performs well in theory, the computational difficulty and explanation difficulty would likely make it difficult to implement in practice.

One final avenue of investigation is considering the variance of the different tournament models. For computational purposes, we only reported the mean of the calculated statistics, but can feasibly also include the variance in these measures for the different types of tournaments.

Appendix

Code

Code and all other resources used in writing this paper can be found at the author's Github.

The software which we used to implement the model and perform simulations is open-source and intended to be extensible. Within our paper, we take advantage of this, by utilizing a common framework for testing different numbers of teams and rounds, as well as creating summary statistics. Furthermore, we implement a random pairing model in the model for comparison testing.

In the paper, we drew our theoretical strengths from a lognormal distribution with a mean $\mu = 0$, and a standard deviation $\sigma = 1$. Our framework includes support for the following distributions:

- Exponential distribution
- Uniform distribution
- Lognormal distribution
- Beta distribution
- Gamma distribution

Each of the above can be specified, along with optional shape parameters in the tournament and simulation keyword arguments. See the author's simulation code for examples on how to make these configurations.

Acknowledgements

Special thanks go to Makena Finger and Kevin Huo for their contributions in reading drafts of this paper. All errors are mine.

Bibliography

Bradley, R, and M Terry. 1952. "Rank analysis of incomplete block designs. I. The method of paired comparisons." *Biometrika* 39: 324–45.

Caron, Francois, and Arnaud Doucet. 2010. "Efficient Bayesian Inference for Generalized Bradley-Terry Models." *Journal of Computational and ...* 21 (2004). Taylor & Francis Group: 1–28. doi:10.1080/10618600.2012.638220.

Cook, Williams, and Andre Rohe. 1999. "Computing Minimum-Weight Perfect Matchings." *INFORMS Journal on Computing* 11 (2): 138–48. doi:10.1287/ijoc.11.2.138.

Edmonds, Jack. 1965. "Paths, trees, and flowers." doi:10.4153/CJM-1965-045-4.

Ford, L. R. 1957. "Solution of a Ranking Problem from Binary Comparisons." *The American Mathematical Monthly* 64 (8): 28. doi:10.2307/2308513.

Galil, Zvi. 1986. "Efficient algorithms for finding maximum matching in graphs." *ACM Computing Surveys* 18 (1). ACM: 23–38. doi:10.1145/6462.6502.

Glickman, Mark E., and Shane T. Jensen. 2005. "Adaptive Paired Comparison Design." *JOURNAL OF STATISTICAL PLANNING AND INFERENCE* 127. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.57.7306.

Hagberg, Aric A, Daniel A Schult, and Pieter J Swart. 2008. "Exploring network structure, dynamics, and function using {NetworkX}." In *Proceedings of the 7th Python in Science Conference (Scipy2008)*, 11–15. Pasadena, CA USA.

Hanes, Russel. 2015. "Study of speaker points and power-matching for 2006-7." http://art-of-logic.blogspot.com/2015/07/study-of-speaker-points-and-power.html.

Hunter, David R. 2004. "MM algorithms for generalized Bradley-Terry models." *Annals of Statistics* 32 (1). Institute of Mathematical Statistics: 384–406. doi:10.1214/aos/1079120141.

Kaye, Ella, and David Firth. 2017. *BradleyTerryScalable: Fits the Bradley-Terry Model to Potentially Large and Sparse Networks of Comparison Data*. https://github.com/EllaKaye/BradleyTerryScalable.

Kendall, M. G. 1945. "The treatment of ties in ranking problems." *Biometrika Trust* 33 (3): 239–51. doi:10.1093/biomet/33.3.239.

Kolmogorov, Vladimir. 2009. "Blossom V: A new implementation of a minimum cost perfect matching algorithm." *Mathematical Programming Computation* 1 (1). Springer-Verlag: 43–67. doi:10.1007/s12532-009-0002-8.

Kujansuu, Eija, Tuukka Lindberg, and M Erkki. 1999. "The Stable Roommates Problem and Chess Tournament Pairings." *Divulgaciones Mathematicas* 7 (1): 19–28. http://emis.ams.org/journals/DM/v71/art3.pdf.

Maystre, Lucas, and Matthias Grossglauser. 2015. "Fast and Accurate Inference of Plackett – Luce Models." *Advances in Neural Information Processing Systems* 28: 1–9. doi:no DOI. URL correct.

McVitie, D. G., and L. B. Wilson. 1971. "The stable marriage problem." *Communications of the ACM* 14 (7). MIT Press: 486–90. doi:10.1145/362619.362632.

Ólafsson, S. 1990. "Weighted matching in chess tournaments." *Journal of the Operational Research Society* 41 (1): 17–24. doi:10.1038/sj/jors/0410103.

R Core Team. 2016. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R

Foundation for Statistical Computing. https://www.r-project.org/.

Saari, Donald G., and Vincent R. Merlin. 1996. "The Copeland method." *Economic Theory* 8 (1). Springer-Verlag: 51–76. doi:10.1007/BF01212012.

Zwillinger, Daniel, and Stephen. Kokoska. 2001. *Standard Probability and Statistics Tables and Formulae*. Vol. 43. 2. Chapman & Hall/CRC. doi:10.1198/tech.2001.s620.