Universidad Autónoma de Madrid

Biblos-e Archivo
Repositorio Institucional UAM

Repositorio Institucional de la Universidad Autónoma de Madrid
https://repositorio.uam.es

Esta es la **versión de autor** del artículo publicado en:
This is an **author produced version** of a paper published in:

Deldjoo, Y., Anelli, V.W., Zamani, H. et al. A flexible framework for evaluating user and item fairness in recommender systems. User Model User-Adap Inter 31, 457–511 (2021).

DOI: https://doi.org/10.1007/s11257-020-09285-1

Copyright: © 2021 Springer Nature

El acceso a la versión del editor puede requerir la suscripción del recurso
Access to the published version may require subscription

User Modeling and User-Adapted Interaction manuscript No.
(will be inserted by the editor)

A Flexible Framework for Evaluating User and Item Fairness in Recommender Systems

Yashar Deldjoo · Vito Walter Anelli · Hamed Zamani · Alejandro Bellogín · Tommaso Di Noia

Received: date / Accepted: date

Abstract One common characteristic of research works focused on fairness evaluation (in machine learning) is that they call for some form of parity (equality) either in treatment – meaning they ignore the information about users' memberships in protected classes during training – or in impact – by enforcing proportional beneficial outcomes to users in different protected classes. In the recommender systems community, fairness has been studied with respect to both users' and items' memberships in protected classes defined by some sensitive attributes (e.g., gender or race for users, revenue in a multi-stakeholder setting for items). Again here, the concept has been commonly interpreted as some form of *equality* – i.e., the degree to which the system is meeting the information needs of all its users *in an equal sense*. In this work, we propose a probabilistic framework based on Generalized Cross Entropy (GCE) to measure fairness of a given recommendation model. The framework comes with a suite of advantages: first, it allows the system designer to define and measure fairness for both users and items and can be applied to any classification task; second, it can incorporate various notions of fairness as it does not rely on specific and pre-defined probability distributions and they can be defined at design time; finally, in its design it uses a gain factor, which can be flexibly defined to contemplate different accuracy-related metrics to measure fairness upon decision-support metrics (e.g., precision,

Hamed Zamani is currently affiliated with Microsoft.

Yashar Deldjoo · Vito Walter Anelli · Tommaso Di Noia
Polytechnic University of Bari, Italy
E-mail: {yashar.deldjoo, vitowalter.anelli, tommaso.dinoia}@poliba.it

Hamed Zamani
University of Massachusetts Amherst, United States
E-mail: zamani@cs.umass.edu

Alejandro Bellogín
Universidad Autónoma de Madrid, Spain
E-mail: alejandro.bellogin@uam.es

recall) or rank-based measures (e.g., NDCG, MAP). An experimental evaluation on four real-world datasets show the nuances captured by our proposed metric regarding fairness on different user and item attributes, where nearest-neighbor recommenders tend to obtain good results under equality constraints. We observed that when the users are clustered based on both their interaction with the system and other sensitive attributes, such as age or gender, algorithms with similar performance values get different behaviors with respect to user fairness due to the different way they process data for each user cluster.

1 Introduction

The use of recommender systems (RS) has expanded dramatically over the last decade, mostly due to their enormous business value. According to the statistics revealed by Netflix, 75% of the downloads and rentals come from their recommendation service. This is a clear mark of the strategic importance of such a service in several companies [2,52]. The success of RS is commonly measured by how well they are capable of making accurate predictions, i.e., items that users will likely interact with, purchase, or consume. Hence, the main effort of the research community over the last decade has been devoted to improving the *utility of recommendations* often measured in terms of effectiveness as well as to address beyond-accuracy aspects (e.g., novelty or diversity).

Collaborative filtering (CF) models such as standard SVD[1], SVD++ [57], WRMF [65,49], SLIM [64], NeuralCF [47], and JSR [86] lie at the core of most modern recommender systems (RS) due to their good performance of recommendation accuracy. Besides, a growing number of research works have leveraged different types of contextual information or external knowledge sources, such as knowledge bases/graphs, multimedia, user-generated tags, and, reviews among others, as additional information beyond the user-item interaction matrix to further enhance the final utility of recommendation.

While recommendation models have reached a remarkable level of maturity in terms of effectiveness/performance in many application scenarios, at the same time, concerns have been recently raised on the *fairness* of the recommendation models. As a matter of fact, recommendation algorithms, like other machine learning algorithms, are prone to imperfections due to algorithmic biases or biases in data. As stated by Barocas et al. [17] "*data can imperfect the algorithms in ways that allow these algorithms to inherit the prejudices of prior decision-makers*". Since RS assist users in many decision-making and mission-critical tasks such as medical, financial, or job-related ones [79,75], unfair recommendation could have far-reaching consequences, impacting people's lives and putting minority groups at a major disadvantage.

In the past, the notion of unfair recommendation was often associated with a non-uniform distribution of the benefits among different groups of users and items, as in [41], where the authors studied this issue for users of different demographic groups. Interestingly, many works in the last years have gone beyond this view and,

[1] `https://sifter.org/~simon/journal/20061211.html`

nowadays, fairness, and, analogously, unfairness can be defined as adopting more fine-grained and non-uniform perspectives. As a consequence, measuring fairness is becoming more complex especially if one wants to quantify it.

Furthermore, according to [84, 85], we can classify the most popular notions of unfairness used in the literature as *disparate treatment* and *disparate impact*. Their common characteristic is that they both call for some form of parity (equality), either by ignoring user's membership in protected classes (parity in treatment) or enforcing parity in the fractions of users belonging to different protected classes, receiving beneficial outcomes (parity in impact). Under an operational lens, we may say that parity in treatment refers to the training phase of a model while parity in impact to its usage. Although they look tightly connected, we know that parity in treatment does not necessarily imply a parity in impact.

From a recommender systems perspective, where users are first-class citizens, there are multiple stakeholders, which raise fairness issues that can be studied for more than one group of participants [25]. Previous work on fairness evaluation in RS has mostly interpreted fairness as some form of *equality* across multiple groups (e.g., gender, race). For example, Ekstrand et al. [41] studied whether RS produce *equal utility* for users of different demographic groups. In addition, Yao and Huang [82] studied various types of unfairness that can occur in collaborative filtering models where, to produce fair recommendations, the authors proposed to penalize algorithms producing disparate distributions of prediction error. Nonetheless, although less common, there are a few works where fairness has been defined beyond uniformity [20, 74, 87]. For instance, Biega et al. [20] concentrate on discovering the relation between *relevance* and *attention* in search (information retrieval). During a search session, searchers are subject to a high degree of *positional bias* due to paying much more attention to the top-ranked items than lower-ranked items. As a consequence, despite having a proper ranking based on relevance, lower-ranked items receive disproportionately less attention than they deserve. Their proposed approach promotes the notion that ranked subjects should receive the attention that is proportional to their *worthiness* in a given search scenario and achieve fairness of attention by making exposure proportional to relevance. These research works, however, have focused on fairness from different perspectives and for different purposes.

In the present work, we argue that fairness does not necessarily imply equality between groups, but instead the proper distribution of utility (benefits) based on merits and needs. As an example, within a commercial system, we expect to have a different behavior between premium and free users. In such cases, we should not be surprised by the different resulting utility values for the two classes of users. Starting from this idea, we mainly focus on quantifying unfairness in recommendation systems, and we propose a probabilistic framework based on Generalized Cross Entropy (GCE) to measure fairness (or unfairness) of a given recommendation model that can be applied to diverse recommendation scenarios. This is a general approach that can be easily adapted to any classification task. Our framework allows the designer to define and measure fairness for groups of users (samples in a generic classification task) and for groups of items (target in a classification task). Moreover, the proposed framework is particularly flexible in the definition of different notions of fairness as it does not rely on specific and predefined probability distributions but they can be de-

fined at design time. This lets the designer consider equality- and non-equality-based fairness notions adopting a single and unified perspective. The main characteristics of the proposed framework can be summarized as follows

- A large portion of previous work defines fairness as some form of **equality** across multiple groups (e.g., gender, race) [41]. However, as pointed out by some researchers [44, 82], fairness is not necessarily equivalent to equality. The proposed framework is sufficiently flexible to allow designers to define fairness based on a given arbitrary probabilistic distribution (in which uniform distribution is equivalent to equality in fairness).
- As a general remark, the proposed fairness-evaluation metric comes with a suite of other advantages compared to prior art:
  1. It incorporates a **gain factor** in its design, which can be flexibly defined to contemplate different accuracy-related metrics to measure fairness. Examples of such measures are *recommendation count* (focused on global count of recommendations), *decision-support metrics* (e.g., precision, recall) or *rank-based metrics* (e.g., nDCG, MAP). Prior art usually focuses on one of these aspects), which makes our approach more encompassing and general (cf. Section 2).
     The introduction of the gain factor derives from the assumption that user satisfaction can be defined in many different ways. Based on the specific scenario, a certain metric could be more useful than others, and, as a consequence, the considered gain factor should differ. Additionally, the generalization of the gain factor allows the designer to adopt ranking-based gains like nDCG. This opens up new interesting perspectives. Let us suppose we would like to measure fairness for different groups of items adopting nDCG as a gain factor. If the adopted probability distribution is not equal among groups, the GCE value will be related to the average position of the items of specific groups in the recommendation lists. The GCE will then measure if a recommender system is promoting relevant items from specific groups to users.
  2. Unlike most previous works that solely focused on either user fairness or item fairness, the proposed framework integrates **both** user-related and item-related notions of fairness. Also, we choose to evaluate fairness considering the various item and user attributes (more specifically, price, year, and popularity for items, and happiness, helpfulness, interactions, age, and gender for users), showing how the different RS behave in this respect. This brings our work closer to multiple stakeholder settings where benefits of multiple parties involved in the recommendation process should be considered (refer to Section 2 for more details).
  3. A critical characteristic of a suitable evaluation metric is their interpretability and explainability power. Generalized Cross-Entropy is designed based on theoretical foundations, which makes it easy to understand and interpret.

The main contributions of this paper are developed around the following research questions:

RQ1. How to define a fairness evaluation metric that considers different notions of fairness (not only equality)? We propose a probabilistic framework for evaluating

RS fairness based on attributes of any nature (e.g., sensitive or insensitive) for both items or users. We show that the proposed framework is flexible enough to measure fairness in RS by considering it as equality or non-equality among groups, as specified by the system designer or any other parties involved in a multi-stakeholder setting.

RQ2. How do classical recommendation models behave in terms of such an evaluation metric, especially under non-equality definitions of fairness? Some studies have been developed under different definitions of fairness, however, in this paper, we shall focus on comparing the effect that equality vs. non-equality notions of fairness may have on classical families of recommendation algorithms.

RQ3. Which user and item attributes are more sensitive to different notions of fairness? Which attribute/recommendation-algorithm combination is more prone to produce fair/unfair recommendations? Since fairness can be defined according to different user or item attributes, we aim to study the sensitivity of recommendation algorithms with respect to these parameters under the proposed probabilistic framework.

We answered the above research questions by performing extensive experiments on four well-known datasets: `Amazon Toys & Games`, `Amazon Video Games`, `Amazon Electronics`, and `MovieLens-1M`. We tuned several well-known baseline recommenders, including item and user-based nearest neighbors [72, 24] and matrix factorization as well as other techniques that optimize ranking [57, 69, 64], which were evaluated by exploiting the proposed framework to measure fairness.

To address the second research question, we considered uniform and non-uniform distributions among groups. This gave us a clear idea about how these classic recommenders behave. The third research question was addressed considering an adequate number of items and users attributes. We considered three attributes for items (Price, Year, and Popularity), and five attributes for users (Happiness, Helpfulness, Interactions, Age, and Gender). While Popularity, Happiness, and Interactions are derived from the original user-item matrix, Price and Helpfulness are two attributes that are, at the same time, dataset-specific, and sensitive attributes; moreover, Age and Gender are two user attributes that are generally considered as sensitive, because of that they are not available in all the datasets, although it should not be too difficult to gather in any recommender system. This research question imposed to re-evaluate all the baseline eight times. However, this effort is paid back by results. On the one hand, they show that some recommenders make large use of popularity and they show a non-uniform behavior. On the other hand, some interesting similarities between different attributes emerged, resulting in recommenders that are more or less prone to produce better recommendations for groups of users or items, according to these attributes.

2 Background and Prior Art

In this section, we briefly review different notions of fairness and the trade-off between fairness and accuracy-oriented metrics explored in the literature.

2.1 Fairness notions

Machine learning (ML) is now involved in life-affecting decision points such as criminal risk prediction, credit risk assessments, housing allocation, loan qualification prediction, or hiring decision making [75,79]. As ML is increasingly being employed to ease or automate decision making for humans, some concerns have been recently raised on *fairness* of such models. Over the last decade, a growing number of research articles in the ML community have focused on defining appropriate notions of fairness and then developing models to ensure fairness in automated decision making (DM). Awareness on fairness and ethics in information retrieval has been raised by Belkin and Robertson already in 1976 [18]. Generally, the current notions of fairness are mainly influenced by the concept of discrimination in social sciences, law and economy [32]. For instance, back in the 90's there was interest to measure the distribution of personal characteristics such as income or wealth for a given population. As a result, the concept of unfairness (or discrimination) referred to disproportionate distribution of these resources.[2]

Defining fairness in an algorithmic context is a subject of debate by members of the computer science community. The work by Verma et al., [79] provides the most prominent definitions of fairness for algorithmic DM in the context of classification task. Here we look more carefully into two important notions of fairness in the ML literature [84,85], that call for some form of parity (i.e., equality), either in treatment, in impact or both:

1. **Treatment disparity**: Anti-discrimination laws [1,17] in many countries prohibit unfair treatment of individuals based on their membership in protected classes (e.g., gender, race). A DM system is called to suffer from *disparate treatment* if the decision an individual receives changes with changes on her sensitive attribute information. In other words, individuals that share similar non-sensitive characteristics (e.g., qualification) are expected to receive similar decision outcomes irrespective of their sensitive attribute information such as gender or race [42, 84].
2. **Impact disparity:** A DM system is called to suffer from *disparate impact* when the decision outcomes disproportionately benefit or hurt users of certain sensitive feature value groups (e.g., women or black). In other words, "*different sensitive attribute groups are expected to receive beneficial decision outcomes in similar proportions*" [42].

As it can be noted, there exists an inevitable trade-off between such definitions of fairness that makes simultaneous control of them a challenging task. To provide the reader with an intuitive insight about fairness in treatment and impact, we present the following illustrating example.

Example 1. To better understand the difference between treatment and impact disparity, let us consider an example from automated DM for university admission process in which *fairness in treatment* implies that the DM system reviews each candidate profile with similar evaluation criteria, for instance: (i) candidate's grade-point

[2] The terms "poverty", "welfare" or "inequality" were used interchangeably in the economy literature [31,32] when referring to discrimination or unfairness.

average (GPA) and (ii) her score on TOEFL or IELTS language tests. Therefore, to achieve fairness in treatment, the system has to merely look at **candidate's merits** and **qualifications** in making the final decision, as though the system does not have access to sensitive attribute information (and thus cannot make use of it). Despite this fact, disregarding the sensitive attribute information can lead to impact disparity since automated DM systems often utilize historical training data, which can be biased or noisy. Ignoring sensitive attribute information here can imply unfairly treating a candidate for instance because she was unfairly treated in the past.

Fairness in *outcome* attempts to build a situated DM system that can accommodate for **situational/contextual characteristics**. This for instance can mean that the DM system (in the example) would take into account that some candidates might have had a disadvantaged background by acknowledging that they did not have equal opportunity to standard education or other resources; as a result, for candidates with such background information, different requirements are adopted by the selection committee when reviewing their profiles. Similarly here, using sensitive information can cause disparate treatment. Thus, we can conclude that controlling for both notions of fairness is a difficult task [5].

Although the above notions proposed in prior studies provide an attractive viewpoint, they often lack flexibility with respect to one or more of the following aspects:

1. They are specifically designed for classification problems and define fairness based on the results of confusion matrix.
2. Fairness is measured with respect to instances of the training data.

In the following, we would discuss further dimensions related to fairness and accuracy of ML and RS.

2.2 Fairness and accuracy trade-off

Recommender systems help users in many decision-making and mission-critical tasks such as entertainment, medical, financial, or job-related applications. One of the key success indicators of RS is linked with the fact that they can alleviate the information overload pressure on information seekers by offering suggestions that match their tastes or preferences. It is common to measure the quality of a personalized recommendation algorithm in terms of *relevance* (e.g., personalized ranking) metrics. In domains such as news, books, movies, and music where the individual preference is paramount, providing personalized recommendations can increase users' trust in and engagement with the system. These are important factors to motivate users to stay in and keep receiving recommendations, resulting in loyalty in the long term and offering benefits to different parties involved in a recommendation setting such as consumers, suppliers, the system designer and other related services. Even in sensitive domains such as job recommendation, where fair opportunities to job seekers is desired, personalization can be relevant, e.g., a job-seeker might be willing to compensate salary with the distance factor or other benefits.

Nonetheless, blindly optimizing for accuracy-oriented metrics (or consumer relevance) may have adverse or unfavorable impacts on the fairness aspect of recommendations [63], e.g., in the employment recommendation context, certain genders

or users from certain areas might be more likely to be recommended a job due to their behavioral differencesand past information collected from users with the same characteristics. For example, male users or users from certain regions with a high-speed internet connection may produce more clicks compared to others. A system optimizing for consumer relevance (understood as the number of clicks logged by the system) might be unfair to less active users such as females or people from areas with less internet activity thereby placing these groups at an unfair disadvantage. There exists an undeniable uncertainty in models trained on the data, e.g., since there are less data for women (in our example) or regions with less internet connectivity — as they interact less often with the system — they are more susceptible to receive low-quality recommendations. On the other hand, exposing all users equally might have a detrimental impact on the relevance and eventual consumer satisfaction. This inadvertently leads to a trade-off between relevance/personalization and fairness, since the more weight the former receives, the more exposed under-represented users would become, leading to unfair situations.

