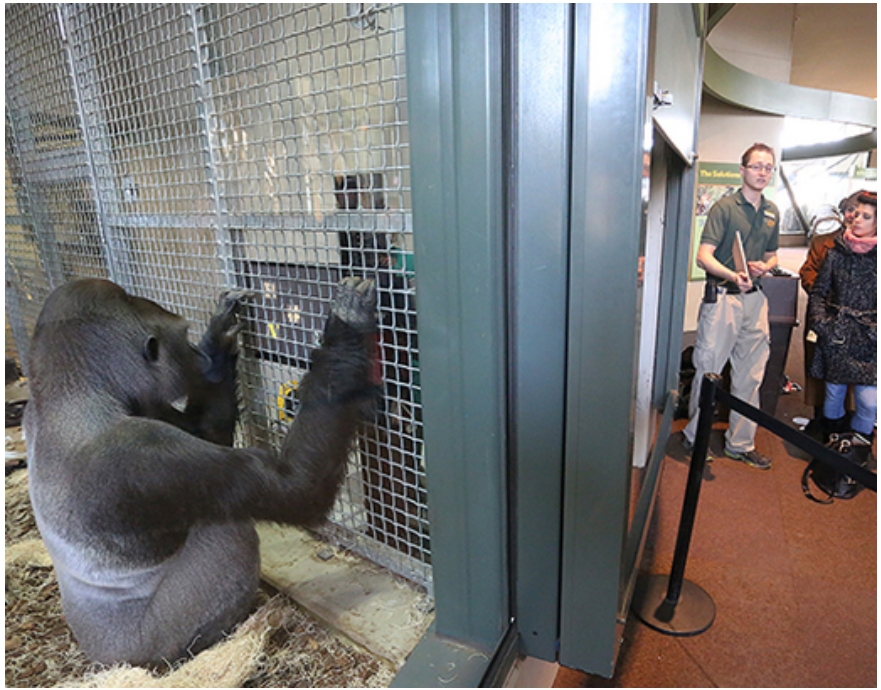
CHALMERS
UNIVERSITY OF TECHNOLOGY

Bayesian Bradley-Terry models: from primates to machine learning

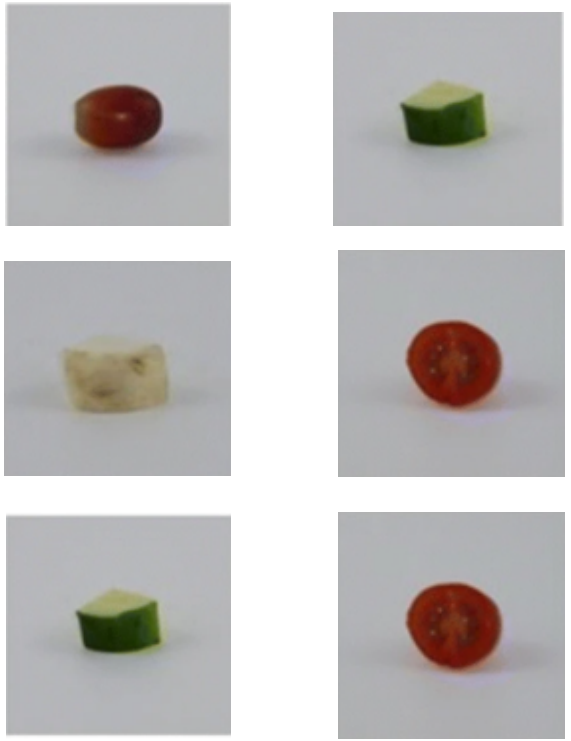
David Issa Mattos

WASP WALLENBERG AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

Gorillas with touchscreens



Lincoln zoo - Chicago



6 food images are presented in pairs

Huskisson, Sarah M., et al. "Using a touchscreen paradigm to evaluate food preferences and response to novel photographic stimuli of food in three primate species (Gorilla gorilla gorilla, Pan troglodytes, and Macaca fuscata)." *International Journal of Primatology* 41.1 (2020): 5-23.

How do we analyze this kind of data?

