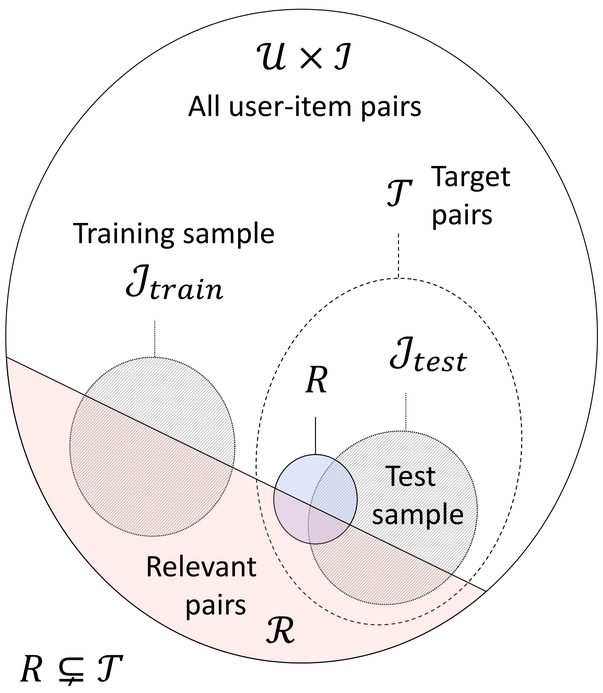
Characterization of Fair Experiments for Recommender System Evaluation – A Formal Analysis

Pablo Castells and Rocío Cañamares
Universidad Autónoma de Madrid
{pablo.castells,rocio.cannamares}@uam.es
IR Group @UAM

Evaluation fairness condition

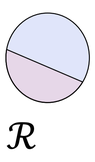
Elements of an experiment



$R \subsetneq \mathcal{T}$
$\mathcal{J}_{test} \subseteq \mathcal{T} \subseteq (\mathcal{U} \times \mathcal{I}) \setminus \mathcal{J}_{train}$
$\mathcal{J}_{train} \cap \mathcal{J}_{test} = \emptyset$

Metric definition

- $P = \dfrac{|R \cap \mathcal{R}|}{|R|} = p(\mathcal{R}|R)$
- $Recall = \dfrac{|R \cap \mathcal{R}|}{|\mathcal{R}|} = p(R|\mathcal{R})$



Metric estimates

- $\hat{P} = \dfrac{|R \cap \mathcal{R} \cap \mathcal{J}_{test}|}{|R|} = \boldsymbol{p(\mathcal{J}_{test}|\mathcal{R}, R)}\, P$
- $\hat{R}ecall = \dfrac{|R \cap \mathcal{R} \cap \mathcal{J}_{test}|}{|R \cap \mathcal{J}_{test}|} = \dfrac{\boldsymbol{p(\mathcal{J}_{test}|\mathcal{R}, R)}}{\boldsymbol{p(\mathcal{J}_{test}|\mathcal{R})}}\, Recall$

Fair estimates

Preservation of system comparisons: $P(R_1) \leq P(R_2) \Leftrightarrow \hat{P}(R_1) \leq \hat{P}(R_2)$ – we say $\hat{P} \propto P$

Metric estimate preserves system comparison $\Leftrightarrow$ $p(\mathcal{J}_{test}|\mathcal{R}, R)$ is the same for all systems

$$\boldsymbol{\hat{P} \propto P \iff p(\mathcal{J}_{test}|\mathcal{R}, R) \sim p(\mathcal{J}_{test}|\mathcal{R}, \mathcal{T}) \;\; \forall R \subset \mathcal{T}}$$

(and same for $\hat{R}ecall \propto Recall$)

Fair experiment $\Leftrightarrow$ test judgments are identically and independently distributed over relevant targets

Empirical fairness test

- Take some sample $\mathcal{J} = \mathcal{J}_{train} \cup \mathcal{J}_{test}$ of user preferences (ratings, judgments, observed interaction)
- **Null hypothesis:** Let user preferences $\mathcal{R}$ be random, i.e. uniformly and independently distributed over items (the sample $\mathcal{J}$ may not)
- Run an experiment over $\mathcal{J}$, $\mathcal{R}$ for a set of recommendation algorithms
- Some system is **better than random** recommendation $\Rightarrow$ **Then your experiment is unfair** (the data sampling/subsampling, the metric, etc.)