In the field of ML, Zafar et al., [83] propose a framework for modeling fairness versus accuracy trade-off in a classifier that suffers from disparate mistreatment. The proposed systems takes into account fairness and accuracy of classification in a unified system by casting them in a convex-concave optimization formulation. This results in improving the fairness criterion of classification system in which disparate mistreatment on false positive and false negative rates are eliminated. The framework allows to measure unfairness in situations where sensitive attributes of protected classes might not be accessible for reasons such as privacy or disparate treatment laws [17] prohibiting their use. In [42], Grgic-Hlaca et al., propose a fairness-aware DM system that focuses on the fairness of outcomes of ML systems. This work introduces insights into a new notion of fairness named *fairness in DM* (or process fairness), which rely on humans' moral judgments or instincts about the fairness of utilizing input attributes in a DM scenario. To this end, this work introduces different measures to model individual's moral sense in deciding whether it is fair to use various input features in the DM process. The authors show that it is possible to obtain a near-optimal trade-off between process fairness and accuracy of a classifier over the set of features and provide the empirical evidence.

In the neighboring field of information retrieval, several works have studied fairness, for instance, to investigate relevance-fairness trade-off by Mehrotra et al. [62] via auditing search engine performance for fairness, and by Biega et al. [20] as well as Singh and Joachims [74] that study fairness in the ranking. Finally, we can mention a fresh perspective on the subject of fairness studied in sociology/economy e.g., by Abebe et al., [8] that propose an approach based on the fair division of resources.

The majority of the above works focused on fairness from the perspective of users (or user fairness). On the research works that focus on the other fairness recipient, we can name the work by Mehrotra et al., [63], which exclusively focuses on supplier fairness in marketplaces. In [76] Sühr et al., investigate the means to achieve *two-sided fairness*, in a ride-hailing platform by spreading fairness over time showing that this approach can enhance the overall utility for the drivers and the passenger.

2.3 From reciprocal recommendation to multiple stakeholders

Reciprocal recommendation views RS as systems fulfilling dual goals; the first goal is associated with satisfying customers' preference — i.e., user-centered utility — while the other purpose is quite often related to the value of recommendations to the vendors — i.e., vendor-centered utility (e.g., profitability) [10]. Reciprocal recommendation regards the recommendation in most scenarios similar to a transaction and states that in generating recommendation, *bilateral considerations* should be made, meaning that the recommendations must be appealing to both parties involved in a transaction. On the domains, which use reciprocal recommendation we can name on-line dating, on-line advertising, scientific collaboration and so on [25]. Maintaining a balance between the user and the vendor-centered utilities is the focal attention of RS acknowledging this viewpoint to the recommendation. In [10], Akoglu et al., propose ValuePick, a framework that integrates the proximity to a target user and the global value of a network to recommend relevant nodes within a network. Several approaches have been proposed to combine the utilities as mentioned above to either optimize profitability or to generate a win-win situation for providers and consumers [50] – according to which recommended items are ranked, see, e.g., [50, 29, 66]. From a technical perspective, various approaches are proposed for instance, based on the heuristic scoring model [29], mathematical optimization model [10, 14, 36], reinforcement learning [73, 55], and more complex models. Some approaches have attempted to place into a mathematical optimization problem additional constraints such as consumer budget and other decision factors, for example, customer satisfaction levels [80]. Systems designed to meet the requirements of multiple stakeholders are referred to as multi-stakeholder recommender systems (MRS) [26]. MRS can be seen as an extension to reciprocal recommendation where the system must account for the needs of more than just the two transacting parties. For instance, Etsy [4] is an e-commerce website focused on handmade products and craft supplies. The recommender system platform in Etsy provides recommendations from small-scale artisans to consumers (shoppers). Hence, the recommender system on such a website needs to deal with the needs of both consumers and sellers [59]. According to Burke et al., [26], we can classify multiple stakeholders involved in an MRS into three main groups: consumers, providers, and the platform (the system). Fairness is a *multisided concept* in which the impact of the recommendation on multiple groups of individuals must be considered. The authors propose to study the fairness issues relative to each one of these groups according to (i) consumers (C-fairness), (ii) providers (P-fairness), and (iii) both (CP-fairness). The authors propose balanced neighborhoods, a mechanism to make a reasonable trade-off between personalization vs. fairness of recommendation outcomes.

Several works have been proposed for evaluating recommendations in MRS. In [7, 27, 89] the authors suggest a utility-based framework for representing multiple stakeholders. As an example, in [89] Zheng et al., propose a utility-based framework for MRS for personalized learning. Specifically, a recommender system is built for suggesting course projects to students by accounting both the student preferences and the instructors' expectations in the model. The model aims to address the challenge of over-expectations (by instructors) and under-expectations (by students) in the utility-

based MRS. Surer et al., [77] approach the MRS issue differently by formulating the problem as a constraint-based integer programming (IP) optimization model, where different sets of constraints can be used to characterize the objectives of different stakeholders. A recent survey [6] by Abdollahpouri et al., provides a good understanding of the MRS topic, providing insights into origins and discussing state-of-the-art in the MRS field.

2.4 Evaluating fairness in recommender systems

Even though research on fairness has been a very active topic in ML community in general, as well as in RS, there are not any works —to the best of our knowledge— where authors address the goal we aim to achieve here: "*propose an evaluation metric that is capable of measuring fairness in RS*". The closest work is [78], where Tsintzou et al., define a metric named "bias disparity" to capture the relative change of biases between the recommendations produced by an algorithm and those biases inherently found in the data. For this, the authors need to categorize both users and items, hence, it is not possible to measure only user or item fairness as allowed by our framework. Moreover, the most important disadvantage of the proposed metric is that the authors do not provide a single value for a given recommender, but a table (similar to a confusion matrix or contingency table) with all the possible combinations for pairs of user and item categories. The proposed evaluation metric in the current work in hand (see Section 3.1) could be interpreted as an aggregation of several values (one for each attribute) by tabulating the data inside the integral allowing us to create a table like the one reported in [78]; however, we prefer not to report the outcome as a table but instead provide a metric that follows the standard definitions in RS and IR evaluation, that is, that returns one value for each user/item.

Nonetheless, even though we have not found other papers specifically tackling the problem of defining a fairness evaluation metric, papers that propose algorithms tailored for fairness need to be evaluated somehow, and these metrics, although usually based on heuristics, can also be considered to evaluate fairness. We start by describing the purely theoretical survey presented in [79], where the authors collect many definitions from the literature about the concept of fairness. The following three could be easily applied in a recommendation context: group fairness (equal probability of being assigned to the positive predicted class), predictive parity (correct positive predictions should be the same for both classes), and overall accuracy equality (groups have equal prediction accuracy). The last two could be computed by measuring the precision or the error in each class and somehow comparing those values across all the groups. This is exactly the idea behind MAD (absolute difference between the mean ratings of different groups) used in [90]. Here, the authors Zhu et al., also use in their experiments the Kolmogorov-Smirnov statistics of two distributions (predicted ratings for groups) as a comparison. The main problem with these two approaches and with some of the definitions in [79] is that they are only valid for 2 groups and are focused on ratings —and, consequently, only valid for the rating prediction task, which has been displaced by the community because it does not correlate with the user satisfaction [43, 61]—, mostly because fairness is addressed as a classification

problem in ML. We find the same situation in [82] where the authors Yao et al., define several unfairness quantities (non-parity, value, absolute, underestimation, overestimation, and balance unfairness) that can only be applied to two groups of users and based on prediction errors.

Finally, we found other types of metrics not directly based on prediction errors. On the one hand, in [59] Liu et al., define a metric tailored for P-fairness (fairness from the perspective of the providers in a multi-stakeholder setting) based on the provider coverage, that is, the number of providers covered by a recommendation algorithm. On the other hand, in [70] Sapiezynski et al., use the Matthew's correlation coefficient, since it allows to quantify the performance of an algorithm at a threshold while, at the same time, it penalizes the classifier for classifying all samples as the target class. In the paper, as some of the metrics presented above, the coefficient is defined only for the binary case where the attribute has two possible values, however, it is possible to compute a multiclass version. Nevertheless, as proposed by the authors, it can only be applied to user attributes.

Summing up, several metrics have been used to evaluate RS under different notions of fairness. Their limitation can be summarized as follows (i) they tend to promote the notion of equality between groups constructed by sensitive attributes; for example, the metric MAD [90] introduced earlier is minimized under equal performance between two groups; (ii) they are often limited to user attributes that can be binarized; (iii) they may not be able to isolate user-fairness and item-fairness evaluation and study them in isolation, such as the bias disparity metric introduced in [78]. Instead, we believe the framework we present in this paper could open up several possibilities in the field, since it overcomes all the above-mentioned limitations.

3 A probabilistic framework to evaluate fairness

We now present a probabilistic framework for evaluating RS fairness based on attributes of any nature (e.g., sensitive or insensitive) for both items or users and show that the proposed framework is flexible enough to measure fairness in RS by considering fairness as equality or non-equality among groups, as specified by the system designer or any other parties involved in a multi-stakeholder setting.

In this section, we propose a framework based on generalized cross-entropy for evaluating fairness in RS. Let U and I denote a set of users and items, respectively and A a set of sensitive attributes, related to users or items, in which fairness is desired. Each attribute can be defined for either users, e.g., gender and race, or items, e.g., item provider (or stakeholder). Given a set M (for models) of recommendation systems, we define the *unfairness measure* as the function

$$\omega \,:\, M \times A \rightarrow \mathbb{R}^{+}$$

The goal is to find a function ω that produces a non-negative real number for a recommender system that represents and measures its (un)fairness. A recommender model $m \in M$ is considered less unfair (i.e., more fair) than $m' \in M$ with respect to the

attribute $a \in A$ if and only if $\omega(m,a) < \omega(m',a)$ [75]. Previous works have used *inequality* measures to evaluate algorithmic unfairness, however, we argue that fairness does not always imply equality.

For instance, let us assume that there are two types of users in the system – regular (free registration) and premium (paid) – and the goal is to compute fairness concerning the users' subscription type. In this example, it might be more fair to produce better recommendations for paid users, therefore, equality is not always equivalent to fairness – note that, in any case, the goal is to ensure that premium users receive good (or better) recommendations without affecting the experience of regular users. As an example, in a car navigation system that takes into account real-time traffic information, it might be important to recommend different routes to users going in the same direction. If they are all recommended to follow the same shortest (in terms of time) path they might create a traffic jam thus giving to the users the feeling that the recommendation engine is not working well. The point is, given a set of possible paths to recommend having the same travel time, how to distribute the recommendations among the different users? A possible solution could be that of recommending scenic (better) routes to premium users first and urban routes to free ones. In this case, concerning the scenic/urban attribute, we have a non-equal behavior but, nonetheless, the experience of regular users in terms of travel time is not affected by the choice of the algorithm.

In this respect, the proposed recommendation does not introduce any unfair behaviour among users regarding the final goal of the system and, at the same time, it fairly takes into account the differences among users to differentiate the final result. Once more, we wish to stress here that we do not want to deliberately differentiate between users. Our proposal bases on the exploitation of items information and knowledge (attributes) that does not affect the user utility of the final recommendations to provide fair diversification in the results.

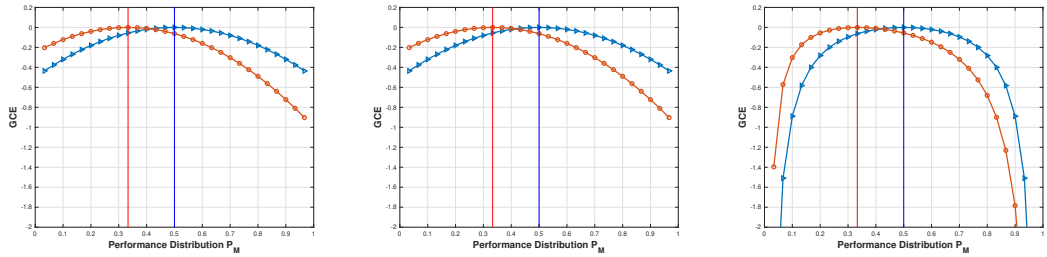
In fact, in some tasks/domains, there might be a "cost" factor regarding the delivery/fruition of certain items. As an example, there could be "item supply" costs in the e-commerce scenario, different "copyright" costs in streaming platforms, or "system performance" costs in edge computing domains. Moreover, in some situations, there might be an "additional advantage" that the system can exploit (if delivered items belong to specific classes) without harming the users' main utility.

3.1 Using Generalized Cross Entropy to measure user and item fairness

We define fairness of a recommender system with respect to an attribute $a \in A$ using the *Csiszar generalized measure of divergence* as follows [34]:

$$\omega(m,a) = \int p_m(a) \cdot \varphi\left(\frac{p_f(a)}{p_m(a)}\right) \, \mathrm{d}a \tag{1}$$

where p_m and p_f respectively denote the probability distribution of the model m's performance and the fair probability distribution, both with respect to the attribute $a \in A$ [23]. A distinguishing property of this measure is that conceptually there are no differences for the case in which p_m and p_f are discrete densities, in such a case



(a) $\beta = -1$ (b) $\beta = 0.5$ (c) $\beta = 2$

Fig. 1 Simulations of values obtained using GCE fairness evaluation metric for different fair distribution types p_f and performance distributions p_m and different β values. For example, when x-axis is 0.3 then $p = [0.3, 0.7]$. The **blue** curve represents $p_f = [0.5, 0.5]$ while the **red** represents $p_f = [0.3, 0.7]$. It can be noted when fairness means equality the representative **blue** curve is used, which is maximized at 0.5; this is while when fairness means non-equality the representative **red** curve should be used, which is maximized at a point non-equal to 0.5 (here 0.3). Curves under different β values differ mainly in their slope toward extremely high (or low) values for p_m.

the integral is simply replaced by the sum. Csiszar's family of measures subsumes all of the information-theoretic measures used in practice (see [54,45]). We restrict our attention to the case when $\varphi(x) = \frac{x^{\beta} - x}{\beta \cdot (\beta - 1)}$ and $\beta \neq 0, 1$ for some parameter β; then, the family of divergences indexed by β boils down to the *Generalized Cross Entropy (GCE)*

$$GCE(m,a) = \frac{1}{\beta \cdot (1-\beta)} \left[\int p_f^{\beta}(a) \cdot p_m^{(1-\beta)}(a) \ \mathrm{d}a - 1 \right] \tag{2}$$

The unfairness measure ω is minimized with respect to attribute $a \in A$ when $p_m = p_f$, meaning that the performance of the system is equal to the performance of a fair system. In the next sections, we discuss how to obtain or estimate these two probability distributions. In the appendix, we present a theoretical analysis of the appropriateness of this metric to measure fairness.

Note that the defined unfairness measure indexed by β includes the Hellinger distance for $\beta = 1/2$, the Pearson's χ^2 discrepancy measure for $\beta = 2$, Neymann's χ^2 measure for $\beta = -1$, the Kullback-Leibler divergence in the limit as $\beta \to 1$, and the Burg CE distance as $\beta \to 0$. Figure 1 illustrates simulations of how GCE changes across different β values.

If the attribute $a \in A$ is discrete or categorical (as typical attributes, such as gender or race), then the unfairness measure is defined as:

$$GCE(m,a) = \frac{1}{\beta \cdot (1-\beta)} \left[\sum_{a_j} p_f^{\beta}(a_j) \cdot p_m^{(1-\beta)}(a_j) - 1 \right] \tag{3}$$

The role of β in the definition of GCE is critical, as we show in Figure 1. We observe, for instance, that at extreme values of p_m, GCE obtains larger values for lower values of β. Besides, according to [23], Pearson's χ^2 measure (which corresponds

to $\beta = 2$) is more robust to outliers than other typical divergence measures such as Kullback-Leibler divergence; hence, in the rest of this paper, unless stated otherwise, we shall use this value for parameter β.

It should be noted that it would be straightforward to extract information for each attribute value, as done in [78], and obtain a contingency table, however we believe that an aggregation of values as presented in Equation 3 is easier to comprehend than such tabulated information.

3.1.1 Defining the fair distribution p_f

The definition of a fair distribution p_f is problem-specific and should be determined based on the problem or target scenario in hand. For example, one may want to ensure that premium users, who pay for their subscription, would receive more relevant recommendations because running complex recommendation algorithms might be costly and not feasible for all users.[3] In this case, p_f should be non-uniform across the user classes (premium versus free users). In other scenarios, a uniform definition of p_f might be desired. Generally, when fairness is equivalent to equality, then p_f should be uniform and in that case, the generalized cross-entropy would be the same as generalized entropy (see [75] for more information).

Note that p_f can be seen as a more general utility distribution, and the goal is to observe such distribution in the output of the recommender system. In this paper, since we focus on recommendation fairness, we refer to p_f as the fair distribution.

Finding fair distribution p_f is challenging. It is task-specific and a fair distribution in one domain is not necessarily a fair distribution in another. However, generalized cross-entropy is a general framework that allows researchers and practitioners in different domains to define the fairness definition based on their needs. We leave discussions on the various definitions of p_f in different domains for the future.

3.1.2 Estimating the model distribution p_m

The model distribution p_m should be estimated based on the output of the recommender system on a test set. In the following, we explain how we can compute this distribution for item attributes. We define the recommendation gain (rg_i) for each item $i \in I$ as follows

$$rg_i = \sum_{u \in U} \phi(i,\ Rec_u^K) \cdot g(u,i,r) \tag{4}$$

where Rec_u^K is the set of top-K items recommended by the system to the user $u \in U$. $\phi(i,\ Rec_u^K) = 1$ if item i is present in Rec_u^K; otherwise $\phi(i,\ Rec_u^k) = 0$. The function $g(u,i,r)$ is the gain of recommending item i to user u with the rank r. Such a gain function can be defined in different ways. In its simplest form, if $g(u,i,r) = 1$, the recommendation gain in Eq. (4) would boil down to recommendation count (i.e., $rg_i = rc_i$).