- We can't do a linear/logistic model
- Frequentist non-parametric models:
  - have many limitations
  - are hard to generalize (e.g. random effects, subject predictors, order effect etc)
  - **Are frequentist**
  - 80% of instructors in statistics in psychology made an incorrect interpretation of the p-value (H. Haller & S. Krauss 2002)
- There is a good chance that SE researchers and practitioners also make p-value interpretation mistakes

image1	image2	y
Grape	Cucumber	0
Grape	Turnip	0
Grape	Tomato	1
Apple	Tomato	0
Apple	Turnip	0
Carrot	Grape	0
Tomato	Turnip	0
Apple	Carrot	0
Carrot	Grape	1
Apple	Turnip	0

How do we analyze paired-comparisons?

- If only two items are presented at a time how can we assess all of them in combination?
- Is there an advantage in using paired comparisons instead of complete blocks?
- What kind of information can we get from this kind of assessment?

The Bradley-Terry model (1952)

$$\mathcal{P}[i \text{ beats } j] = \frac{\alpha_i}{\alpha_i + \alpha_j}$$

- Alpha is a ‘strength’ variable
- This model just computes the probability of selecting i instead of j

or

$$\mathcal{P}[i \text{ beats } j] = \frac{\exp \lambda_i}{\exp \lambda_i + \exp \lambda_j}$$

- We just use a log transformation so we can compute lambda from [- infinity. + infinity]

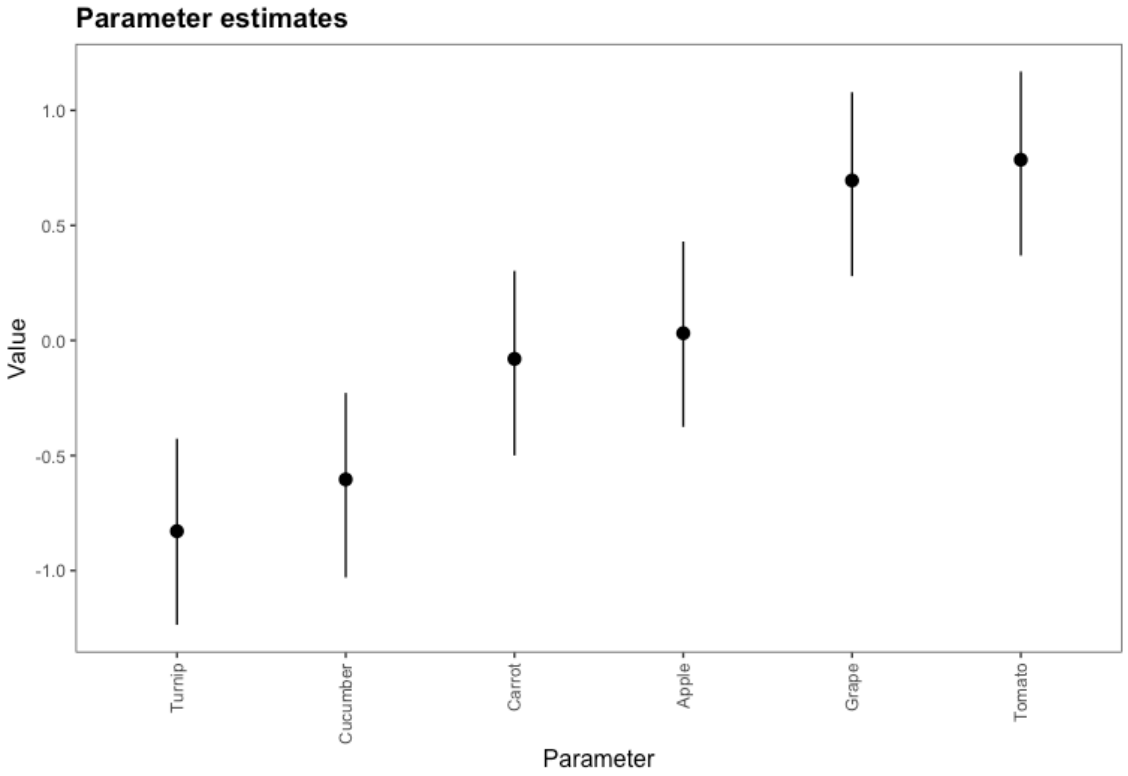
Mattos, I. D. and Ramos, E. M. S. “Bayesian Paired-Comparison with the bpcs package” In submission (https://arxiv.org/pdf/2101.11227.pdf)