Analysis of common experimental protocols

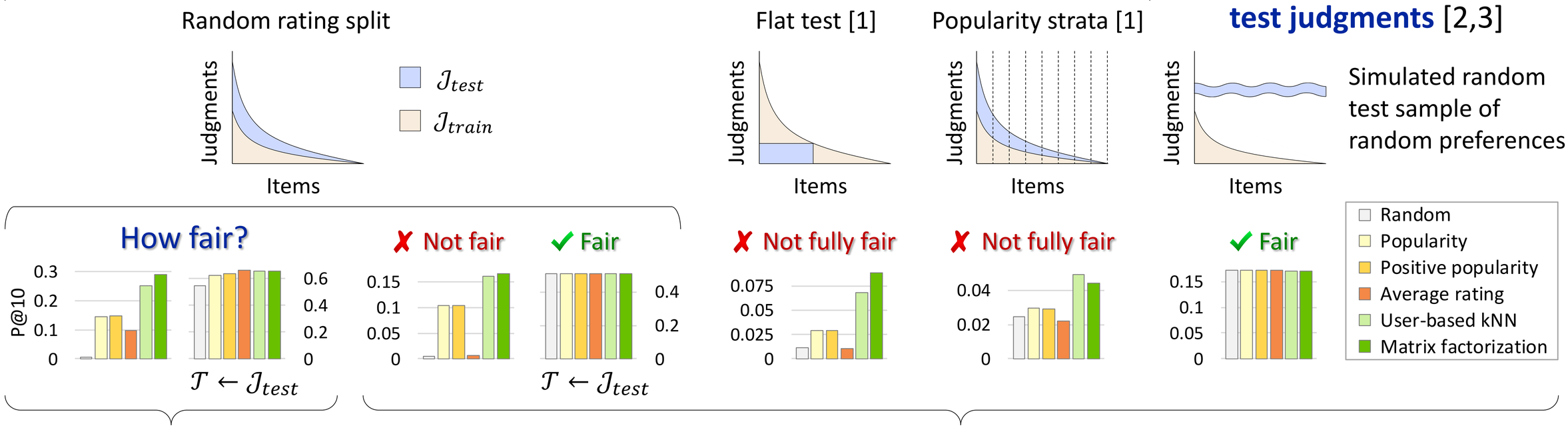
Null hypothesis recreation: taking e.g. MovieLens 1M, keep ratings (judgment set $\mathcal{J}$) but shuffle rating values ($\mathcal{R}$ set) over ratings

→ User preferences become random, uniform and independent between users

→ Judgment distribution retains popularity biases and inter-user (and inter-item) dependencies

1. Free user feedback

2. Randomized (forced) test judgments [2,3]



Original data (MovieLens 1M) **Null hypothesis** (randomized MovieLens 1M preferences)

Conclusions

- Only randomized test judgments or $\mathcal{T} \leftarrow \mathcal{J}_{test}$ ensure fairness
  - But $\mathcal{T} \leftarrow \mathcal{J}_{test}$ is not as realistic as $\mathcal{T} \leftarrow (\mathcal{U} \times \mathcal{I}) \setminus \mathcal{J}_{train}$ (plus coverage shortfalls)
  - Forced judgments to be handled with some care to be fully fair (see in paper)
- Other protocols are biased to non-random patterns in observations
  - Popularity, inter-user dependences, etc. (avg rating would not seem affected though)
- We also examine experimental protocols analytically
  - Empirical fairness test is consistent with analytical fairness condition
- Temporal split can be usually expected to be still biased
- Interleaved AB tests should be fair

1. A. Bellogín, P. Castells and I. Cantador. Statistical Biases in Information Retrieval Metrics for Recommender Systems. Information Retrieval 20(6), July 2017, pp. 606-634.
2. R. Cañamares and P. Castells. Should I Follow the Crowd? A Probabilistic Analysis of the Effectiveness of Popularity in Recommender Systems. SIGIR 2018, Ann Arbor, MI, USA, July 2018, pp. 415-424.
3. B. Marlin and R. Zemel. Collaborative prediction and ranking with nonrandom missing data. RecSys 2009, New York, NY, USA, October 2009, pp. 5-12.

Workshop on offline evaluation for recommender systems (REVEAL 2018) at the 12th ACM Conference on Recommender Systems (RecSys 2018)