[3] These scenarios are becoming more and more realistic especially in edge computing settings where computational resources are often quite limited.

A binary gain in which $g(u,i,r) = 1$ when item i recommended to user u is relevant and $g(u,i,r) = 0$ otherwise, is another simple form of the gain function based on relevance. The gain function g can be also defined based on ranking information, i.e., recommending relevant items to users in higher ranks is given a higher gain. In such a case, we propose to use the discounted cumulative gain (DCG) function that is widely used in the definition of nDCG [53], given by $\frac{2^{\text{rel}(u,i)}-1}{\log_2(r+1)}$ where $\text{rel}(u,i)$ denotes the relevance label for the user-item pair u and i. We can further normalize the above formula based on the ideal DCG for user u to compute the gain function g.

As we can see in the definition of the gain function for items, it is possible to flexibly specify the constraint under which fairness needs to be satisfied (e.g., based on recommendation count, relevance, ranking, or a combination thereof). As such, our approach extends considerably the previous approaches, e.g., [20,74,87] which focused on a single aspect of fairness, e.g., either based on error or ranking.

Then, the model probability distribution p^I_m is computed proportionally to the recommendation gain for the items associated to an item attribute value a_j. Formally, the probability $p^I_m(a_j)$ used in Eq. (3) is defined as:

$$p^I_m(a_j) = \frac{1}{Z} \sum_{\{i \in I : a_i = a_j\}} rg_i \tag{5}$$

where Z is a normalization factor set equal to $Z = \sum_i rg_i$ to make sure that $\sum p^I_m(a_j) = 1$. Under an analogous formulation, we could define a variation of fairness for users $u \in U$ based on Eq. (4)

$$rg_u = \sum_{i \in I} \phi(i,\ Rec^K_u) \cdot g(u,i,r) \tag{6}$$

where in this case, the gain function cannot be reduced to 1, otherwise, all users would receive the same recommendation gain rg_u. Then, to compute $p^U_m(a_j)$, we similarly normalize these gains as shown by Eq. (5).

It should be noted that, to avoid zero probabilities, we smoothed the previous computations by using the Jelinek-Mercer method [88] as follows, where p^E_m corresponds to either p^I_m or p^U_m depending if rg_i or rg_u are used:

$$\begin{aligned}
\widetilde{p}^E_m(a_j) &= \frac{1}{Z} \sum_{\{e \in E : a_e = a_j\}} rg_e \\
\widehat{p}^E_m(a_j) &= \lambda \cdot \widetilde{p}^E_m(a_j) + (1-\lambda) \cdot p_C \\
\widehat{Z} &= \sum_j \widehat{p}^E_m(a_j) \\
p^E_m(a_j) &= \frac{\widehat{p}^E_m(a_j)}{\widehat{Z}}
\end{aligned}$$

Here, smoothing is applied in the second equation, where we use a background probability p_C. In the experiments, we used $\lambda = 0.95$ and $p_C = 0.0001$. Additionally, to obtain more robust values of the probabilities estimated using the recommendation gains, a slightly more complicated version of these formulations could be used where

Table 1 A set of 6 users belonging to groups (classes) g1 and g2 and 10 items along with their true labels marked by ✓and 3 recommended items by recommenders Rec 0, Rec 1, Rec 2. All recommendation lists recommend Top@3 items. Over all items recommended, in total Rec 0 is able to recommend 3 relevant items for free users and 6 relevant items for premium users respectively; Rec 1 generates 1 relevant item for each user regardless of her/his class; Rec 2 can recommend Top@3 items that are all relevant (i.e., ideal precision equal to 1) for all users regardless of their class. The relevant items are marked as bold in each recommendation list.

		True Items										Actually recommended		
		i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	**Rec 0**	**Rec 1**	**Rec 2**
a_1	**user 1**	✓		✓				✓				$\{\mathbf{i}_1, i_6, i_8\}$	$\{\mathbf{i}_1, i_5, i_9\}$	$\{\mathbf{i}_1, \mathbf{i}_3, \mathbf{i}_7\}$
a_1	**user 2**					✓			✓			$\{i_2, \mathbf{i}_5, i_9\}$	$\{i_2, \mathbf{i}_5, i_7\}$	$\{i_1, \mathbf{i}_5, \mathbf{i}_8\}$
a_1	**user 3**		✓					✓				$\{i_1, i_6, \mathbf{i}_7\}$	$\{\mathbf{i}_2, i_5, i_9\}$	$\{\mathbf{i}_2, \mathbf{i}_7, i_9\}$
a_2	**user 4**			✓	✓					✓		$\{\mathbf{i}_3, \mathbf{i}_6, \mathbf{i}_9\}$	$\{\mathbf{i}_4, i_5, i_6\}$	$\{\mathbf{i}_3, \mathbf{i}_4, \mathbf{i}_9\}$
a_2	**user 5**					✓		✓			✓	$\{i_1, \mathbf{i}_5, \mathbf{i}_7\}$	$\{i_1, i_2, \mathbf{i}_{10}\}$	$\{\mathbf{i}_5, \mathbf{i}_7, \mathbf{i}_{10}\}$
a_2	**user 6**	✓		✓			✓			✓		$\{i_2, \mathbf{i}_6, \mathbf{i}_9\}$	$\{\mathbf{i}_1, i_5, i_8\}$	$\{\mathbf{i}_3, \mathbf{i}_6, \mathbf{i}_9\}$

Table 2 Fairness of different recommenders in the toy example presented in Table 1 according to proposed GCE and individual-level accuracy metrics. Note that $p_{f_0} = [\frac{1}{2}, \frac{1}{2}]$, $p_{f_1} = [\frac{2}{3}, \frac{1}{3}]$, and $p_{f_2} = [\frac{1}{3}, \frac{2}{3}]$ characterize the fair distribution as uniform or non-uniform distribution (of resources) among two groups.

	GCE $(p_f, p_m, \beta = 2)$			**P@3**	**R@3**
	p_{f_0}	p_{f_1}	p_{f_2}		
Rec 0	-0.0952	-0.3201	-0.0026	$\frac{1}{2}$	$\frac{1}{6} \cdot \frac{19}{6} = 0.530$
Rec 1	0	-0.0556	-0.0556	$\frac{1}{3}$	$\frac{1}{6} \cdot \frac{9}{4} = 0.375$
Rec 2	-0.0079	-0.1067	-0.0220	1	$\frac{1}{6} \cdot \frac{23}{4} = 0.958$

the probabilities consider the average of gains rg_e in a user-basis instead of such gains directly, since this is how typical evaluation metrics are computed in the RS literature. For the sake of space, we avoid including such formulation here.

3.2 Toy example

For the illustration of the proposed concept, in Table 1 we provide a toy example on how our approach for fairness evaluation framework could be applied in a real recommendation setting. A set of six users belonging to two groups, each group is associated with an attribute value a_1 (red) or a_2 (green), who are interacting with a set of items is shown in Table 1. Let us assume the red group represents users with *regular* (free registration) subscription type on an e-commerce website while the green group represents users with *premium* (paid) subscription type. A set of recommendations produced by different systems (**Rec0**, **Rec1**, and **Rec2**) is shown in the last columns. The goal is to compute fairness using the proposed fairness evaluation metric based on GCE given by Eqs. (3) and (6). The results of the evaluation using three different evaluation metrics are shown in Table 2. The metrics used for the evaluation of fairness and accuracy of the system include (i) GCE, (ii) Precision, and (iii) Recall, all at cutoff 3. Note that $GCE = 0$ means the system is completely fair, and the closer the value is to zero, the more fair the respective system is.

By looking at the recommendation results from **Rec0**, one can note that *if fairness is defined as equality between two groups*, defined through fair distribution p_f = $[\frac{1}{2}, \frac{1}{2}]$, then **Rec0** is not a completely fair system, since $GCE = -0.09 \neq 0$. In con-

trast, *if fairness is defined as providing recommendation of higher utility (usefulness) to green users who are users with paid premium membership type*, (e.g., by setting $p_{f_2} = [\frac{1}{3}, \frac{2}{3}]$) then, since GCE is smaller, we can say that recommendations produced by **Rec0** are more fair for this type of users and also with respect to the other recommenders. Both of the above conclusions are drawn concerning the attribute "subscription type" (with categories free/paid premium membership). This is an interesting insight that shows the evaluation framework is flexible enough to capture fairness based on the interest of the system designer by defining what she considers as fair recommendation through the definition of p_f. While in many application scenarios we may define fairness as equality among different classes (e.g., gender, race), in some scenarios (such as those where the target attribute is not sensitive, e.g., regular vs. premium users) fairness may not be equivalent to equality.

Furthermore, by comparing the performance results of **Rec1** and **Rec2**, we observe that, even though precision and recall improve for **Rec2** and becomes the most accurate recommendation list, it fails to keep a decent amount of fairness for every parameter settings of GCE, as in all the cases it is outperformed by the other methods. Moreover, GCE only reaches the optimal value for **Rec1** and p_{f_0}, since that recommender produces the same number of relevant items (one) for every user, independently of the user group; in the other cases, since there are more relevant items on the green than red users, the results reflect the amount of inherent biases in the data due to the unequal distribution of resources among classes.

This evidences that optimizing an algorithm to produce relevant recommendations does not necessarily result in more fair recommendation rather, conversely, a trade-off between the two evaluation properties can be noticed.

4 Experimental settings

In this section, we describe in detail the experimental setting adopted to validate the proposed fairness evaluation model for RS.

4.1 Datasets

To address the research questions presented in Section 1, we use datasets from different domains with more or less sensitive attributes. This allows us to evaluate several notions of fairness under user and item dimensions. More specifically, we have used multiple product categories of the Amazon Review dataset [46,3]. This dataset is a collection of product reviews aggregated at the category level, which also includes metadata from Amazon; in total it contains 142.8 million reviews spanning from May 1996 to July 2014. Beyond ratings, these datasets include reviews (which consist of ratings, text, timestamp, and votes from other users to determine how helpful a review is), product metadata (descriptions, category information, price, brand, and image features [60]), and links (graphs with information about also viewed/also bought items).

Overall the `Amazon Dataset` contains 24 different category-level datasets. Based on the number of users, items, and transactions we have selected the following four

datasets to conduct our study. The smallest one is `Amazon Video Games`, with more than 1 million ratings, devoted to videogames sold on the Amazon Store. The second dataset is `Amazon Toys & Games`, with more than 2 million transactions of toys and tangible games. The last and largest dataset is `Amazon Electronics`, with almost 8 million overall ratings. Finally, we have also considered a classic recommender systems dataset, MovieLens 1 Million (`MovieLens-1M`), that contains 1,000,209 transactions on the popular movie platform Movielens. It collects user feedback in the movie domain on a 5-star scale, considering 6,040 users, and almost 3,900 items. Additionally, the dataset provides users' and items' metadata, like user age, gender, and occupation, while item descriptions contain the title, the distribution year, and the genres.

4.2 Evaluation Protocol and Temporal Splitting

The experimental evaluation is conducted adopting the so-called "All Items" evaluation protocol [19] in which, for each user, all the items that are not rated yet by the user are considered as candidates when building the recommendation list.

To simulate an online real scenario as realistically as possible, we use the fixed-timestamp splitting method [13, 12], initially suggested in [28, 43]. The core idea is choosing a single timestamp that represents the moment in which test users are on the platform waiting for recommendations. Their past will correspond to the training set, whereas the performance is evaluated exploiting data that occurs after that moment. In this work, we select the splitting timestamp that maximizes the number of users involved in the evaluation by setting two reasonable constraints: the training set of each user should keep at least 15 ratings, while the test set should contain at least 5 ratings; these thresholds were selected to keep a decent amount of users both in training and test while having enough information in each split to train the recommendation algorithms and compute the evaluation metrics. Training set and test set for the four datasets are made publicly available for research purposes, along with the splitting code.[4]

Finally, the statistics of the training and test datasets used in the experiments are depicted in Table 3, where the difference in the number of transactions between the original datasets (see the previous section) and the ones used in the experiments is due to the constraints imposed in the splitting process. It is important to note that, in any case, the processed datasets keep very small density values – between 0.054% and 0.48% – as it is standard in the literature. Conversely, this severe splitting strategy is not compatible with more classic (and smaller) datasets, like `MovieLens-1M`. In `MovieLens-1M` a fixed timestamp splitting removes the majority of the transactions. To include this classic recommender systems dataset, we have opted for a more lazy temporal hold-out splitting. Even here, we split training and test data temporally, by retaining the first 80% of user history as the training set, and the remainder as the test set. However, in this setting, the split is made on a user-basis, by computing a splitting timestamp for each user.

[4] `https://github.com/sisinflab/DatasetsSplits/`

Table 3 Statistics about the datasets used in the experiments.

			Training Set			
Dataset	**#Users**	**#Items**	**#Transactions**	**Sparsity**	**From**	**To**
Amazon Electronics	5,351	56,727	164,375	99.94584	07/14/1999	05/14/2013
Amazon Toys & Games	1,108	24,158	38,317	99.85685	07/22/2000	08/30/2013
Amazon Videogames	479	8,892	20,369	99.52177	11/18/1999	10/28/2011
MovieLens - 1M	6,040	3,667	800,193	96,38718	04/25/2000	02/24/2003

			Test Set			
Dataset	**#Users**	**#Items**	**#Transactions**	**Sparsity**	**From**	**To**
Amazon Electronics	5,351	28,792	74,090	99.95191	05/15/2013	07/23/2014
Amazon Toys & Games	1,108	9,192	15,169	99.85106	08/31/2013	07/22/2014
Amazon Videogames	479	4,171	8,114	99.59387	10/29/2011	07/21/2014
MovieLens - 1M	6,040	3,535	200,016	99,06322	04/25/2000	02/28/2003

4.3 Attribute selection and discretization

In this work, we follow an attribute-based analysis of fairness in RS. In particular, we assume that users and items are associated with some attributes. Each attribute partitions the users/items into a number of groups (classes) where users/items in each group have the same attribute value (e.g., male or female for users) or (e.g., low-priced or high-priced items). One of the main objectives in the attribute-based study of fairness is to avoid discrimination against protected groups; as such these attributes are quite often chosen as nontrivial or (in some cases) sensitive. Therefore, in this section, we describe which user and item attributes were selected and how they were discretized in a limited number of groups or classes.

We start by selecting some attributes that we feel are common enough to be found in almost any recommender system, in this way, the presented analysis could be relevant for both researchers that use domains not addressed in this work and industry practitioners with different data. For items, we focus on their popularity, which corresponds to the number of interactions received by the items. Since the popularity of items is a signal of the common ratings (or clicks, views, etc.) between users, we aim to explore whether the most common CF algorithms are more prone to suggest popular items. Similarly, for users we focus on the number of interactions registered by the system from each user, that is, the level of user activity. In this way, we aim to analyze the behavior of algorithms with respect to cold (i.e., user with few interactions) or warm (i.e., users with many interactions) users, as they are topically referenced in the literature. Additionally, we interpret the average rating provided by the users as a signal of the level of satisfaction with respect to the system, we name this user feature as *happiness*. In our experiments we aim to investigate whether the recommenders behave fairly for satisfied (happy) and unsatisfied (unhappy) users.

Now, we select two attributes that are more specific to Amazon datasets and that are, to some extent, sensitive for both users and system developers: item price and user helpfulness. The price of an item is indeed an interesting and sensitive attribute,

since many users may decide to select or buy a product just because of its price, even when they know that another product might be more beneficial or suitable for them. Hence, by including this attribute we aim to study whether classical recommendation approaches are more (or less) prone to recommend expensive or cheap products – without including such information into the recommendation algorithm – which might be perceived as not fair from the user perspective. The user helpfulness, on the other hand, is a piece of information that is not widely available, but it is becoming a frequent signal in review-based systems, since it allows users to vote on other users' reviews, increasing the confidence on the system. In this way, we aim to analyze if the most helpful users are provided with the best recommendations or not.

Finally, we select two attributes that can be found – or at least, requested for – in any recommendation system, however, for privacy concerns they are not usually included in public datasets: age and gender of users. Since these attributes are highly sensitive, among the datasets considered in this work, they are only available in `MovieLens-1M`. Hence, we aim to analyze whether the recommenders behave in a similar way regarding the different classes of these attributes, that is, if males and females[5] receive recommendations of the same quality, and similarly for young or older people (see later for a more detailed specification of the actual ranges considered).

Once the different user and item attributes are selected, we present how we discretized their values into a small number of classes or clusters. This step is not mandatory since our proposed metric could work with any number of categories or attribute values, however, to make the presentation and discussion of results less cumbersome and confusing, we prefer to limit the number of categories to a maximum of 4 in every case. In general, we decided to create clusters based on quartiles, which are particularly intuitive and allow to be generalizable to datasets of different nature, since the intrinsic distribution of the attributes is taken into account. More specifically, **item price**, **user helpfulness**, and **user interaction** were directly clustered into 4 quartiles according to their original distributions. However, the rest of the attributes presented some problems which made it impossible to apply a standard clustering technique based on the quartiles. First, the **item popularity** showed so many ties for the least popular items that it was not possible to define boundaries for the quartiles; for instance, in `Amazon Electronics` the least $34,955$ items had only 1 rating, while the next $8,719$ items had only 2 ratings, and so on.. To address this issue, we increased the number of considered quantiles until we obtained 4 distinct clusters; this number corresponds to 30 for `Amazon Toys & Games`, and 10 for `Amazon Video Games` and `Amazon Electronics`. Regarding the last attribute, **user happiness**, we faced a different problem, where the average user ratings is approximately 4. As an example, in `Amazon Electronics`, the average user rating is 4.2, and 63.82% of the user ratings are between 3.5 and 4.5, hence, the clustering based on quartiles would have lost meaning. For this reason, we decided to set a reasonable threshold equal to 4 (common to the four datasets) to create only two categories: users whose average rating is smaller than 4, and the rest, to separate users according to a predefined level of satisfaction or happiness. Finally, for `MovieLens-1M`, we have analyzed three additional

[5] We need to resort to a binary classification for gender since this is the information available in this dataset.