Making it Bayesian (1970)

$$\mathcal{P}[i \text{ beats } j] = \frac{\exp \lambda_i}{\exp \lambda_i + \exp \lambda_j}$$

$$y_{i,j} \sim \text{Bernoulli}(\mathcal{P}[i \text{ beats } j])$$

$$\lambda_i \sim \mathcal{N}(0, \sigma_\lambda^2), \text{ [Prior]}$$



These probabilities are a direct (and easy to interpret) measure of effect size

Estimated posterior probabilites

i	j	i_beats_j	j_beats_i
Apple	Carrot	0.54	0.46
Apple	Cucumber	0.68	0.32
Apple	Grape	0.36	0.64
Apple	Tomato	0.39	0.61
Apple	Turnip	0.74	0.26
Carrot	Cucumber	0.64	0.36
Carrot	Grape	0.25	0.75
Carrot	Tomato	0.26	0.74
Carrot	Turnip	0.73	0.27
Cucumber	Grape	0.15	0.85
Cucumber	Tomato	0.17	0.83
Cucumber	Turnip	0.51	0.49
Grape	Tomato	0.58	0.42
Grape	Turnip	0.83	0.17
Tomato	Turnip	0.83	0.17

What does it have to do with SE (or Id)?

Survey design

- Likert scale is not always good (actually it has loads of problems)
  - Bias by anchoring (you answer the next question based on the previous one)
  - Faking response for social acceptability
  - Not suitable for some people/contexts
  - It is common to use statistics wrong to analyze likert scale data
  - See more: Liddell & Kruschke, 2018, Coetzee & Taylor, 1996; Luckett, Burns, & Jenkinson, 2020; Petrou, 2003
- Forced choice assessment
  - You must choose one alternative only (like the gorillas)
  - It can be between two (more common) or several alternatives

- Bradley-Terry models have been used in a wide range of fields of science
  - Health care
  - Survey design/analysis
  - Study with children
  - Ranking web pages in search engine (Bing)
  - Tourism
  - Primates
  - Sports

What does it have to do with SE?

Benchmark experiments

- You want to compare tools/methods tested against a benchmark suite
- Many repetitions of the same tool in the same benchmark test
  - Introduces dependency in the data
- Some benchmarks are not comparable in term of response
  - Use of a ranking system or data transformation
- There can be clusters in the benchmarks
  - Introduces dependency in the data

Benchmark experiments

- If one tool (i) performs better than the other (j) in a benchmark test, we can say that:
  - Tools i beats tool j in benchmark s
- Extending the Bradley-Terry model for random effects
  - We want to compensate for dependency in the data
  - **This is not optional**

$$\mathcal{P}[i \text{ beats } j] = \frac{\exp \lambda_{i,s}}{\exp \lambda_{i,s} + \exp \lambda_{j,s}}$$

$$\lambda_{i,s} = \lambda_i + U_{i,s}$$

$$y_{i,j} \sim \text{Bernoulli}(\mathcal{P}[i \text{ beats } j])$$

$$\lambda_i \sim \mathcal{N}(0, \sigma_\lambda^2), \text{ [Prior]}$$

$$U_{i,s} \sim \mathcal{N}(0, U_{\text{std}}^2), \text{ [Prior]}$$

$$U_{\text{std}} \sim \mathcal{N}(0, \sigma_U^2), \text{ [Prior]}$$

Classification of algorithms for automated labelling

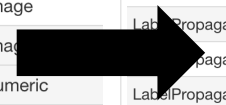
Teodor's research

- There are different semi-supervised learning algorithms (ask Teodor)
  - Label Spreading KNN and RBF
  - Label Propagation KNN and RBF
- Experimental setup with 18 datasets in three categories and 10 repetitions of each algorithm in each dataset (with different seeds)
- How do we rank them in terms of accuracy?

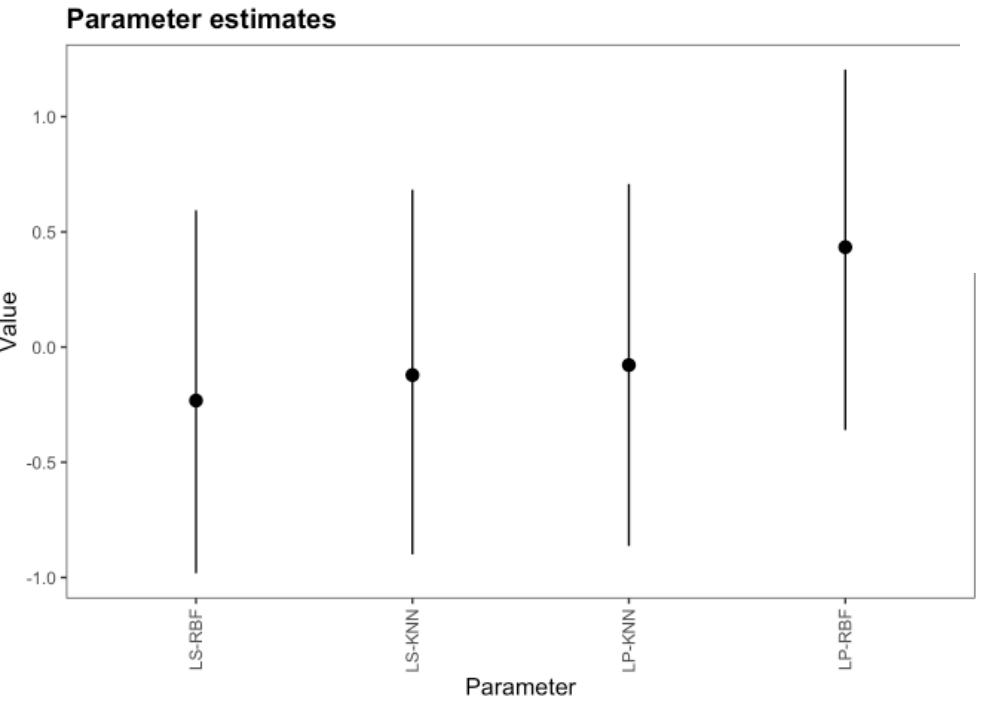
Fredriksson T. Mattos D. I., Bosch J., Olsson. H. H. "An Empirical Evaluation of Algorithms for Data Labeling" In submission 2021

- First, we expand the data into paired comparisons

Value	Iteration	Variable	Model	Dataset	DataType
0.6967609	7	50%	LabelSpreading_rbf	digits	image
0.3333333	2	50%	LabelSpreading_knn	corpus	text
0.1666667	3	50%	LabelPropagation_knn	reuters	text
0.0181818	8	10%	LabelPropagation_rbf	cifar	image
0.0004808	8	50%	LabelPropagation_rbf	cifar2	ima
0.2756910	4	50%	LabelPropagation_knn	german	numeric
0.0181818	4	10%	LabelPropagation_knn	mnist	image
0.1666667	9	10%	LabelPropagation_knn	reuters	text
0.1280959	1	10%	LabelPropagation_rbf	20news	text
0.2635986	0	50%	LabelPropagation_knn	musk1	numeric



model0	model1	y	Iteration	Dataset	DatasetType
LabelPropagation_rbf	LabelSpreading_rbf	1	0	wine	numeric
LabelSpreading_knn	LabelSpreading_rbf	0	8	wine	numeric
LabelPropagation_rbf	LabelSpreading_knn	0	5	corpus	text
LabelPropagation_rbf	LabelSpreading_rbf	0	6	wine	numeric
Lab ropagation_knn	LabelPropagation_rbf	1	3	cifar	image
pagation_rbf	LabelSpreading_rbf	0	0	corpus	text
LabelPropagation_knn	LabelSpreading_rbf	0	7	20news	text
LabelPropagation_knn	LabelSpreading_rbf	0	8	reuters	text
LabelPropagation_rbf	LabelSpreading_knn	0	5	musk1	numeric
LabelSpreading_knn	LabelSpreading_rbf	1	9	cifar	image
LabelPropagation_knn	LabelSpreading_rbf	0	1	20news	text



Estimated posterior probabilites

i	j	i_beats_j	j_beats_i
LabelPropagation_knn	LabelPropagation_rbf	0.33	0.67
LabelPropagation_knn	LabelSpreading_knn	0.45	0.55
LabelPropagation_knn	LabelSpreading_rbf	0.45	0.55
LabelPropagation_rbf	LabelSpreading_knn	0.62	0.38
LabelPropagation_rbf	LabelSpreading_rbf	0.56	0.44
LabelSpreading_knn	LabelSpreading_rbf	0.41	0.59

Estimated posterior ranks

Parameter	MedianRank	MeanRank	StdRank
lambda[LabelPropagation_rbf]	1	1.426	0.756
lambda[LabelPropagation_knn]	3	2.700	0.976
lambda[LabelSpreading_knn]	3	2.812	0.959
lambda[LabelSpreading_rbf]	3	3.062	0.976

What about ties?