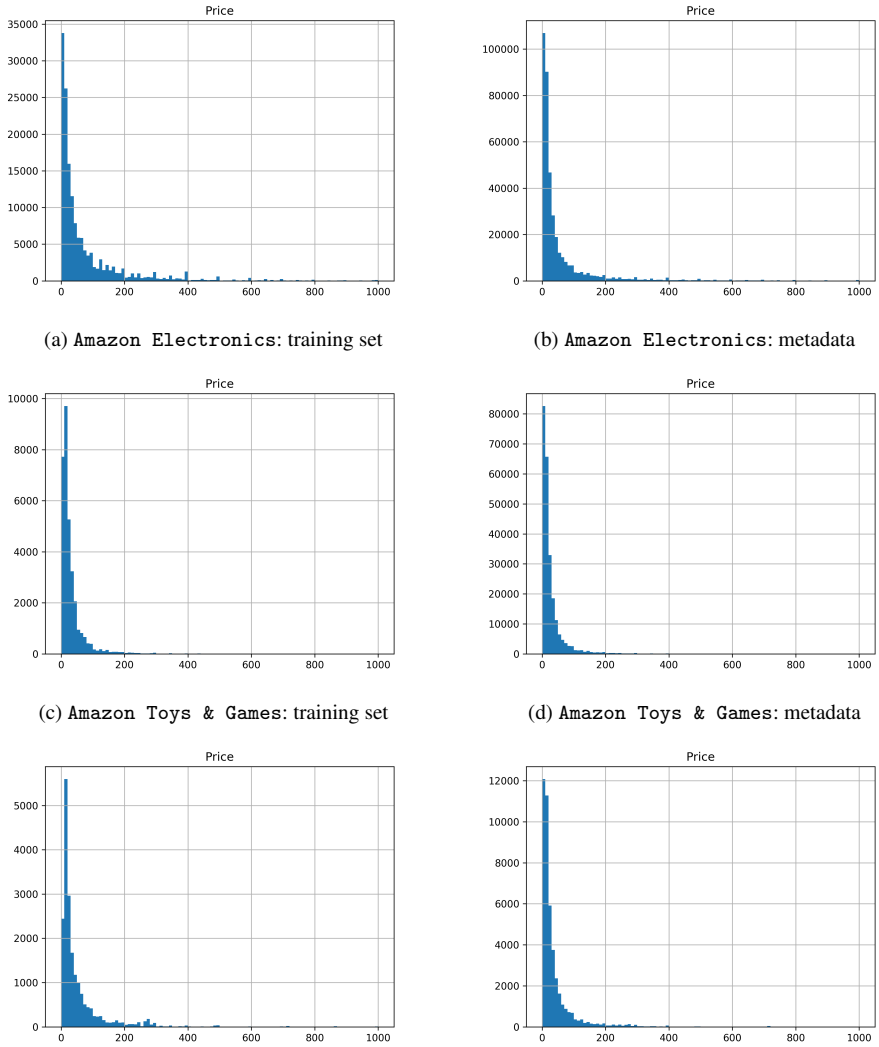
clusterings based on the available metadata: user age, user gender, and the distribution of item year. In detail, user age, and user gender are categorical features, while item year is numerical. Regarding the **item year**, we have considered the same technique depicted before that makes use of quartiles. Concerning the **user age**, we have built four age groups from the original age categories to make their size the most similar to each other. For **user gender**, the two groups are already naturally clustered, even though these groups are unbalanced. Tables 4-7 present statistics about the resulting clusterings, respectively for `Amazon Toys & Games`, `Amazon Video Games`, `Amazon Electronics`, and `MovieLens-1M`.

Finally, we note an issue we had to address regarding the computation of quantiles with respect to the availability of side information. First, not all items had associated metadata, whereas this is true for users, information for items is incomplete. Second, items in the training set only correspond to a small fraction of the items in the whole collection; hence, they might not be representative of the entire collection. Because of this, we computed the quartiles (for the item price attribute, which is the only one obtained through the metadata) according to two strategies: either based on the overall metadata information or based only on the items with metadata that appear in the training set. This information is included in Tables 4-6 in columns *Price (TS)* for the case where the clustering is computed based on the training set, and in *Price (M)* when the whole metadata is used. Additionally, in Figure 2 we present the histograms of the 3 datasets comparing the two strategies to compute the quartiles. In the tables we observe that the resulting item distribution in clusters when using all the metadata is no longer uniform; similarly, in the histograms we see that the distribution is dominated by those very cheap items when using all metadata information, whereas other price values become visible when only the training items are represented. Hence, because of these issues, we shall work from now on with the strategy based on building the clusters using information from the training set.

4.4 Baseline recommenders

We evaluate several families of Collaborative Filtering recommendation models. Beyond Nearest Neighbors memory-based models, we include latent factors models considering two different kinds of optimization: the minimization of the prediction error, and a pairwise learning-to-rank approach. More specifically, we include:

- `Random`, a non-personalized algorithm that produces a random recommendation list for each user. The items are chosen according to a uniform distribution.
- `MostPopular`, a non-personalized algorithm that produces the same recommendation list for all the users. This list is computed by measuring the items' popularity and ordering the items according to that value in descending order. It is acknowledged that popularity ranking typically show very good performance because of statistical biases in the data [19] and it is an important baseline to compare against [33].
- `ItemKNN` [71,72], an item-based implementation of the K-nearest neighbor algorithm. It finds the K-nearest item neighbors based on a specific similarity function. Usually, as similarity functions, Binarized and standard Cosine Vector Sim-



(a) `Amazon Electronics`: training set

(b) `Amazon Electronics`: metadata

(c) `Amazon Toys & Games`: training set

(d) `Amazon Toys & Games`: metadata

(e) `Amazon Video Games`: training set

(f) `Amazon Video Games`: metadata

Fig. 2 Histograms of the item price attribute (considering 100 bins) comparing two strategies to extract the values from (that will be used later to compute the attribute categories): based on items from the training set or based on all the items with associated metadata.

ilarity [16,21,58], Jaccard Coefficient [40,68], and Pearson Correlation [48] are considered. The items in the neighborhood are then used to predict a score for each user-item pair.

– `UserKNN` [24], a user-based implementation of the K-nearest neighbor algorithm. It finds the K-nearest user neighbors based on a similarity function (usually the

Table 4 Statistics about the user and item clustering methods for `Amazon Toys & Games`, where TS means Training Set, M stands for Metadata, Pop Popularity, Hlpf Helpfulness, Int Interactions, and Hpns Happiness. The rows 25%, 50% and 75% indicate the values of each attribute at that point of the distribution, which correspond to the boundaries between the first and second, second and third, and third and fourth quartiles. Note that, ideally, the number of items (#Items) and users (#Users) in each cluster is expected to be as uniform as possible.

	Items Clusterings			Users Clusterings		
Statistics	**Price (TS)**	**Price (M)**	**Pop**	**Hpns**	**Hlpf**	**Int**
count	19,543	19,543	24,158	1,108	1,108	1,108
mean	33.82	30.91	1.59	4.28	0.36	34.58
std	57.30	56.42	2.31	0.44	0.17	40.67
min	0.01	0	1	1.71	0	15
25%	9.69	7.99	1	4	0.24	18
50%	18.12	15.85	1	4.31	0.34	23
75%	35	30.99	1	4.61	0.46	35
max	999.99	999.99	50	5	1.00	525
Clusters	**#Items**	**#Items**	**#Items**	**#Users**	**#Users**	**#Users**
0	4,893	3,870	22,234	264	277	326
1	4,880	4,808	729	844	277	246
2	4,911	5,213	461		277	262
3	4,859	5,652	734		277	274

Table 5 Statistics about the user and item clustering methods for `Amazon Video Games`, notation as in Table 4.

	Items Clusterings			Users Clusterings		
Statistics	**Price (TS)**	**Price (M)**	**Pop**	**Hpns**	**Hlpf**	**Int**
count	8,297	8,297	8,892	479	479	479
mean	56.28	40.89	2.29	3.93	0.51	42.52
std	85.66	67.98	2.92	0.57	0.18	65.67
min	0.01	0	1	1.67	0.04	15
25%	14.99	9.99	1	3.62	0.38	19
50%	28.99	19.99	1	4.01	0.51	25
75%	59.99	39.99	2	4.3	0.63	43
max	999.99	999.99	45	5	0.98	785
Clusters	**#Items**	**#Items**	**#Items**	**#Users**	**#Users**	**#Users**
0	2,075	1,411	6,812	231	120	146
1	2,076	2,182	736	248	120	99
2	2,143	1,829	684		119	116
3	2,003	2,875	660		120	118

same functions as described before for ItemKNN). The computed neighbors are then used to predict a score for each user-item pair.

- `SVD++` [56, 57], an algorithm that takes advantage of a simple latent factors model (trained through the stochastic gradient descent method), and it models and computes user and item biases. SVD++ also considers implicit feedback to improve learning.
- `BPRMF` (Bayesian personalized ranking - Matrix Factorization) [69, 57], a matrix factorization algorithm that exploits the Bayesian Personalized Ranking criterion [69] to minimize the ranking errors.

Table 6 Statistics about the user and item clustering methods for `Amazon Electronics`, notation as in Table 4.

	Items Clusterings			Users Clusterings		
Statistics	**Price (TS)**	**Price (M)**	**Pop**	**Hpns**	**Hlpf**	**Int**
count	47,660	47,660	56,727	5,351	5,351	5,351
mean	71.92	61.20	2.90	4.21	0.40	30.72
std	124.10	118.68	6.36	0.46	0.16	25.57
min	0.01	0.01	1	1.53	0	15
25%	10.06	9.95	1	3.96	0.28	18
50%	24.99	19.99	1	4.27	0.39	23
75%	71	51.91	2	4.54	0.51	34
max	999.99	999.99	275	5	1.00	500
Clusters	**#Items**	**#Items**	**#Items**	**#Users**	**#Users**	**#Users**
0	11,915	11,058	43,674	1,434	1,338	1,607
1	11,940	10,042	3,743	3,917	1,338	1,230
2	11,893	11,562	4,345		1,337	1,245
3	11,912	14,998	4,965		1,338	1,269

Table 7 Statistics about the user and item clustering methods for `MovieLens-1M`, notation as in Table 4. In the right we specify the available statistics for the categorical attributes.

	Items Clusterings		Users Clusterings	
Statistics	**Pop**	**Year**	**Hpns**	**Int**
count	3,667	3,883	6,040	6,040
mean	218.21	1986.07	3.72	132.48
std	328.97	16.90	0.43	154.19
min	1	1919	1	16
25%	25	1982	3.47	35
50%	92	1994	3.76	77
75%	273	1997	4.02	166
max	3,157	2000	5	1,851
Clusters	**#Items**	**#Items**	**#Users**	**#Users**
0	941	980	4,378	1,522
1	898	1,125	1,662	1,516
2	913	1,002		1,499
3	915	776		1,503

Users Categorical Clusterings			
Cluster	**Age**	**Cluster**	**Gender**
≤ 18	1,325	**Female**	1,709
$> 18 \wedge \leq 25$	2,096	**Male**	4,331
$> 25 \wedge \leq 35$	1,193		
> 35	1,426		

– `BPRSlim` (Bayesian personalized ranking - SLIM) [64], an algorithm that produces recommendations using a sparse aggregation coefficient matrix trained with Sparse Linear method (SLIM), trained to maximize the `BPR` criterion.

These recommenders, as we shall analyze in the next sections, may produce some biased or unfair recommendations. We now briefly discuss some starting hypotheses about these algorithms regarding their sensitivity to the different attributes considered. First, regarding the Random method, it may replicate any inherent bias already present in the user or item data, such as one class being over-represented, although the recommendations are generated without exploiting any of the attributes, so this effect might be reduced. Second, MostPopular would show a bias towards more popular items and, as a consequence, to any characteristics shared by those items (such as price); it may also characterize better those users that are more satisfied with *pop-*

Table 8 Tuned hyper-parameters for each of the tested recommendation methods.

	Number of Neighbors	Similarity Function
ItemKNN	[50,60,70,90,100]	[Jaccard Coefficient,Binary Cosine, Cosine, Pearson Correlation]
UserKNN	[50,60,70,90,100]	[Jaccard Coefficient,Binary Cosine, Cosine, Pearson Correlation]

	Number of Latent Factors	Learning Rate	Iterations
SVD++	[10,20,30,50,100,150]	[0.0005,0.005,0.05]	[30]
BPRMF	[10,20,30,50,100,150]	[0.005,0.05,0.5]	[30]
BPRSlim		[0.0005, 0.005, 0.05, 0.5]	[5,10,20,30]

ular recommendations. Third, ItemKNN and UserKNN exploit item-item and user-user similarities based on interaction data, hence they are not expected to promote a particular type of user or item, unless those are already over-represented in the input recommendation data; however, it is true that researchers have exposed that, depending on their parameters, these algorithms might behave as slightly personalized versions of the MostPopular algorithm, hence replicating the same biased/unfair suggestions [19,51,22]. A similar situation occur with the other recommenders, SVD++, BPRMF, and BPRSlim, since they only exploit the user-item interaction matrix but, depending on their hyper-parameters they might generate recommendations tailored towards popularity (mostly, when these algorithms are optimized with respect to accuracy), since these are expected to satisfy most users in the system.

For all these recommenders, we have performed a grid search to tune the parameters. We consider the range of values as suggested by the authors or by varying the values of the parameters around the ones shown in the original papers as the best performing ones; a summary of the considered values is shown in Table 8. Since a fixed timestamp splitting simulates a realistic online scenario [28,13], k-fold cross-validation would have been not applicable. Therefore, we have trained the models with each considered combination of parameters relying only on training set data. we have measured, for all the resulting models — by considering each combination as an independent model — the accuracy and the fairness metrics. Lastly, for the sake of clarity, we have reported in the paper the variants that maximize the nDCG metric at cutoff 10. The optimal values are reported in Table 9.

4.5 Evaluation metrics

In our experiments, we compute accuracy metrics as it is standard in the literature [43]. The *top-N* recommendation accuracy metrics we have used are Precision ($P@N$), Recall ($R@N$), and nDCG ($nDCG@N$), all at a cutoff N, which means that we only consider *top-N* items within each recommendation list.

Precision is defined as the proportion of recommended items that are relevant to the user:

$$Precision_u@N = \frac{|Rec_u^N \cap TS_u^+|}{N}$$

where Rec_u^N is the recommendation list up to the N-th element and TS_u^+ is the set of relevant test items for user u. Precision measures the system's ability to reject any

Table 9 Optimal hyper-parameters according to nDCG@10.

ItemKNN	**Amazon Toys & Games**	**Amazon Videogames**	**Amazon Electronics**	**MovieLens-1M**
Number of Neighbors	50	50	50	70
Similarity Function	Jaccard Coefficient	Jaccard Coefficient	Jaccard Coefficient	Pearson
UserKNN	**Amazon Toys & Games**	**Amazon Videogames**	**Amazon Electronics**	**MovieLens-1M**
Number of Neighbors	90	50	100	90
Similarity Function	Cosine	Jaccard Coefficient	Jaccard Coefficient	Cosine
SVD++	**Amazon Toys & Games**	**Amazon Videogames**	**Amazon Electronics**	**MovieLens-1M**
Number of Latent Factors	150	100	100	50
Learning Rate	0.0005	0.005	0.0005	0.005
Iterations	30	30	30	30
BPRMF	**Amazon Toys & Games**	**Amazon Videogames**	**Amazon Electronics**	**MovieLens-1M**
Number of Latent Factors	10	10	150	100
Learning Rate	0.5	0.5	0.5	0.005
Iterations	30	30	30	30
BPRSlim	**Amazon Toys & Games**	**Amazon Videogames**	**Amazon Electronics**	**MovieLens-1M**
Learning Rate	5	0.05	0.05	0.0005
Iterations	30	30	30	30

non-relevant documents in the recommended set. Recall, on the other hand, is defined as the proportion of relevant items that are actually recommended:

$$Recall_u@N = \frac{|Rec_u^N \cap TS_u^+|}{|TS_u^+|}$$

Hence, recall measures the system's ability to find all the relevant documents. Since these two metrics do not pay attention to whether a relevant item was recommended near the top or closer to the cutoff, in IR metrics that explicitly assign a gain to each ranking position are usually considered. The discounted cumulative gain (DCG) is a metric of ranking quality that measures the usefulness of a document based on its position in the result list. Since recommendation results may vary in length depending on the user, it is not possible to compare performance among different users, so the cumulative gain at each position should be normalized across users. Hence, normalized discounted cumulative gain, or nDCG, is defined as:

$$nDCG_u@N = \frac{1}{IDCG@N} \sum_{k=1}^{N} \frac{2^{r_{uk}} - 1}{\log_2(1+k)}$$

where k is the position of an item in the recommendation list and $IDCG@N$ indicates the score obtained by an ideal ranking of the recommendation list Rec_u^N that contains only relevant items.

For comparison with the proposed GCE metric, we include two complementary baseline metrics based on the absolute deviation between the mean ratings of different groups as defined in [90]:

$$MAD(R^{(i)}, R^{(j)}) = \left| \frac{\sum R^{(i)}}{\left|R^{(i)}\right|} - \frac{\sum R^{(j)}}{\left|R^{(j)}\right|} \right| \tag{7}$$

where $R^{(i)}$ denotes the predicted ratings for all user-item combinations in group i and $\left|R^{(i)}\right|$ is its size. Larger values for MAD imply more considerable difference

between the groups, interpreted as unfairness. Therefore, this notion interprets unfairness, where it is seen as inequality. Given that our proposed GCE in user-fairness evaluation is based on nDCG, we adapt this definition to also compare average nDCG for each group. We refer to these two baselines as MAD-rating (or MADr) and MAD-ranking (or MADR). Finally, the reported MAD corresponds to the average MAD between all the pairwise combinations within the groups involved, i.e.,

$$MAD = \text{avg}_{i,j}(MAD(R^{(i)}, R^{(j)})) \tag{8}$$

When possible, the metrics are computed on a per-user basis, and the reported results are the average of these individual values. We have used the RankSys[6] framework and adopted the *threshold-based relevant items* condition [28]. The evaluation has been performed considering Top-5, Top-10, and Top-20 recommendations for all datasets and a threshold of 3 for a 1-5 rating scale for deeming items as relevant for all the four datasets.

5 Results

In this section, we discuss the results obtained in our experiments.

5.1 Analysis of item fairness results

We show in Table 10 a comparison of the item-based GCE using popularity as the item feature for the four tested datasets. Due to space constraints, we only show results for cutoff 10 and the nDCG performance metric, since performance at other cutoffs or based on Precision and Recall were similar. Please note that largest nDCG correspond to the most accurate system, and the highest GCE correspond to the most fair system (GCE is always negative).[7] The alternative baseline fairness metric MAD (both variations MADR and MARr) produce most fair results closer to zero, i.e., the smaller MAD the more fair is the model.

We observe that accuracy (defined through nDCG) and fairness (either defined as our proposed GCE metric or using the MAD metric as reference) do not usually match each other, in the sense that the best recommenders for one dimension are different to those for a different dimension; for instance, Random is usually the best recommender based on MADR and MADr, whereas BPRMF and UserKNN are the best in terms of nDCG.

[6] `http://ranksys.org/`

[7] Please note that in Section 3 we defined an unfairness metric ω, the one producing a non-negative value in which if $\omega(m,a) < \omega(m',a)$ we can conclude that model m is less unfair than model m' (or more fair). This would make our unfairness metric consistent with the literature, e.g., see [75] Section 2.3. "Axioms for measuring inequality" where the authors define *inequality* as a non-negative value. Our GCE metric reports values that are all negative, with the maximum occurring when GCE ≈ 0. Our proposed GCE metric can be seen as a fairness metric, while the absolute form $|GCE|$ represent unfairness (always positive). For simplicity in discussing the results, however, we keep reporting the raw values for $|GCE|$, considering the sign when saying larger or smaller.

Table 10 ItemGCE using popularity as feature on the four tested datasets. The fair probability distributions are defined as p_{f_i} so that $p_{f_i}(j) = 0.1$ when $j \neq i$ and 0.7 otherwise – except for p_{f_0} that denotes the uniform distribution – and each column denotes the value obtained by GCE when such probability distribution is used as p_f in Equation 3. In bold, highlighted the best values for each metric.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	**−7.97**	**−32.39**	−32.39	**−1.14**	−2.51	**0.002**	**0.000**
MostPopular	0.008	−688.97	−1,874.78	−1,874.78	−1,874.78	−110.06	0.029	0.751
ItemKNN	0.004	−445.87	−82.79	−1,778.92	−1,778.92	−71.16	0.021	0.000
UserKNN	0.014	−520.00	−4,074.78	−85.48	−85.24	−83.08	0.043	0.001
SVD++	0.012	−1,082.10	−2,944.10	−2,944.10	−2,944.10	−172.96	0.045	0.029
BPRMF	**0.018**	−1,299.14	−3,534.45	−3,534.45	−3,534.45	−207.68	0.012	0.011
BPRSlim	0.007	−9.05	−38.61	**−22.80**	−14.74	**−1.28**	0.005	0.003

(a) `Amazon Electronics`

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−15.98	−2.38	−44.25	−44.25	−44.25	**0.000**	**0.000**
MostPopular	0.001	−126.91	−345.97	−345.97	−345.97	−20.13	0.008	0.045
ItemKNN	0.002	−0.08	−1.52	−0.42	**−0.52**	**−0.33**	0.055	0.001
UserKNN	**0.004**	**−0.01**	−0.81	**−0.38**	−0.54	−0.53	0.028	0.001
SVD++	0.003	−305.36	−831.35	−831.35	−831.35	−48.68	0.006	0.004
BPRMF	0.002	−146.48	−24.61	−586.48	−586.48	−23.30	0.010	0.008
BPRSlim	0.003	−0.12	**−0.12**	−1.40	−1.04	−0.62	0.011	0.036

(b) `Amazon Toys & Games`

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−11.85	**−1.87**	−48.40	−2.11	−48.40	**0.001**	**0.001**
MostPopular	0.004	−490.77	−1,335.68	−1,335.68	−1,335.68	−78.34	0.017	0.116
ItemKNN	0.013	−1.25	−3.68	−1.03	−7.72	−0.11	0.100	0.002
UserKNN	**0.019**	**−1.23**	−10.09	**−0.95**	**−1.13**	−0.18	0.084	0.002
SVD++	0.005	−499.25	−1,358.75	−1,358.75	−1,358.75	−79.70	0.023	0.017
BPRMF	0.008	−3.08	−2.18	−17.12	−8.14	−0.36	0.024	0.032
BPRSlim	0.011	−1.45	−3.18	−8.48	−2.45	**−0.11**	0.034	0.019

(c) `Amazon Video Games`

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.004	**−1.88**	−13.70	**−2.76**	**−1.15**	**−0.22**	**0.034**	**0.010**
MostPopular	0.081	−7,579.95	−20,618.25	−20,618.25	−20,618.25	−1,212.61	0.578	199.972
ItemKNN	0.095	−5.73	−40.18	−5.73	−3.16	−0.78	0.188	0.016
UserKNN	**0.107**	−6,584.00	−26,336.27	−26,336.27	−1,055.21	−1,053.29	0.271	0.040
SVD++	0.070	−5.84	**−7.20**	−38.98	−3.76	−0.79	0.420	0.387
BPRMF	0.094	−5,841.91	−23,366.07	−23,366.07	−940.19	−934.54	0.278	0.360
BPRSlim	0.097	−2,938.53	−23,031.61	−476.20	−472.90	−470.02	0.190	0.448

(d) `MovieLens-1M`

Under equality – i.e., p_{f_0} – UserKNN is the recommender system with highest values of GCE (the most fair) in `Amazon Toys & Games` and `Amazon Video Games`, whereas Random and BPRSlim are the most fair ones in `Amazon Electronics` and `MovieLens-1M`. As a validation of the proposed metric, by focusing on the row for the MostPopular recommender, we notice that it always obtains higher (better) values of GCE under p_{f_4}. This is the situation where recommending more popular items is deemed (more) fair by the system designer (they have a larger weight in the probability distribution).

Table 11 ItemGCE using price as feature on `Amazon Toys & Games`. Notation as in Table 10.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−11.95	−48.74	−1.84	−48.74	−2.29	**0.001**	0.000
MostPopular	0.001	−84.80	−339.23	−15.84	−339.23	−13.41	0.028	0.149
ItemKNN	0.002	−0.15	−1.88	−0.60	−0.80	−0.14	0.009	**0.000**
UserKNN	**0.004**	**−0.07**	**−0.43**	−1.17	−0.92	−0.22	0.025	0.002
SVD++	0.003	−203.82	−33.81	−815.83	−815.83	−32.47	0.022	0.017
BPRMF	0.002	−0.79	−2.67	−4.47	−1.52	**−0.03**	0.017	0.014
BPRSlim	0.003	−0.29	−0.57	**−0.34**	**−0.32**	−3.37	0.016	0.053

If we now focus on the two extreme non-uniform situations (either very long tail or very popular items, i.e., p_{f_1} or p_{f_4}, respectively), `Amazon Electronics` and `Amazon Video Games` show similar results, since BPRSlim has the largest values for popular items and Random for long-tail items; on the `Amazon Toys & Games` dataset, on the other hand, BPRSlim is the most fair regarding long-tail items and ItemKNN for popular items, even though BPRSlim also shows good values for popular items, consistent with the results found for the other datasets. `MovieLens-1M`, on the other hand, shows a different behavior: SVD++ provides more fair results in terms of long-tail items, whereas Random and ItemKNN show good values for popular items. These results do not match any of the previously discussed datasets, probably because the domains and rating elicitation process are very different (in Amazon, ratings are associated with a review).

Hence, we conclude that ItemKNN, BPRSlim, and UserKNN are prone to suggest more items from the head of the distribution, although this does not mean they do not recommend tail items, since the values obtained when less popular items are promoted through the fair distribution are not too small either – as it is the case of the MostPopular algorithm, which only recommends items from the fourth category of items, and thus the final GCE value gets distorted by the near-zero probability of the other categories –; SVD++, on the other hand, and BPRMF to a lesser extent (since this depends on the dataset), seems to be tailored to promote mostly popular items, producing similar values as those obtained by the MostPopular algorithm. These results agree with previous observations on the biases evidenced by different algorithms in several datasets [19,51,22], as discussed previously in Section 4.4. Moreover, if we look at the results through the lens of which models promote recommendation of long tail items, we can see that BPRSlim and Random are the most capable methods.

For the sake of space, from now on we focus our attention on the analysis of the `Amazon Toys & Games` dataset, the rest are shown in Appendix B. Hence, Table 11 shows the item-based GCE values obtained using price as the item feature. In this case, and in contrast to the scenario where popularity is used as the item feature, the MostPopular recommender does not show a distinctive pattern, since it obtains higher values for p_{f_2} and p_{f_4}; this is probably due to the inherent biases in the data, indicating that popular items tend to appear in the low-to-medium and high price clusters. These patterns in the data are also evident when checking the results for Random, where the same two clusters (p_{f_2} and p_{f_4}) produce the highest GCE values.

As with the popularity feature, UserKNN is the best method under equality constraints on the same dataset; however, the situation changes drastically when other

fair distributions are considered since the nature of the item features is very different. For instance, now, the method that produces more fair recommendations for more expensive items (p_{f_4}) is BPRMF, followed by ItemKNN. Regarding the least expensive items, UserKNN performs the best, followed by BPRSlim, which also obtains good performance values in the two intermediate clusters. These results provide interesting insights into the performance of these models. We can observe that nDCG produces results for most recommendation models that are too close - without providing any transparent/distinguishable difference; under the proposed GCE metric, we obtain a more transparent/detailed understanding of how these models behave with respect to the promotion of more expensive items against cheap items. This capability might be important for the system designer in order to know which models to choose for different e-commerce settings.

Based on this, we conclude that there are some recommendation techniques more prone to recommend expensive or cheap products – a highly sensitive item attribute, even when this information is not included in the training data of the algorithms. Thus, we observe that ItemKNN, UserKNN, and BPRMF tend to include more expensive items in their recommendations (also considering the results for the other domains as presented in Table 14). However, as discussed before, some of these results might be attributed to the inherent biases of some algorithms to produce more popular items. This effect is, indeed, not negligible, although it depends strongly on the domain since the relation between popularity and price may change from domain to domain: the MostPopular algorithm obtains very good values of ItemGCE in `Amazon Electronics` for all distributions except when the most expensive items are promoted, a similar situation is found in `Amazon Video Games`, although the obtained values are larger, whereas the optimal cases in `Amazon Toys & Games` are found for p_{f_2} and p_{f_4}.

5.2 Analysis of user fairness results

In this section, we analyze the user variation of the GCE metric. For this, in Table 12 we show the results of the three tested user features (happiness, helpfulness, and interactions) on the `Amazon Toys & Games` dataset; results for the other datasets are included in the appendix.

The first thing we notice when considering equality as fairness is that the values are much smaller than in the case of item features, even reaching an optimal of 0, for BPRMF in the happiness feature, although the other optimal recommenders in the other datasets for this scenario also obtain values very close to the optimal one.

Let us now analyze the other scenarios, where fairness is not equivalent to equality. In this case, there is no recommender that obtains a perfect value, although again happiness seems to be the easiest feature where something similar to perfect fairness might be achieved, since BPRSlim shows a -0.02 value for p_{f_2}. ItemKNN and BPRSlim tend to obtain good performance values for the three features, in particular ItemKNN is the best recommender for the users with more interactions, together with those users with least helpful reviews; BPRSlim on the other hand is the best one for the users with least interactions and for the happiest users.

Table 12 UserGCE for `Amazon Toys & Games` dataset using the three user features considered. Notation as in Table 10, except for the Happiness attribute, where $p_{f_i}(j) = 0.1$ when $j \neq i$ and 0.9 otherwise when used as p_f in Equation 3.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	MADR	MADr
Random	0.000	−4.05	−13.83	−0.09	0.000	**0.000**
MostPopular	0.001	−0.01	−0.44	−0.23	0.000	0.121
ItemKNN	0.002	−0.25	−1.44	−0.04	0.002	0.008
UserKNN	**0.004**	−0.20	−1.23	−0.05	0.003	0.049
SVD++	0.003	−0.01	−0.43	−0.23	0.001	0.018
BPRMF	0.002	**0.00**	**−0.33**	−0.31	**0.000**	0.104
BPRSlim	0.003	−2.01	−7.20	**−0.02**	0.003	0.939

(a) Happiness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−6.24	−0.96	−25.93	−1.25	−25.93	**0.000**	**0.000**
MostPopular	0.001	−0.26	−2.27	−0.14	−1.57	−0.36	0.001	0.118
ItemKNN	0.002	**−0.01**	**−0.45**	−0.54	**−0.44**	−0.84	0.001	0.013
UserKNN	**0.004**	−0.10	−0.91	**−0.14**	−0.60	−1.32	0.003	0.031
SVD++	0.003	−0.08	−1.24	−0.23	−0.94	−0.39	0.002	0.002
BPRMF	0.002	−0.28	−0.94	−0.22	−3.01	**−0.33**	0.002	0.081
BPRSlim	0.003	−0.02	−0.72	−0.44	−0.78	−0.36	0.001	3.145

(b) Helpfulness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−9.54	−26.73	−26.73	−26.73	−1.35	**0.000**	**0.000**
MostPopular	0.001	−19.60	−154.54	−4.09	−3.32	−3.28	0.001	0.186
ItemKNN	0.002	−0.05	−0.58	−0.70	−1.03	**−0.22**	0.001	0.025
UserKNN	**0.004**	**−0.02**	−0.84	−0.55	−0.63	−0.31	0.001	0.038
SVD++	0.003	−0.03	−0.70	−0.72	−0.71	−0.25	0.001	0.003
BPRMF	0.002	−0.02	−0.58	**−0.46**	−0.36	−0.94	0.001	0.208
BPRSlim	0.003	−0.04	**−0.49**	−0.64	**−0.28**	−1.07	0.001	9.001

(c) Interactions

In summary, we conclude that ItemKNN is inherently tailored to users with many interactions and less helpful users, whereas BPRSlim seems to provide better and fair recommendations for happy and cold (few interactions) users. Interestingly, although both of these models leverage item-item similarities based on user interactions, we can observe that, under GCE, we find contrasting results with respect to their performance in the study of item and user fairness.

Regarding the most sensitive attributes (i.e., helpfulness in these results, together with age and gender as shown in the Appendix), we conclude that helpfulness is a dimension that is not too discriminated against by the algorithms, since its GCE values in all datasets and under any distribution function is low. On the other hand, age and gender (only reported for `MovieLens-1M` as explained before for privacy concerns) as reported in Table 17 present much more variation; in particular, the most popular algorithm provides better recommendations to younger users, whereas Random, ItemKNN, and BPRSlim produce good recommendations to users on the other end of the spectrum. Additionally, it is very interesting to observe that under equality (i.e., p_{f_0}), almost every algorithm obtains fair recommendations with respect to the gender attribute; this is not true, however, when fairness is defined as

Table 13 Spearman correlation value between recommenders ranked based on GCE values for p_{f_0} and the indicated fair distribution p_f for Amazon datasets and user and item attributes.

Attribute	p_f	Amazon Electronics	Amazon Toys & Games	Amazon Video Games
Price	p_{f_1}	0.10	0.85	0.85
	p_{f_2}	0.28	0.70	0.74
	p_{f_3}	0.17	0.78	0.52
	p_{f_4}	0.73	0.63	0.83
Popularity	p_{f_1}	0.77	0.84	0.91
	p_{f_2}	0.95	0.93	0.78
	p_{f_3}	0.93	0.93	0.84
	p_{f_4}	0.99	0.92	0.94
Happiness	p_{f_1}	1.00	0.63	0.59
	p_{f_2}	−1.00	−0.29	−0.22
Helpfulness	p_{f_1}	0.10	0.74	0.50
	p_{f_2}	0.25	0.35	0.11
	p_{f_3}	0.36	0.64	0.77
	p_{f_4}	0.58	0.43	0.52
Interactions	p_{f_1}	0.66	0.88	0.41
	p_{f_2}	0.55	0.73	0.39
	p_{f_3}	0.46	0.68	0.37
	p_{f_4}	−0.21	0.29	0.21

promoting one of the two considered genders. In particular, females (p_{f_1}) obtain better results with Random, ItemKNN, and BPRSlim, probably because they are under-represented, whereas males receive good enough recommendations simply by exploiting the most popular algorithm, evidencing that the tastes of the majority of the population matches those of the over-represented attribute value.

5.3 Discussion

When analyzing the presented approach and reported results from a global point of view, we can finally answer the three research questions posed at the beginning of the paper.

RQ1. How to define a fairness evaluation metric that considers different notions of fairness (not only equality)? We have presented a novel metric that seamlessly works with either user or item features while, at the same time, it is sensitive to different notions of fairness (through the definition of a specific fair distribution): either based on equality (by using a uniform distribution) or favoring some of the attribute values (such as most expensive items or less happy users). This is a critical difference with respect to other metrics proposed in the literature to measure fairness, which should be tailored to either users or items or that implicitly assume equality as fairness (see Section 2). In our experiments, this becomes obvious when comparing the results found for the proposed GCE against those found for MAD-based metrics, since the optimal recommender in the latter case is usually Random, mostly because this type of algorithm is unbiased by definition. However, the proposed GCE metric allows capturing other concepts typically considered when evaluating recommender systems such as relevance and ranking.

RQ2. How do classical recommendation models behave in terms of such an evaluation metric, especially under non-equality definitions of fairness? We summarize the results obtained as follows. Recommendation algorithms based on neighbors performs well in general: whereas UserKNN performs well (considering producing fair recommendation) under equality for item attributes, ItemKNN (together with BPRMF) perform well either under equality or non-equality constraints. Additionally, BPRSlim produces fair results under extreme scenarios of fairness (i.e., p_{f_1} or p_{f_4}), again for item attributes. These conclusions also apply, to some extent, to the results not discussed so far, which are shown in the appendix. It should be considered that the presented results correspond to the values obtained when optimizing for accuracy (the recommenders were selected according to their nDCG@5 values), hence, a slightly different behavior could have been obtained if each metric was optimized independently. We do not include these results because we are more interested in analyzing how state-of-the-art algorithms (typically selected and assessed with respect to accuracy metrics) behave with respect to fairness oriented metrics.

RQ3. Which user and item attributes are more sensitive to different notions of fairness? Which attribute/recommendation-algorithm combination is more prone to produce fair/unfair recommendations? We assume this can be understood as those cases where results for equality differ too much from results for non-equality. To properly analyze this issue, we compare the rankings obtained for all the tested recommenders (not only the 7 presented which correspond to those with optimal parameters, but the 95 combinations for all parameters) and compute Spearman correlation between the results using the distribution under equality constraints and the other cases (see Table 13). We observe that the item popularity is more or less stable, whereas the item price depends heavily on the dataset; on the other hand, the user attributes (helpfulness, interactions, and especially happiness) are the least stable, since their correlations are the lowest ones. This evidences that user attributes are more sensitive to different notions of fairness, since the performance of recommenders change more drastically when equality and non-equality distributions are used.

6 Limitations

Although the experimental evaluation shows the effectiveness of the proposed fairness evaluation system, there are some limitations we highlight and discuss in the following. The aim of this section is to shed light on these shortcomings and what we deem important for future extension. We further discuss our proposals for future works on Section 7.

- *Granularity of attributes:* it is not obvious how different granularities of the protected attributes (finer or larger) may impact on the proposed metric, or even if more than one attribute wants to be considered at the same time, for instance, by combining multiple attributes or exploring how some attributes impact on others. However, we argue that this is a potential issue that many fairness-aware metrics would be sensitive to, since all of them consider – to some extent – the range of the attributes, either as raw values or by comparing their frequencies or probability distributions (as we do here).

- *Choice of recommendation models:* The main recommendation models considered in this work were different variation of CF models, namely ItemKNN, UserKNN, SVD++, BPRMF, and BPRSlim. Hence, all of these techniques exploit, in some way or another, the similarity of interactions performed by the users. It would have been interesting to consider the performance in terms of fairness of approaches based on content (or hybrid models). Modern recommendation models utilize a wealth of side-information beyond the user-item matrix such as as social connections [15], multimedia content [38,39] as well as contextual data [11] to build more domain-dependent recommendations models. In particular, it may be interesting to analyze the impact and sensibility of sensitive attributes-based recommendation strategies on fairness evaluation metrics. As an example, it could be mentioned the demographic recommenders, that take the age or gender into account when showing the recommendations. Moreover, it would be interesting to evaluate also other recommender systems families, like Graph-based [81] and Neural Network-based recommenders [47].
- *Connection with constrained-based recommendation:* Given the flexibility of the presented framework to measure a non-uniform distribution of resources among members of protected groups — defined by sensitive features —, we believe it would have been useful if the problem formulation of the framework could incorporate constraints factors, for example capacity constraints, time constraints, space/size constraints, and so forth (see e.g., [30] for good pointers to the topic). These are the situations in which we may want to distribute recommendations benefits in a non-uniform manner.
- *Evaluation of fairness for user-item categories:* In this work, we have analyzed the fairness by considering user or item attributes. However, another interesting research path is to consider user and item attributes jointly. In this respect, we may represent the joint distribution of users and items in the clustering via a matrix (or a tensor) in which each axis represents a specific clustering. This challenging idea paves the way to a different perspective on fairness. In this sense, while the idea of combining user and item attributes in fairness is not novel, to the best of our knowledge, it could be the first attempt to analyze fairness inequality considering both users and items.
- *Parameter selection:* As discussed in Section 3, the GCE metric for fairness evaluation has some parameters that need to be set by domain experts. The fair distribution p_f is one of these parameters that may be difficult to obtain without comprehensive research.

7 Conclusions and the road ahead

In this paper, we proposed a flexible, probabilistic framework for evaluating user and item fairness in recommender systems. We conducted extensive experiments on real-world datasets and demonstrated the flexibility of the proposed solution in various recommendation settings. In summary, our framework can evaluate fairness beyond equality, can evaluate both user fairness and item fairness, and is designed based on theoretically sound foundations, which makes it interpretable. In the preliminary

version of this work presented at the RMSE workshop at the ACM RecSys 2019 conference [37], we analyzed the results from the conducted experiments by winning participants using our proposed fairness evaluation metric. We realized that an evaluation based on the item fairness as defined in the RecSys Challenge 2017 [9] —that is according to the types of users (regular vs. premium)— captures additional nuances about the different submissions.[8] For instance, the proposed winner system produces balanced recommendations across the two membership classes. This is in contrast to our expectation that premium users should be provided with more favorable recommendations (under a scenario where there is a cost in the item supply).

On the other hand, when exploiting user attributes in a classical recommendation task to evaluate user fairness, we observed interesting insights related to the different recommendation algorithms. So far, we have studied the case where users are clustered according to their activity in the system (interaction attribute), but also according to more sensitive attributes such as age and gender. In both cases, we have found that algorithms with very similar performance values obtain very different values of user fairness, mostly because the recommendation methods behave strikingly different at each user cluster, hence validating the expected behavior of the proposed metric. Additionally, we compare our proposed metric against baseline metrics defined in the literature (such as MAD [90]), which have been extended to be also suitable for ranking scenarios; it becomes evident that these metrics, cannot incorporate other definitions of fairness in its computation; hence their flexibility is very limited.

In summary, our framework is especially useful when there are some qualities that the system designer wants to discriminate among the users, either based on merits, their needs or in a general case of free/premium users. We can mention other examples under this general case, consider as an example mobile v.s. PC users, probably particularized to a specific algorithm; in this case for instance we expect that a contextualized method performs better when the user is moving. The same holds for new v.s. old users where we expect a non-personalized algorithm should work better for old users when there is no known history and so forth.

In the future, we aim to extend this work along the following dimensions. First, in this work, we presented a principled way to derive an evaluation measure for fairness objectives. The evaluation framework presented in this work is learning-model agnostic, which means it is not validated for building an actual fairness-aware system. Rather the focus was to **measure** fairness of RS based on different user and item attributes. A natural extension would be to use this metric in the learning step of recommendation models, e.g., by optimizing the model parameters with respect to the proposed fairness metric. Second, we plan to simultaneously incorporate user and item fairness into the generalized cross-entropy computation, in order to evaluate both multiple objectives in a single framework. Third, another natural extension of our proposed fairness evaluation framework is to utilize it for scenarios where the system designer has to take into account multiple sensitive attributes (e.g., gender and race) simultaneously as a fairness criterion. As a first approach, this could be achieved by constructing all possible combinations of the sensitive attribute values (e.g., black man, black woman, white man and white woman) and measure how

[8] In this challenge, the users correspond to the items being recommended.

fair recommendations are for each individual combination separately [84]. Moreover, it this study we mainly studied the trade-off between accuracy and fairness metrics, however, recommendation evaluation consists of many other aspects, including diversity. Exploring the connections between these metrics and recommendation fairness evaluation would be an interesting future direction. Last but not least, conducting user studies to understand the correlation between user satisfaction and fairness computed using GCE is an interesting future direction that we would like to pursue.

Appendix A: Theoretical analysis of GCE properties

In this appendix, we provide a theoretical analysis of the proposed probabilistic metric, i.e., GCE, for measuring unfairness. Previous work [75] has explored four properties to be satisfied by inequality indices, including unfairness inequality. These properties are (1) anonymity, (2) population invariance, (3) transfer principle, and (4) zero normalization.

We claim that GCE satisfies these four properties. In this appendix, for the sake of clarity, we prove that the mentioned properties are satisfied by a simplified version of the proposed probabilistic unfairness metric, i.e., GCE when p_f is uniform. We follow our proofs based on the GCE formulation for discrete attributes, presented in Equation (3). Assuming p_f being uniform, the GCE formulation is:

$$\begin{aligned} I_{\text{uniform}}(m,a) &= \frac{1}{\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{1}{n}\right)^{\beta}\cdot p_m^{(1-\beta)}(a_j)-1\right] \\ &= \frac{1}{\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{1}{n}\right)^{\beta}\cdot\left(\frac{v_j}{Z}\right)^{(1-\beta)}-1\right] \\ &= \frac{1}{n\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{v_j}{\mu}\right)^{(1-\beta)}-n\right] \end{aligned} \tag{9}$$

where $p_m(a_j) = v_j/Z$, i.e., $Z = \sum_{j=1}^{n} v_j$, and $\mu = Z/n$ denotes the average value. For brevity, we denote $I_{\text{uniform}}(m,a)$ as $I_{\text{uniform}}(\mathbf{v})$ where $\mathbf{v} = [v_1, v_2, \cdots, v_n] \in \mathbb{R}^n$ is the vector of all values corresponding to the attribute a obtained by the model m.

Anonymity. According to the anonymity property, the inequality measure should not depend on the characteristics of attributes except for their values obtained by the model. As shown in Equation (9), the inequality measure only depends on the value of attributes, i.e., v_js and the average value μ which again is computed based on the values as $\frac{\sum_{j=1}^{n} v_j}{n}$. Therefore, this property is satisfied by I_{uniform}.

Population invariance. This property indicates that the inequality measure is independent of the population size.

Proof. To prove that I_{uniform} satisfies the population invariance property, assume $\mathbf{v}' =< \mathbf{v}, \mathbf{v}, \cdots, \mathbf{v} >\in \mathbb{R}^{nk}$ denotes a k-replication of the vector $\mathbf{v}$. Therefore, $I_{\text{uniform}}(\mathbf{v}')$ is computed as:

$$\begin{aligned} I_{\text{uniform}}(\mathbf{v}') &= \frac{1}{nk\beta\cdot(1-\beta)}\left[\sum_{j=1}^{nk}\left(\frac{v'_j}{\mu}\right)^{(1-\beta)}-nk\right] \\ &= \frac{1}{nk\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(k\frac{v_j}{\mu}\right)^{(1-\beta)}-nk\right] \\ &= \frac{1}{n\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{v_j}{\mu}\right)^{(1-\beta)}-n\right] \\ &= I_{\text{uniform}}(\mathbf{v}) \end{aligned}$$

□

The transfer principle. According to the transfer principle, also known as the Pigou-Dalton principle [35,67], transferring benefit from a high-benefit attribute value to a low-benefit value, if it does not reverse the relative position of values, must decrease the inequality.

Proof. Assume we transfer δ from v_j to $v_{j'}$, such that $v_j > v_{j'}$ and $0 < \delta < \frac{v_j - v_{j'}}{2}$ so this transfer does not reverse the relative position of these two attribute values. This results in $\mathbf{v}' = [v_1, v_2, \cdots, v_j - \delta, \cdots, v_{j'} + \delta, \cdots, v_n] \in \mathbb{R}^n$. Therefore,

$$\begin{aligned}
I_{\text{uniform}}(\mathbf{v}') - I_{\text{uniform}}(\mathbf{v}) &= \frac{1}{n\beta \cdot (1-\beta)} \left[\sum_{j=1}^{n} \left(\frac{v'_j}{\mu} \right)^{(1-\beta)} - n \right] - \frac{1}{n\beta \cdot (1-\beta)} \left[\sum_{j=1}^{n} \left(\frac{v_j}{\mu} \right)^{(1-\beta)} - n \right] \\
&= \frac{1}{n\beta \cdot (1-\beta)} \left[\sum_{j=1}^{n} \left(\left(\frac{v'_j}{\mu} \right)^{(1-\beta)} - \left(\frac{v_j}{\mu} \right)^{(1-\beta)} \right) \right] \\
&= \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[(v_j - \delta)^{(1-\beta)} + (v_{j'} + \delta)^{(1-\beta)} - v_j^{(1-\beta)} - v_{j'}^{(1-\beta)} \right] \quad (10)
\end{aligned}$$

To obtain the maximum value of this function, we compute its derivative with respect to δ and set it to zero, as follows:

$$\begin{aligned}
&\frac{\partial (I_{\text{uniform}}(\mathbf{v}') - I_{\text{uniform}}(\mathbf{v}))}{\partial \delta} = 0 \\
\Rightarrow &\frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[-(1-\beta)(v_j - \delta)^{-\beta} + (1-\beta)(v_{j'} + \delta)^{-\beta} \right] = 0 \\
\Rightarrow &-(v_j - \delta)^{-\beta} + (v_{j'} + \delta)^{-\beta} = 0 \\
\Rightarrow &\delta = \frac{v_j - v_{j'}}{2}
\end{aligned}$$

Since $\frac{\partial^2 (I_{\text{uniform}}(\mathbf{v}') - I_{\text{uniform}}(\mathbf{v}))}{\partial \delta^2} < 0$, the computed δ gives us the maximum value for the given function. Therefore, according to Equation (10), since $0 < \delta < \frac{v_j - v_{j'}}{2}$, we have:

$$\begin{aligned}
I_{\text{uniform}}(\mathbf{v}') - I_{\text{uniform}}(\mathbf{v}) &< \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[\left(v_j - \frac{v_j - v_{j'}}{2} \right)^{(1-\beta)} + \left(v_{j'} + \frac{v_j - v_{j'}}{2} \right)^{(1-\beta)} - v_j^{(1-\beta)} - v_{j'}^{(1-\beta)} \right] \\
&= \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[\left(\frac{v_j + v_{j'}}{2} \right)^{(1-\beta)} + \left(\frac{v_j + v_{j'}}{2} \right)^{(1-\beta)} - v_j^{(1-\beta)} - v_{j'}^{(1-\beta)} \right] \\
&= \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[2^{\beta} (v_j + v_{j'})^{(1-\beta)} - v_j^{(1-\beta)} - v_{j'}^{(1-\beta)} \right] \\
&< \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[2^{\beta} (2v_{j'})^{(1-\beta)} - v_j^{(1-\beta)} - v_{j'}^{(1-\beta)} \right] \\
&= \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[2(v_{j'})^{(1-\beta)} - v_j^{(1-\beta)} - v_{j'}^{(1-\beta)} \right] \\
&= \frac{1}{n\beta \cdot (1-\beta) \cdot \mu^{(1-\beta)}} \left[(v_{j'})^{(1-\beta)} - v_j^{(1-\beta)} \right] \\
&< 0
\end{aligned}$$

Therefore, $I_{\text{uniform}}(\mathbf{v}') < I_{\text{uniform}}(\mathbf{v})$, and thus I_{uniform} satisfies the transfer principle. □

Zero normalization. According to this property, the inequality measure should be minimized when all attribute values are equal (i.e., the uniform distribution). The minimum value for the fairness metric should be zero.

Proof. To prove this property, we use the Lagrange multiplier approach. The Lagrange function is defined as:

$$\mathscr{L}(\mathbf{v},\lambda) = \frac{1}{n\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{v_j}{\mu}\right)^{(1-\beta)} - n\right] - \lambda\left(\sum_{j=1}^{n}\frac{v_j}{n} - \mu\right) \tag{11}$$

where λ is the Lagrange multiplier. Therefore, we have:

$$\begin{cases} \frac{\partial\mathscr{L}(\mathbf{v},\lambda)}{\partial v_j} = \frac{1}{n\beta\mu}\cdot\left(\frac{v_j}{\mu}\right)^{-\beta} - \frac{\lambda}{n} \\ \frac{\partial\mathscr{L}(\mathbf{v},\lambda)}{\partial\lambda} = \sum_{j=1}^{n}\frac{v_j}{n} - \mu \end{cases} \tag{12}$$

Setting the above partial derivatives to zero results in $v_1 = v_2 = \cdots = v_n = \mu$. Therefore, we have:

$$\begin{aligned} &\min_{\mathbf{v}} \frac{1}{n\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{v_j}{\mu}\right)^{(1-\beta)} - n\right] \\ &= \frac{1}{n\beta\cdot(1-\beta)}\left[\sum_{j=1}^{n}\left(\frac{\mu}{\mu}\right)^{(1-\beta)} - n\right] \\ &= \frac{1}{n\beta\cdot(1-\beta)}[n-n] \\ &= 0 \end{aligned}$$

Therefore, I_{uniform} satisfies the zero normalization property. □

Summary . In this appendix, we theoretically study GCE and the provided proofs show that GCE satisfies the anonymity, population invariance, transfer principle, and zero normalization properties, under the uniformity assumption for the fair distribution. The proofs can be extended to the general case by relaxing the uniformity assumption, since we do not use any property of the uniform distribution in the proofs and just use its simple form to improve the readability and clarity.

Appendix B: Full results

In this section we present the results for all the datasets and item and user attributes that were not included in the paper because of space constraints. First, we show in Table 14 the item GCE based on the price attribute for the datasets (instead of only limited to toys, as in Section 5.1).

Second, our results on user attributes – that is, interactions, helpfulness, and happiness for Amazon datasets, and age and gender for MovieLens – is presented for the datasets (together with the analysis already shown in Section 5.2 for `Amazon Toys & Games`): `Amazon Electronics` is described in Table 15, `Amazon Video Games` in Table 16, and `MovieLens-1M` in Table 17.

Acknowledgements The authors thank the reviewers for their thoughtful comments and suggestions. This work was supported in part by the Ministerio de Ciencia, Innovación y Universidades (reference: PID2019-108965GB-I00), and in part by the Center for Intelligent Information Retrieval. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsors.

References

1. Title vii of the civil rights act of 1964. `https://www.eeoc.gov/laws/statutes/titlevii.cfm`. Accessed: 2019-07-31
2. Netflix TechBlog (2012 (accessed June 5, 2019)). `http://techblog.netflix.com/2012/04/netflix-recommendations-beyond-5-stars.html`

Table 14 ItemGCE using price as feature on the tested datasets. Notation as in Table 10.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−0.09	**−0.29**	−1.15	**−0.28**	−1.14	**0.000**	**0.000**
MostPopular	0.008	−0.12	−0.42	**−0.25**	−0.62	−1.82	0.056	1.330
ItemKNN	0.004	−0.03	−0.96	−0.29	−0.57	**−0.57**	0.004	0.000
UserKNN	0.014	**−0.02**	−0.52	−0.43	−0.42	−0.98	0.015	0.000
SVD++	0.012	−361.26	−58.07	−57.78	−62.81	−2,829.16	0.076	0.047
BPRMF	**0.018**	−1.47	−1.01	−0.66	−0.30	−12.42	0.019	0.020
BPRSlim	0.007	−0.09	−0.50	−0.36	−0.39	−1.62	0.001	0.000

(a) `Amazon Electronics`

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−11.95	−48.74	−1.84	−48.74	−2.29	**0.001**	0.000
MostPopular	0.001	−84.80	−339.23	−15.84	−339.23	−13.41	0.028	0.149
ItemKNN	0.002	−0.15	−1.88	−0.60	−0.80	−0.14	0.009	**0.000**
UserKNN	**0.004**	**−0.07**	**−0.43**	−1.17	−0.92	−0.22	0.025	0.002
SVD++	0.003	−203.82	−33.81	−815.83	−815.83	−32.47	0.022	0.017
BPRMF	0.002	−0.79	−2.67	−4.47	−1.52	**−0.03**	0.017	0.014
BPRSlim	0.003	−0.29	−0.57	**−0.34**	**−0.32**	−3.37	0.016	0.053

(b) `Amazon Toys & Games`

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−10.57	−43.16	−2.21	−1.60	−43.16	**0.002**	0.001
MostPopular	0.004	−164.68	−28.64	−27.11	−26.27	−1,290.25	0.023	0.152
ItemKNN	0.013	−0.17	−2.30	−0.52	−0.22	−0.56	0.004	**0.000**
UserKNN	**0.019**	−0.12	−1.80	−0.75	−0.33	**−0.28**	0.063	0.004
SVD++	0.005	−0.70	−3.22	−1.07	**−0.04**	−3.70	0.034	0.025
BPRMF	0.008	−0.26	−1.22	−0.71	−0.07	−2.32	0.028	0.030
BPRSlim	0.011	**−0.05**	**−1.11**	**−0.39**	−0.28	−0.83	0.010	0.005

(c) `Amazon Video Games`

3. Amazon product data (2014 (accessed July 31, 2019)). `http://jmcauley.ucsd.edu/data/amazon/`
4. Etsy.com-Shop for anything from creative people everywhere (2019 (accessed August 12, 2019)). `https://www.etsy.com`
5. What is fair when it comes to AI bias? (2020 (accessed April 11, 2020)). `https://www.strategy-business.com/article/What-is-fair-when-it-comes-to-AI-bias?gko=827c0`
6. Abdollahpouri, H., Adomavicius, G., Burke, R., Guy, I., Jannach, D., Kamishima, T., Krasnodebski, J., Pizzato, L.: Multistakeholder recommendation: Survey and research directions. User Modeling and User-Adapted Interaction **30**(1), 127–158 (2020)
7. Abdollahpouri, H., Burke, R., Mobasher, B.: Recommender systems as multistakeholder environments. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, UMAP 2017, Bratislava, Slovakia, July 09 - 12, 2017, pp. 347–348 (2017). DOI 10.1145/3079628.3079657. URL `https://doi.org/10.1145/3079628.3079657`
8. Abebe, R., Kleinberg, J.M., Parkes, D.C.: Fair division via social comparison. In: K. Larson, M. Winikoff, S. Das, E.H. Durfee (eds.) Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017, pp. 281–289. ACM (2017). URL `http://dl.acm.org/citation.cfm?id=3091171`
9. Abel, F., Deldjoo, Y., Elahi, M., Kohlsdorf, D.: Recsys challenge 2017: Offline and online evaluation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys '17, pp. 372–373. ACM, New York, NY, USA (2017). DOI 10.1145/3109859.3109954. URL `http://doi.acm.org/10.1145/3109859.3109954`
10. Akoglu, L., Faloutsos, C.: Valuepick: Towards a value-oriented dual-goal recommender system. In: W. Fan, W. Hsu, G.I. Webb, B. Liu, C. Zhang, D. Gunopulos, X. Wu (eds.) ICDMW 2010, The

Table 15 UserGCE for `Amazon Electronics` dataset using the three user features considered. Notation as in Table 12.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	MADR	MADr
Random	0.000	−1.68	−6.13	**−0.01**	**0.000**	**0.000**
MostPopular	0.008	**−0.02**	**−0.51**	−0.19	0.003	0.271
ItemKNN	0.004	−0.04	−0.61	−0.15	0.002	0.004
UserKNN	0.014	−0.04	−0.63	−0.15	0.007	0.003
SVD++	0.012	−0.04	−0.61	−0.16	0.006	0.017
BPRMF	**0.018**	−0.05	−0.69	−0.13	0.010	0.004
BPRSlim	0.007	−0.06	−0.73	−0.12	0.004	0.158

(a) Happiness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−2.53	−11.04	−11.04	**−0.35**	−0.81	**0.000**	**0.000**
MostPopular	0.008	−0.01	−0.42	−0.44	−0.71	−0.67	0.002	0.759
ItemKNN	0.004	**−0.01**	−0.44	−0.46	−0.69	**−0.63**	0.001	0.009
UserKNN	0.014	−0.06	−0.25	**−0.42**	−0.86	−1.08	0.007	0.012
SVD++	0.012	−0.03	−0.28	−0.52	−0.82	−0.75	0.004	0.003
BPRMF	**0.018**	−0.04	−0.25	−0.46	−0.80	−1.02	0.008	0.010
BPRSlim	0.007	−0.04	**−0.23**	−0.52	−0.94	−0.84	0.003	0.444

(b) Helpfulness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−2.45	−0.74	−10.74	−10.74	−0.35	**0.000**	**0.000**
MostPopular	0.008	−0.01	−0.67	−0.59	−0.57	−0.39	0.001	1.805
ItemKNN	0.004	−0.16	−1.24	−1.67	**−0.41**	**−0.15**	0.003	0.027
UserKNN	0.014	−0.01	−0.73	**−0.53**	−0.55	−0.41	0.002	0.023
SVD++	0.012	0.00	−0.60	−0.57	−0.62	−0.41	0.002	0.000
BPRMF	**0.018**	**0.00**	**−0.54**	−0.61	−0.58	−0.45	0.002	0.013
BPRSlim	0.007	−0.05	−1.11	−0.74	−0.48	−0.25	0.003	1.302

(c) Interactions

10th IEEE International Conference on Data Mining Workshops, Sydney, Australia, 13 December 2010, pp. 1151–1158. IEEE Computer Society (2010). DOI 10.1109/ICDMW.2010.68. URL `https://doi.org/10.1109/ICDMW.2010.68`

11. Anelli, V.W., Bellini, V., Di Noia, T., Bruna, W.L., Tomeo, P., Di Sciascio, E.: An analysis on time- and session-aware diversification in recommender systems. In: UMAP, pp. 270–274. ACM (2017)
12. Anelli, V.W., Noia, T.D., Sciascio, E.D., Ragone, A., Trotta, J.: Time-aware personalized popularity in top-n recommendation. In: Workshop on Recommendation in Complex Scenarios co-located with 12th ACM Conference on Recommender Systems (RecSys 2018), Vancouver, BC, Canada, October 2-7, 2018 (2018)
13. Anelli, V.W., Noia, T.D., Sciascio, E.D., Ragone, A., Trotta, J.: Local popularity and time in top-n recommendation. In: L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, D. Hiemstra (eds.) Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I, *Lecture Notes in Computer Science*, vol. 11437, pp. 861–868. Springer (2019). DOI 10.1007/978-3-030-15712-8_63. URL `https://doi.org/10.1007/978-3-030-15712-8\_63`
14. Azaria, A., Hassidim, A., Kraus, S., Eshkol, A., Weintraub, O., Netanely, I.: Movie recommender system for profit maximization. In: Q. Yang, I. King, Q. Li, P. Pu, G. Karypis (eds.) Seventh ACM Conference on Recommender Systems, RecSys '13, Hong Kong, China, October 12-16, 2013, pp. 121–128. ACM (2013). DOI 10.1145/2507157.2507162. URL `https://doi.org/10.1145/2507157.2507162`
15. Backstrom, L., Leskovec, J.: Supervised random walks: predicting and recommending links in social networks. In: Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011, pp. 635–644 (2011)

Table 16 UserGCE for `Amazon Video Games` dataset using the three user features considered. Notation as in Table 12.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	MADR	MADr
Random	0.000	−7.22	−24.10	−0.22	**0.000**	**0.000**
MostPopular	0.004	−0.05	−0.66	−0.14	0.002	0.373
ItemKNN	0.013	**0.00**	−0.37	−0.27	0.002	0.006
UserKNN	**0.019**	−0.01	−0.42	−0.24	0.004	0.010
SVD++	0.005	−0.48	−2.22	**−0.01**	0.006	0.618
BPRMF	0.008	−0.12	−0.97	−0.08	0.007	0.038
BPRSlim	0.011	0.00	**−0.26**	−0.39	0.002	0.099

(a) Happiness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−7.58	−31.08	−1.95	−1.10	−31.08	**0.000**	**0.000**
MostPopular	0.004	**−0.01**	−0.43	−0.65	**−0.45**	−0.67	0.001	0.201
ItemKNN	0.013	−0.05	−0.32	−0.44	−1.30	−0.56	0.005	0.015
UserKNN	**0.019**	−0.10	**−0.16**	−0.50	−1.21	−1.08	0.012	0.006
SVD++	0.005	−0.44	−0.42	−0.56	−4.51	**−0.29**	0.003	0.103
BPRMF	0.008	−0.29	−0.23	**−0.28**	−2.92	−1.14	0.006	0.103
BPRSlim	0.011	−0.10	−0.25	−0.33	−1.47	−0.93	0.006	0.346

(b) Helpfulness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.000	−6.46	−0.94	−26.69	−26.69	−1.60	**0.000**	**0.000**
MostPopular	0.004	−0.13	−1.00	−1.68	**−0.30**	−0.25	0.003	0.356
ItemKNN	0.013	−0.05	−1.02	**−0.22**	−0.71	−0.60	0.006	0.046
UserKNN	**0.019**	**−0.04**	−1.13	−0.41	−0.51	−0.40	0.006	0.014
SVD++	0.005	−0.11	−1.73	−0.59	−0.60	**−0.19**	0.003	0.047
BPRMF	0.008	−0.04	**−0.44**	−0.43	−0.44	−1.14	0.002	0.327
BPRSlim	0.011	−0.08	−0.59	−0.45	−1.51	−0.27	0.005	1.260

(c) Interactions

16. Balabanovic, M., Shoham, Y.: Content-based, collaborative recommendation. Commun. ACM **40**(3), 66–72 (1997). DOI 10.1145/245108.245124. URL `http://doi.acm.org/10.1145/245108.245124`
17. Barocas, S., Selbst, A.D.: Big data's disparate impact. Cal. L. Rev. **104**, 671 (2016)
18. Belkin, N.J., Robertson, S.E.: Some ethical and political implications of theoretical research in information science. In: Proceedings of the Association for Information Science, ASIS '76, pp. 597–605 (1976)
19. Bellogín, A., Castells, P., Cantador, I.: Statistical biases in information retrieval metrics for recommender systems. Inf. Retr. Journal **20**(6), 606–634 (2017). DOI 10.1007/s10791-017-9312-z. URL `https://doi.org/10.1007/s10791-017-9312-z`
20. Biega, A.J., Gummadi, K.P., Weikum, G.: Equity of attention: Amortizing individual fairness in rankings. In: K. Collins-Thompson, Q. Mei, B.D. Davison, Y. Liu, E. Yilmaz (eds.) The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018, pp. 405–414. ACM (2018). DOI 10.1145/3209978.3210063. URL `https://doi.org/10.1145/3209978.3210063`
21. Billsus, D., Pazzani, M.J.: User modeling for adaptive news access. User Model. User-Adapt. Interact. **10**(2-3), 147–180 (2000). DOI 10.1023/A:1026501525781. URL `https://doi.org/10.1023/A:1026501525781`
22. Boratto, L., Fenu, G., Marras, M.: The effect of algorithmic bias on recommender systems for massive open online courses. In: L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, D. Hiemstra (eds.) Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I, *Lecture Notes in Computer Science*, vol. 11437, pp.

Table 17 UserGCE for `MovieLens-1M` dataset using the two user features considered for the rest of the datasets, plus age and gender. Notation as in Table 12.

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.004	−0.01	−0.72	−0.43	−0.52	**−0.53**	**0.001**	**0.000**
MostPopular	0.081	−0.03	**−0.43**	**−0.37**	−0.56	−1.04	0.024	57.740
ItemKNN	0.095	**0.00**	−0.45	−0.51	−0.55	−0.67	0.011	0.028
UserKNN	**0.107**	0.00	−0.51	−0.46	−0.51	−0.71	0.013	0.019
SVD++	0.070	−0.01	−0.44	−0.45	−0.58	−0.76	0.012	0.014
BPRMF	0.094	−0.01	−0.49	−0.43	−0.52	−0.80	0.016	0.083
BPRSlim	0.097	0.00	−0.53	−0.49	**−0.48**	−0.70	0.011	0.539

(a) Age

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	MADR	MADr
Random	0.004	**0.00**	**−0.37**	−0.28	**0.001**	**0.000**
MostPopular	0.081	−0.02	−0.51	**−0.19**	0.029	79.377
ItemKNN	0.095	0.00	−0.37	−0.27	0.012	0.045
UserKNN	**0.107**	−0.01	−0.41	−0.24	0.021	0.027
SVD++	0.070	−0.01	−0.44	−0.23	0.018	0.026
BPRMF	0.094	−0.01	−0.42	−0.24	0.020	0.066
BPRSlim	0.097	0.00	−0.38	−0.27	0.014	0.717

(b) Gender

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	MADR	MADr
Random	0.004	−0.01	−0.22	−0.45	0.001	**0.000**
MostPopular	0.081	0.00	−0.28	−0.37	0.009	55.632
ItemKNN	0.095	**0.00**	−0.32	−0.32	**0.000**	0.063
UserKNN	**0.107**	0.00	−0.33	−0.31	0.001	0.032
SVD++	0.070	−0.01	**−0.22**	−0.46	0.020	0.412
BPRMF	0.094	0.00	−0.29	−0.35	0.007	0.075
BPRSlim	0.097	0.00	−0.33	**−0.31**	0.001	0.832

(c) Happiness

Rec	nDCG	p_{f_0}	p_{f_1}	p_{f_2}	p_{f_3}	p_{f_4}	MADR	MADr
Random	0.004	−0.19	−1.53	−1.34	−0.83	**−0.06**	**0.004**	**0.000**
MostPopular	0.081	−0.13	−1.43	−1.09	**−0.60**	−0.11	0.059	215.619
ItemKNN	0.095	**−0.02**	−0.63	**−0.75**	−0.73	−0.26	0.029	0.222
UserKNN	**0.107**	−0.04	**−0.53**	−0.88	−0.81	−0.24	0.042	0.170
SVD++	0.070	−0.15	−1.60	−1.06	−0.62	−0.11	0.053	0.070
BPRMF	0.094	−0.05	−0.88	−0.92	−0.63	−0.18	0.045	0.315
BPRSlim	0.097	−0.03	−0.54	−0.87	−0.70	−0.28	0.033	3.698

(d) Interactions

457–472. Springer (2019). DOI 10.1007/978-3-030-15712-8_30. URL `https://doi.org/10.1007/978-3-030-15712-8\_30`

23. Botev, Z.I., Kroese, D.P.: The generalized cross entropy method, with applications to probability density estimation. Methodology and Computing in Applied Probability **13**(1), 1–27 (2011)
24. Breese, J.S., Heckerman, D., Kadie, C.M.: Empirical analysis of predictive algorithms for collaborative filtering. In: G.F. Cooper, S. Moral (eds.) UAI '98: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, University of Wisconsin Business School, Madison, Wisconsin, USA, July 24-26, 1998, pp. 43–52. Morgan Kaufmann (1998). URL `https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1\&smnu=2\&article\_id=231\&proceeding\_id=14`
25. Burke, R.: Multisided fairness for recommendation. arXiv preprint arXiv:1707.00093 (2017)

26. Burke, R., Sonboli, N., Ordonez-Gauger, A.: Balanced neighborhoods for multi-sided fairness in recommendation. In: S.A. Friedler, C. Wilson (eds.) Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA, *Proceedings of Machine Learning Research*, vol. 81, pp. 202–214. PMLR (2018). URL `http://proceedings.mlr.press/v81/burke18a.html`
27. Burke, R.D., Abdollahpouri, H., Mobasher, B., Gupta, T.: Towards multi-stakeholder utility evaluation of recommender systems. In: F. Cena, M.C. Desmarais, D. Dicheva (eds.) Late-breaking Results, Posters, Demos, Doctoral Consortium and Workshops Proceedings of the 24th ACM Conference on User Modeling, Adaptation and Personalisation (UMAP 2016), Halifax, Canada, July 13-16, 2016., *CEUR Workshop Proceedings*, vol. 1618. CEUR-WS.org (2016). URL `http://ceur-ws.org/Vol-1618/SOAP\_paper2.pdf`
28. Campos, P.G., Díez, F., Cantador, I.: Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. User Model. User-Adapt. Interact. **24**(1-2), 67–119 (2014). DOI 10.1007/s11257-012-9136-x. URL `https://doi.org/10.1007/s11257-012-9136-x`
29. Chen, L., Hsu, F., Chen, M., Hsu, Y.: Developing recommender systems with the consideration of product profitability for sellers. Inf. Sci. **178**(4), 1032–1048 (2008). DOI 10.1016/j.ins.2007.09.027. URL `https://doi.org/10.1016/j.ins.2007.09.027`
30. Christakopoulou, K., Kawale, J., Banerjee, A.: Recommendation with capacity constraints. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017, pp. 1439–1448. ACM (2017). DOI 10.1145/3132847.3133034. URL `https://doi.org/10.1145/3132847.3133034`
31. Cowell, F.A.: Measurement of inequality. Handbook of income distribution **1**, 87–166 (2000)
32. Cowell, F.A., Kuga, K.: Inequality measurement: an axiomatic approach. European Economic Review **15**(3), 287–305 (1981)
33. Cremonesi, P., Koren, Y., Turrin, R.: Performance of recommender algorithms on top-n recommendation tasks. In: X. Amatriain, M. Torrens, P. Resnick, M. Zanker (eds.) Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26-30, 2010, pp. 39–46. ACM (2010). DOI 10.1145/1864708.1864721. URL `https://doi.org/10.1145/1864708.1864721`
34. Csiszár, I.: A class of measures of informativity of observation channels. Periodica Mathematica Hungarica **2**(1-4), 191–213 (1972)
35. Dalton, H.: The Measurement of the Inequality of Incomes. The Economic Journal **30**(119), 348–361 (1920)
36. Das, A., Mathieu, C., Ricketts, D.: Maximizing profit using recommender systems. CoRR **abs/0908.3633** (2009). URL `http://arxiv.org/abs/0908.3633`
37. Deldjoo, Y., Anelli, V.W., Zamani, H., Kouki, A.B., Noia, T.D.: Recommender systems fairness evaluation via generalized cross entropy. In: R. Burke, H. Abdollahpouri, E.C. Malthouse, K.P. Thai, Y. Zhang (eds.) Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019), Copenhagen, Denmark, September 20, 2019, *CEUR Workshop Proceedings*, vol. 2440. CEUR-WS.org (2019). URL `http://ceur-ws.org/Vol-2440/short3.pdf`
38. Deldjoo, Y., Dacrema, M.F., Constantin, M.G., Eghbal-zadeh, H., Cereda, S., Schedl, M., Ionescu, B., Cremonesi, P.: Movie genome: alleviating new item cold start in movie recommendation. User Model. User-Adapt. Interact. **29**(2), 291–343 (2019). DOI 10.1007/s11257-019-09221-y. URL `https://doi.org/10.1007/s11257-019-09221-y`
39. Deldjoo, Y., Schedl, M., Hidasi, B., Knees, P.: Multimedia recommender systems. In: S. Pera, M.D. Ekstrand, X. Amatriain, J. O'Donovan (eds.) Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018, pp. 537–538. ACM (2018). DOI 10.1145/3240323.3241620. URL `https://doi.org/10.1145/3240323.3241620`
40. Dong, W., Moses, C., Li, K.: Efficient k-nearest neighbor graph construction for generic similarity measures. In: Proceedings of the 20th international conference on World wide web, pp. 577–586. ACM (2011)
41. Ekstrand, M.D., Tian, M., Azpiazu, I.M., Ekstrand, J.D., Anuyah, O., McNeill, D., Pera, M.S.: All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In: Conference on Fairness, Accountability and Transparency, pp. 172–186 (2018)
42. Grgic-Hlaca, N., Zafar, M.B., Gummadi, K.P., Weller, A.: Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In: S.A. McIlraith, K.Q. Weinberger (eds.) Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the

30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pp. 51–60. AAAI Press (2018). URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16523

43. Gunawardana, A., Shani, G.: Evaluating recommender systems. In: F. Ricci, L. Rokach, B. Shapira (eds.) Recommender Systems Handbook, pp. 265–308. Springer (2015). DOI 10.1007/978-1-4899-7637-6_8. URL https://doi.org/10.1007/978-1-4899-7637-6_8
44. Hardt, M., Price, E., Srebro, N.: Equality of opportunity in supervised learning. In: D.D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (eds.) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, pp. 3315–3323 (2016). URL http://papers.nips.cc/paper/6374-equality-of-opportunity-in-supervised-learning
45. Havrda, J., Charvát, F.: Quantification method of classification processes. concept of structural a-entropy. Kybernetika **3**(1), 30–35 (1967)
46. He, R., McAuley, J.J.: Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In: J. Bourdeau, J. Hendler, R. Nkambou, I. Horrocks, B.Y. Zhao (eds.) Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016, pp. 507–517. ACM (2016). DOI 10.1145/2872427.2883037. URL https://doi.org/10.1145/2872427.2883037
47. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.: Neural collaborative filtering. In: R. Barrett, R. Cummings, E. Agichtein, E. Gabrilovich (eds.) Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017, pp. 173–182. ACM (2017). DOI 10.1145/3038912.3052569. URL https://doi.org/10.1145/3038912.3052569
48. Herlocker, J.L., Konstan, J.A., Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. Inf. Retr. **5**(4), 287–310 (2002). DOI 10.1023/A:1020443909834. URL https://doi.org/10.1023/A:1020443909834
49. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy, pp. 263–272. IEEE Computer Society (2008). DOI 10.1109/ICDM.2008.22. URL https://doi.org/10.1109/ICDM.2008.22
50. Jannach, D., Adomavicius, G.: Price and profit awareness in recommender systems. CoRR **abs/1707.08029** (2017). URL http://arxiv.org/abs/1707.08029
51. Jannach, D., Lerche, L., Kamehkhosh, I., Jugovac, M.: What recommenders recommend: an analysis of recommendation biases and possible countermeasures. User Model. User-Adapt. Interact. **25**(5), 427–491 (2015). DOI 10.1007/s11257-015-9165-3. URL https://doi.org/10.1007/s11257-015-9165-3
52. Jannach, D., Resnick, P., Tuzhilin, A., Zanker, M.: Recommender systems - : beyond matrix completion. Commun. ACM **59**(11), 94–102 (2016). DOI 10.1145/2891406. URL https://doi.org/10.1145/2891406
53. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of ir techniques. ACM Transactions on Information Systems (TOIS) **20**(4), 422–446 (2002)
54. Kapur, J.N., Kesavan, H.K.: The generalized maximum entropy principle (with applications). Sandford Educational Press Waterloo, Ontario (1987)
55. Kim, Y., Stratos, K., Sarikaya, R.: Frustratingly easy neural domain adaptation. In: N. Calzolari, Y. Matsumoto, R. Prasad (eds.) COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan, pp. 387–396. ACL (2016). URL http://aclweb.org/anthology/C/C16/C16-1038.pdf
56. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Y. Li, B. Liu, S. Sarawagi (eds.) Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, pp. 426–434. ACM (2008). DOI 10.1145/1401890.1401944. URL https://doi.org/10.1145/1401890.1401944
57. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009)
58. Lang, K.: Newsweeder: Learning to filter netnews. In: in Proceedings of the 12th International Machine Learning Conference (ML95 (1995)
59. Liu, W., Burke, R.: Personalizing fairness-aware re-ranking. In: 2nd FATREC Workshop on Responsible Recommendation (2018)

60. McAuley, J.J., Targett, C., Shi, Q., van den Hengel, A.: Image-based recommendations on styles and substitutes. In: R.A. Baeza-Yates, M. Lalmas, A. Moffat, B.A. Ribeiro-Neto (eds.) Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015, pp. 43–52. ACM (2015). DOI 10.1145/2766462.2767755. URL https://doi.org/10.1145/2766462.2767755
61. McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: G.M. Olson, R. Jeffries (eds.) Extended Abstracts Proceedings of the 2006 Conference on Human Factors in Computing Systems, CHI 2006, Montréal, Québec, Canada, April 22-27, 2006, pp. 1097–1101. ACM (2006). DOI 10.1145/1125451.1125659. URL https://doi.org/10.1145/1125451.1125659
62. Mehrotra, R., Anderson, A., Diaz, F., Sharma, A., Wallach, H.M., Yilmaz, E.: Auditing search engines for differential satisfaction across demographics. In: R. Barrett, R. Cummings, E. Agichtein, E. Gabrilovich (eds.) Proceedings of the 26th International Conference on World Wide Web Companion, Perth, Australia, April 3-7, 2017, pp. 626–633. ACM (2017). DOI 10.1145/3041021.3054197. URL https://doi.org/10.1145/3041021.3054197
63. Mehrotra, R., McInerney, J., Bouchard, H., Lalmas, M., Diaz, F.: Towards a fair marketplace: Counterfactual evaluation of the trade-off between relevance, fairness & satisfaction in recommendation systems. In: A. Cuzzocrea, J. Allan, N.W. Paton, D. Srivastava, R. Agrawal, A.Z. Broder, M.J. Zaki, K.S. Candan, A. Labrinidis, A. Schuster, H. Wang (eds.) Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, pp. 2243–2251. ACM (2018). DOI 10.1145/3269206.3272027. URL https://doi.org/10.1145/3269206.3272027
64. Ning, X., Karypis, G.: SLIM: sparse linear methods for top-n recommender systems. In: D.J. Cook, J. Pei, W. Wang, O.R. Zaïane, X. Wu (eds.) 11th IEEE International Conference on Data Mining, ICDM 2011, Vancouver, BC, Canada, December 11-14, 2011, pp. 497–506. IEEE Computer Society (2011). DOI 10.1109/ICDM.2011.134. URL https://doi.org/10.1109/ICDM.2011.134
65. Pan, R., Zhou, Y., Cao, B., Liu, N.N., Lukose, R.M., Scholz, M., Yang, Q.: One-class collaborative filtering. In: Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy, pp. 502–511. IEEE Computer Society (2008). DOI 10.1109/ICDM.2008.16. URL https://doi.org/10.1109/ICDM.2008.16
66. Panniello, U., Gorgoglione, M., Hill, S., Hosanagar, K.: Incorporating profit margins into recommender systems: A randomized field experiment of purchasing behavior and consumer trust (2014)
67. Pigou, A.: Wealth and Welfare. PCMI collection. Macmillan and Company, limited (1912)
68. Qamar, A.M., Gaussier, E., Chevallet, J.P., Lim, J.H.: Similarity learning for nearest neighbor classification. In: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on, pp. 983–988. IEEE (2008)
69. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: bayesian personalized ranking from implicit feedback. In: J.A. Bilmes, A.Y. Ng (eds.) UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009, pp. 452–461. AUAI Press (2009). URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1\&smnu=2\&article_id=1630\&proceeding_id=25
70. Sapiezynski, P., Kassarnig, V., Wilson, C.: Academic performance prediction in a gender-imbalanced environment. In: 1st FATREC Workshop on Responsible Recommendation (2017)
71. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of recommendation algorithms for e-commerce. In: Proceedings of the 2nd ACM conference on Electronic commerce, pp. 158–167. ACM (2000)
72. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: V.Y. Shen, N. Saito, M.R. Lyu, M.E. Zurko (eds.) Proceedings of the Tenth International World Wide Web Conference, WWW 10, Hong Kong, China, May 1-5, 2001, pp. 285–295. ACM (2001). DOI 10.1145/371920.372071. URL https://doi.org/10.1145/371920.372071
73. Shani, G., Heckerman, D., Brafman, R.I.: An mdp-based recommender system. J. Mach. Learn. Res. **6**, 1265–1295 (2005). URL http://jmlr.org/papers/v6/shani05a.html
74. Singh, A., Joachims, T.: Fairness of exposure in rankings. In: Y. Guo, F. Farooq (eds.) Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018, pp. 2219–2228. ACM (2018). DOI 10.1145/3219819.3220088. URL https://doi.org/10.1145/3219819.3220088
75. Speicher, T., Heidari, H., Grgic-Hlaca, N., Gummadi, K.P., Singla, A., Weller, A., Zafar, M.B.: A unified approach to quantifying algorithmic unfairness: Measuring individual & group unfairness via

inequality indices. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2239–2248. ACM (2018)

76. Sühr, T., Biega, A.J., Zehlike, M., Gummadi, K.P., Chakraborty, A.: Two-sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019, pp. 3082–3092 (2019). DOI 10.1145/3292500.3330793. URL `https://doi.org/10.1145/3292500.3330793`
77. Sürer, Ö., Burke, R., Malthouse, E.C.: Multistakeholder recommendation with provider constraints. In: Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018, pp. 54–62 (2018). DOI 10.1145/3240323.3240350. URL `https://doi.org/10.1145/3240323.3240350`
78. Tsintzou, V., Pitoura, E., Tsaparas, P.: Bias disparity in recommendation systems. In: Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019), Copenhagen, Denmark, September 20, 2019, *CEUR Workshop Proceedings*, vol. 2440. CEUR-WS.org (2019). URL `http://ceur-ws.org/Vol-2440/short4.pdf`
79. Verma, S., Rubin, J.: Fairness definitions explained. In: Proceedings of the International Workshop on Software Fairness, FairWare@ICSE 2018, Gothenburg, Sweden, May 29, 2018, pp. 1–7. ACM (2018). DOI 10.1145/3194770.3194776. URL `https://doi.org/10.1145/3194770.3194776`
80. Wang, H., Wu, C.: A mathematical model for product selection strategies in a recommender system. Expert Syst. Appl. **36**(3), 7299–7308 (2009). DOI 10.1016/j.eswa.2008.09.006. URL `https://doi.org/10.1016/j.eswa.2008.09.006`
81. Wang, X., He, X., Chua, T.: Learning and reasoning on graph for recommendation. In: J. Caverlee, X.B. Hu, M. Lalmas, W. Wang (eds.) WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020, pp. 890–893. ACM (2020). DOI 10.1145/3336191.3371873. URL `https://doi.org/10.1145/3336191.3371873`
82. Yao, S., Huang, B.: Beyond parity: Fairness objectives for collaborative filtering. In: I. Guyon, U. von Luxburg, S. Bengio, H.M. Wallach, R. Fergus, S.V.N. Vishwanathan, R. Garnett (eds.) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp. 2921–2930 (2017). URL `http://papers.nips.cc/paper/6885-beyond-parity-fairness-objectives-for-collaborative-filtering`
83. Zafar, M.B., Valera, I., Gomez-Rodriguez, M., Gummadi, K.P.: Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In: R. Barrett, R. Cummings, E. Agichtein, E. Gabrilovich (eds.) Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017, pp. 1171–1180. ACM (2017). DOI 10.1145/3038912.3052660. URL `https://doi.org/10.1145/3038912.3052660`
84. Zafar, M.B., Valera, I., Gomez-Rodriguez, M., Gummadi, K.P.: Fairness constraints: A flexible approach for fair classification. J. Mach. Learn. Res. **20**, 75:1–75:42 (2019). URL `http://jmlr.org/papers/v20/18-262.html`
85. Zafar, M.B., Valera, I., Gomez-Rodriguez, M., Gummadi, K.P., Weller, A.: From parity to preference-based notions of fairness in classification. In: I. Guyon, U. von Luxburg, S. Bengio, H.M. Wallach, R. Fergus, S.V.N. Vishwanathan, R. Garnett (eds.) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp. 229–239 (2017). URL `http://papers.nips.cc/paper/6627-from-parity-to-preference-based-notions-of-fairness-in-classification`
86. Zamani, H., Croft, W.B.: Learning a joint search and recommendation model from user-item interactions. In: Proceedings of the 13th International Conference on Web Search and Data Mining, WSDM '20, p. 717–725. Association for Computing Machinery, New York, NY, USA (2020). DOI 10.1145/3336191.3371818. URL `https://doi.org/10.1145/3336191.3371818`
87. Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., Baeza-Yates, R.A.: Fa*ir: A fair top-k ranking algorithm. In: E. Lim, M. Winslett, M. Sanderson, A.W. Fu, J. Sun, J.S. Culpepper, E. Lo, J.C. Ho, D. Donato, R. Agrawal, Y. Zheng, C. Castillo, A. Sun, V.S. Tseng, C. Li (eds.) Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017, pp. 1569–1578. ACM (2017). DOI 10.1145/3132847.3132938. URL `https://doi.org/10.1145/3132847.3132938`
88. Zhai, C., Lafferty, J.D.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: W.B. Croft, D.J. Harper, D.H. Kraft, J. Zobel (eds.) SIGIR 2001: Proceedings

of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 9-13, 2001, New Orleans, Louisiana, USA, pp. 334–342. ACM (2001). DOI 10.1145/383952.384019. URL https://doi.org/10.1145/383952.384019

89. Zheng, Y., Ghane, N., Sabouri, M.: Personalized educational learning with multi-stakeholder optimizations. In: G.A. Papadopoulos, G. Samaras, S. Weibelzahl, D. Jannach, O.C. Santos (eds.) Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization, UMAP 2019, Larnaca, Cyprus, June 09-12, 2019, pp. 283–289. ACM (2019). DOI 10.1145/3314183.3323843. URL https://doi.org/10.1145/3314183.3323843
90. Zhu, Z., Hu, X., Caverlee, J.: Fairness-aware tensor-based recommendation. In: A. Cuzzocrea, J. Allan, N.W. Paton, D. Srivastava, R. Agrawal, A.Z. Broder, M.J. Zaki, K.S. Candan, A. Labrinidis, A. Schuster, H. Wang (eds.) Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, pp. 1153–1162. ACM (2018). DOI 10.1145/3269206.3271795. URL https://doi.org/10.1145/3269206.3271795