Surveys

- If you have a survey with questions such as:
  - Which do you prefer? A, B or doesn't matter
- You need to take ties (doesn't matter) into consideration

Benchmark experiments

- If your benchmarks have ties:
  - Solve then randomly (if very few ties)
  - You model the ties (many ties)

The Davidson model

- Extension of the Bradley-Terry model to handle ties
- Extra parameter nu controls how much ties you have, if they depend on the strength values or if the data have ties independently of the strength

$$\mathcal{P}[i \text{ beats } j | \text{not tie}] = \frac{\exp \lambda_i}{\exp \lambda_i + \exp \lambda_j + \exp\left(\nu + \frac{\lambda_i + \lambda_j}{2}\right)}$$

$$\mathcal{P}[i \text{ ties } j] = \frac{\exp\left(\nu + \frac{\lambda_i + \lambda_j}{2}\right)}{\exp \lambda_i + \exp \lambda_j + \exp\left(\nu + \frac{\lambda_i + \lambda_j}{2}\right)}$$

$$y_i \sim \text{Bernoulli}(\mathcal{P}[i \text{ beats } j | \text{not tie}])$$

$$\text{tie}_{i,j} \sim \text{Bernoulli}(\mathcal{P}[i \text{ ties } j])$$

$$\lambda_i \sim \mathcal{N}(0, \sigma_\lambda^2), \text{ [Prior]}$$

$$\nu \sim \mathcal{N}(0, \sigma_\nu^2), \text{ [Prior]}$$

Modeling the Brazilian football league

- Similar to the other models, we can use the Davidson model to model the ties and obtain a posterior rank estimation

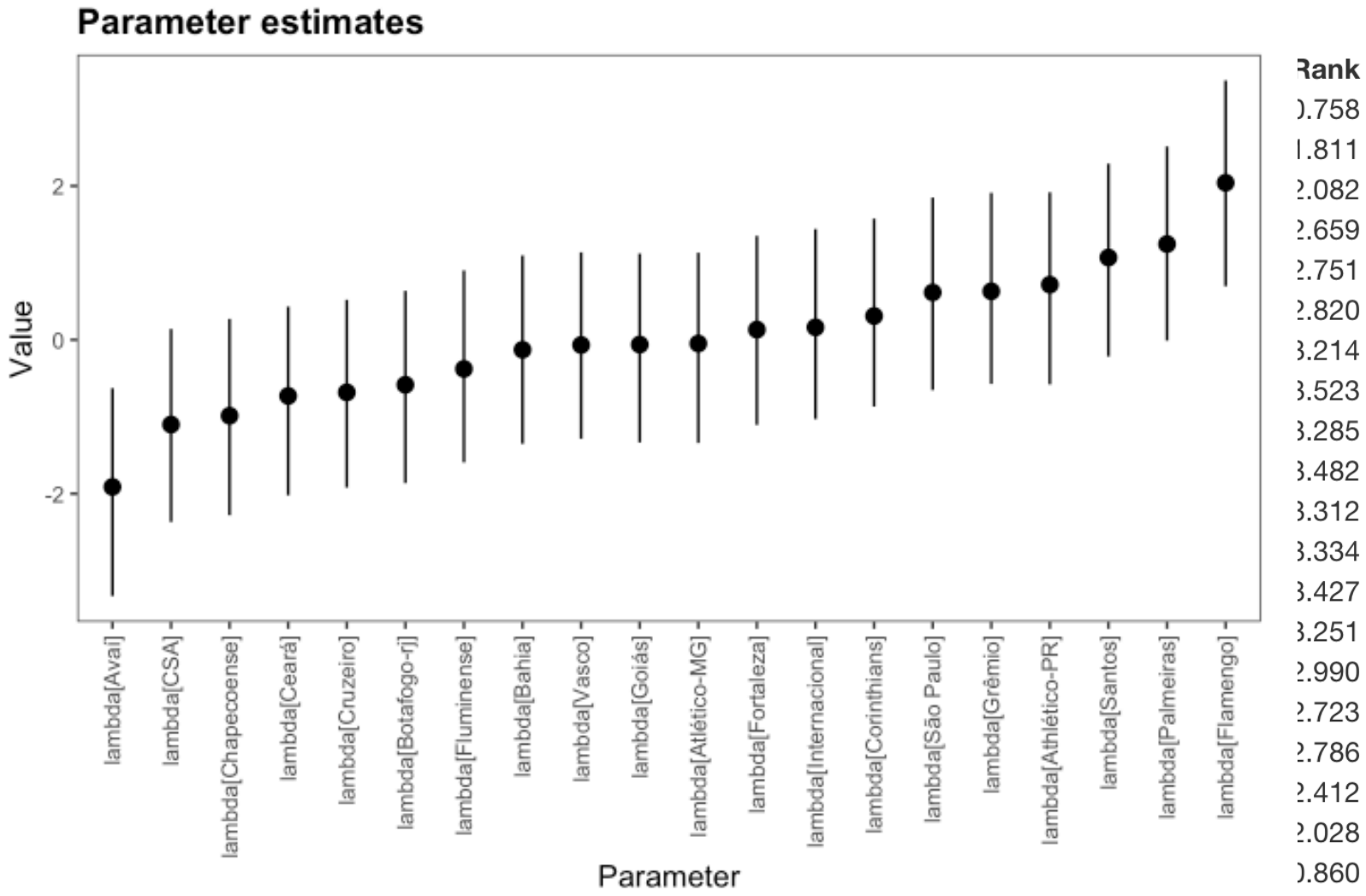
It indicates that ties occur a bit more often than by just the strength of the teams

Parameters estimates

Parameter	Mean	HPD_lower	HPD_higher
lambda[Avaí]	-1.912	-3.331	-0.627
lambda[Internacional]	0.162	-1.031	1.440
lambda[Cruzeiro]	-0.683	-1.921	0.520
lambda[Botafogo-rj]	-0.584	-1.862	0.637
lambda[Vasco]	-0.067	-1.286	1.137
lambda[Corinthians]	0.310	-0.867	1.576
lambda[CSA]	-1.101	-2.368	0.143
lambda[Goiás]	-0.061	-1.335	1.122
lambda[Fortaleza]	0.133	-1.107	1.352
lambda[Santos]	1.070	-0.217	2.291
lambda[Grêmio]	0.631	-0.573	1.912
lambda[Palmeiras]	1.245	-0.011	2.514
lambda[Atlético-MG]	-0.048	-1.340	1.133
lambda[Chapecoense]	-0.986	-2.280	0.272
lambda[Ceará]	-0.729	-2.022	0.433
lambda[Athlético-PR]	0.717	-0.580	1.919
lambda[Flamengo]	2.042	0.696	3.372
lambda[São Paulo]	0.616	-0.654	1.849
lambda[Bahia]	-0.129	-1.353	1.098
lambda[Fluminense]	-0.378	-1.594	0.903
nu	0.385	0.159	0.624

Not
indi
ther
as v
not

The
suc
adv
you



- Note that it does not necessarily indicates who won the league since there are other rules involved (such as win=3pts and tie=1pt) there are not considered in the model.
- Bradley-Terry models are competitive with (more famous ML) models in predicting probability of winning in sports tournoments
- There are other possible extensions that can be combined
  - such as order effects (home field advantage), i.e. the order in which you present the items matter
  - Subject predictors (e.g. characteristics of the subjects that rate the items)
  - Generalized models (e.g. specific characteristics of the items that make them beat more another item)
  - Time effect for panel data

How to do this (very cool) type of analysis?

- Download the bpcs package (https://github.com/davidissamattos/bpcs) or in CRAN
- Simple interface
- Uses Stan as the HMC
- Inspect the posterior distribution and measure effect size (the actual probabilities)
- Publication-ready functions (export tables to Latex and HTML) and generate figures

```
m1_autol <-
  bpc(
    d_acc_bt,
    player0 = 'model0',
    player1 = 'model1',
    result_column = 'y',
    model_type = 'bt-U',
    cluster = c("Dataset"),
    priors = list(prior_lambda_std = 0.5, prior_U1_std=1.0),
    iter = 3000
  )
```

```
get_probabilities_table(m1_gor, format='html')
```

```
get_parameters_table(m_football, format = 'html')
```

```
plot(m1_gor, rotate_x_labels = T)
```

Some links

- Reproducible code and presentation
  - https://davidissamattos.github.io/SESeminarBT2021/
  - https://github.com/davidissamattos/SESeminarBT2021
- Bpcs package
  - https://github.com/davidissamattos/bpcs
  - https://davidissamattos.github.io/bpcs/index.html (online documentation)
- Paper describing the models
  - https://arxiv.org/pdf/2101.11227.pdf

CHALMERS
UNIVERSITY OF TECHNOLOGY

Thanks!

davidis@chalmers.se

WASP WALLENBERG AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM