Towards a Fair Marketplace: Counterfactual Evaluation of the trade-off between Relevance, Fairness & Satisfaction in Recommendation Systems

Rishabh Mehrotra¹, James McInerney¹, Hugues Bouchard¹, Mounia Lalmas¹, Fernando Diaz²

¹Spotify Research, ²Microsoft Research

{rishabh, jamesm, hb, mounial}@spotify.com, diazf@acm.org

ABSTRACT

Two-sided marketplaces are platforms that have customers not only on the demand side (e.g. users), but also on the supply side (e.g. retailer, artists). While traditional recommender systems focused specifically towards increasing consumer satisfaction by providing relevant content to consumers, two-sided marketplaces face the problem of additionally optimizing for supplier preferences, and visibility. Indeed, the suppliers would want a fair opportunity to be presented to users. Blindly optimizing for consumer relevance may have a detrimental impact on supplier fairness. Motivated by this problem, we focus on the trade-off between objectives of consumers and suppliers in the case of music streaming services, and consider the trade-off between relevance of recommendations to the consumer (i.e. user) and fairness of representation of suppliers (i.e. artists) and measure their impact on consumer satisfaction.

We propose a conceptual and computational framework using counterfactual estimation techniques to understand, and evaluate different recommendation policies, specifically around the trade-off between relevance and fairness, without the need for running many costly A/B tests. We propose a number of recommendation policies which jointly optimize relevance and fairness, thereby achieving substantial improvement in supplier fairness without noticeable decline in user satisfaction. Additionally, we consider user disposition towards fair content, and propose a personalized recommendation policy which takes into account consumer’s tolerance towards fair content. Our findings could guide the design of algorithms powering two-sided marketplaces, as well as guide future research on sophisticated algorithms for joint optimization of user relevance, satisfaction and fairness.

ACM Reference Format:

¹Work conducted while the author was at Spotify.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM ’18, October 22–26, 2018, Torino, Italy
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-6014-2/18/10...$15.00
https://doi.org/10.1145/3269206.3272027

Figure 1: Exposure of artist playlists on a music app. A small number of artists receive the highest relevance score for most users.

1 INTRODUCTION

Two-sided marketplaces act as intermediaries that help facilitate economic interaction between two sets of agents, for example, consumers and suppliers, or users and advertisers. In recent years, online two-sided marketplaces have steadily emerged as a central point for users to accomplish their tasks in a number of different contexts, including finding accommodation (Airbnb, Booking.com), watching video content (YouTube, Dailymotion), ridesharing (Uber, Didi, Lyft), online shopping (Etsy, Ebay, music (Spotify, Pandora, Soundcloud), finding apps (Apple and Google App Store) and searching for jobs (LinkedIn). Users buy products, watch movies, listen to music, and hire services through intermediaries who connect buyers and suppliers. The scale, convenience, and speed of such marketplaces are enabled by recommendation systems and search engines that use predicted relevance to match buyers to suppliers.

The attention given to suppliers as a result of predicted relevance is often vastly unequal. A relatively small group of superstar suppliers receive a large portion of attention in many marketplaces while the majority of suppliers in the long tail receives very little. For example, Figure 1 shows the relative exposure given to artist playlists in a music app across the popularity spectrum.

Prior user familiarity and exposure outside the online marketplace (e.g. through advertising, word of mouth) are no doubt the main reason for such disparities. For example, the relevance of a Tom Cruise movie to a fan of action movies tends to be high not only because the actor has the skill and clout to star in high quality movies but also because of pre-existing familiarity with the actor and movie. In this way, the predicted relevance score from standard recommendation methods (e.g. matrix factorization, word2vec) represents both the match quality and familiarity extent between a user and item.

In addition to pre-existing familiarity, we identify a second contributing factor to this attention disparity - the recommendation
strategies powering 2-sided marketplaces. Recommendation systems suffer from an inherent problem of "superstar economics" [21]: rankings have a top and a tail end, not just for popularity, but also for relevance, as is evident in Figure 1. In an attempt to maximize user satisfaction, recommender system optimize for relevance. This inadvertently leads to lock-in of popular and relevant suppliers, especially for users who want to minimize the effort required to interact with the system. A major side-effect of the superstar economics is the impedance to suppliers on the tail-end of the spectrum, who struggle to attract consumers, given the low exposure, and thus, are not satisfied with the marketplace. Indeed, to continue to attract more suppliers to the platform, two-sided marketplaces face an interesting problem of optimizing their models for supplier exposure, and visibility. Indeed, the suppliers (e.g. retailers, artists) would want a fair opportunity to be presented to the users.

In this work, we aim at understanding the interplay between relevance, fairness and satisfaction in a two-sided marketplace ecosystem. Specifically, we consider the trade-off between consumer relevance and supplier fairness and discuss its impact on consumer satisfaction. A system optimizing for relevance might be unfair to unpopular suppliers, i.e. conditioned on being known to a user, a supplier with the same user satisfaction might have a lower probability of being recommended. On the other hand, exposing all suppliers equally might severely impact consumer satisfaction. Indeed, evaluating such trade-offs between relevance and exposure to different suppliers is hard, since offline estimates are usually confounded and running large scale A/B tests is not only expensive and time consuming, but it might severely impact user experience.

We propose to address this problem using causal inference techniques, under the counterfactual evaluation framework. This approach effectively allows one to run many costly A/B tests offline from logged data, making it possible to estimate and optimize different metrics quickly and inexpensively. We consider the case of music recommendation in a streaming platform, and present results on the impact on user satisfaction when a system optimizes for consumer relevance, versus a system optimizing for supplier fairness. Further, we propose a number of recommendation policies which balance the objectives of consumer relevance and supplier fairness. Additionally, we study user-level disposition and establish that users have varying affinity and sensitivity towards popular and fair content. We leverage this insight and propose a personalized method of recommendation which takes into account consumer level disposition while recommending content. Unbiased estimation of satisfaction metrics using the counterfactual framework enables us to understand, devise and evaluate fairness aware recommendation strategies.

In summary, we make the following contributions:

- We identify, and formalize the importance of explicitly considering supplier fairness, and investigate the interplay between fairness, relevance in a 2-sided marketplace setting.
- We propose a conceptual and computational framework, based on counterfactual estimation techniques which provide an unbiased estimate of metrics. We leverage the proposed framework to understand how fairness and relevance impact user satisfaction in a live music streaming platform.
- We propose a number of recommendation policies, which jointly optimize for supplier fairness and consumer relevance.
- We propose a personalized fairness aware recommendation strategy, which leverages user level disposition towards fair content.

The proposed framework is not limited to a specific definition of fairness or satisfaction, and is generic enough to enable plugging of various definitions and constraints.

2 RELATED WORK

**Fairness, Relevance & Satisfaction:**
The growing ubiquity of data-driven learning models in algorithmic decision-making has recently boosted concerns about the issues of fairness and bias. Recent work has explored the development of classification models with fairness-aware regularization [30], fairness aware decision making systems [9], prediction models for individual fairness [6], fair division of resources [2], auditing for fairness [17] and fairness in rankings [4, 25]. While most work on fairness so far has focused on users, and individuals, our work focuses exclusively on supplier fairness in marketplaces.

Satisfaction can be understood as the fulfillment of a specified desire or goal [13]. However, satisfaction itself is a subjective construct and is difficult to measure. Techniques to estimate satisfaction range from collecting explicit feedback from users [8, 10, 18], to implicit signals of satisfaction [11, 28]. Recent work has also explored ways of incorporating (business) constraints while optimizing metrics in a ranking setting [24].

**Marketplaces:**
Research on platforms, and marketplaces have enjoyed a long history of detailed research [23, 26], with past work exploring competition [3], strategies [7] and economies [16] in such marketplaces. The concept of multiple stakeholders in recommender systems is also suggested in prior research [1], including a previous attempt on considering multi-sided fairness in marketplaces [5]. There is a substantial literature in real-time targeted advertising in which advertisers constraints are incorporated into the decision to deliver personalized advertising to a user (see [29] for a detailed survey). The current work is among the first to consider the impact on user satisfaction in an online streaming marketplace.

**Counterfactual Estimation:**
Off-policy evaluation, and counterfactual estimation of metrics has recently gained a lot of attention in the research community, with recent work focusing on counterfactual estimation of search engine metrics [14], for evaluating slate recommendations [27]. Nedelec et al. [20] present a detailed overview and comparison of many different counterfactual estimators.

3 PROBLEM FORMULATION

To continuously attract suppliers to the platform, recommender systems powering 2-sided marketplaces should not only consider consumer relevance and satisfaction, but also take into account the impact of their recommendations on suppliers. We motivate the
need for explicitly considering supplier fairness while recommend-
ing content.

Data Context: We consider the specific use case of a global music streaming platform as a 2-sided marketplace, with consumers being users who listen to music, and artists being the suppliers. The recommendation system recommends a set of tracks (i.e. songs) to the user, each of which could come from different artists. Different sets have varying degree of relevance to user’s interests, and users could be satisfied with the recommended set to varying extent.

3.1 Definitions
We define key concepts of user relevance, supplier fairness and user satisfaction, which are used throughout the paper.

Relevance: Personalization is the ability to recommend content and services tailored to individuals (i.e. users) based on knowledge about their preferences and behavior. Personalized recommendations rely on making recommendations that are relevant to the user. We operationalize the notion of relevance for the user and identify a recommendation as relevant if it closely resembles user’s interest profile. We train a skip-gram model [19] to learn embedding based representation in the join space for both users and tracks, based on historic user interactions with tracks. We then average the track vectors to compute the representation for a set of tracks. The relevance score of a set to a user is computed using cosine similarity. The higher the relevance score, the more relevant a given set is to a user based on their music interests.

User Satisfaction: The central idea in online marketplaces is to keep users satisfied. In this work, we consider satisfaction from the consumer perspective and define it as the subjective measure on the utility of recommendations. The higher the system utility is for the consumer, the more satisfied the consumer would be. Since satisfaction cannot be measured at scale using explicit feedback, recommender systems often rely on implicit feedback based on behavioral signals, such as the number of clicks or the dwell time (i.e., the time interacting with a recommended item). In our analysis and experiments, satisfaction is measured as the number of tracks the user listens to in a recommended set. Higher values indicate greater user satisfaction.

Fairness: Recommendations suffer from an inherent problem of “superstar economics” [21]: rankings have a top and a tail end, and consequently more popular choices remain more popular because they appear at the top of the ranking. This inadvertently leads to lock-in of popular products and items, especially for users who want to minimize access costs. A major side-effect of the superstar economics is the impedance to suppliers on the tail-end of the spectrum, who struggle to attract consumers, and thus, are not satisfied with the marketplace.

Given the prevalence of superstar economics in marketplaces, the recommender systems powering marketplaces should surface content from not only popular artists but also from less popular and new artists. Doing so has major advantages for marketplaces, including attracting new artists, among others. As recommender systems surface content that is increasingly more relevant to the user, the corresponding distributions of recommended tracks & artists tend to become narrow, leasing to extreme personalization where recommendations skew majorly to a single or a small set of content type. Often, this ends up in a small number of popular artists being recommended to users, while newer or less popular artists are not being surfaced by recommender systems. To counter this, we introduce a notion of group fairness, which requires that the content shown to users be spread well across the wide long-tailed popularity spectrum, rather than focusing on a small subset of popular artists.

We operationalize the concept of fairness for the supplier (i.e. artists), using the popularity of the supplier. We divide the artists into different groups based on their position in the popularity spectrum. Specifically, we consider the popularity distribution of all artists, and bin the artists into ten bins of equal size. In light of group fairness, a set of tracks is fair if it contains tracks from artists that belong to different bins. For a set of tracks that only contains tracks from (say) the most popular bins, the fairness estimates should be low. To this end, we compute the group fairness measure (ψ) of a set of tracks (s) as:

\[
\psi(s) = \sum_{i=1}^{K} \sqrt{|A_i \cap P_i|} \psi(s)
\]

where \( K = 10 \) is the number of popularity bins considered, \( P_i \) is the set of artists belonging to the popularity bin \( i \), \( s \) is the recommended set, and \( A_i(s) \) is the collection of artists in the set \( s \), with \( a_j \) being the \( j \)-th track in the set. \( \psi(s) \) rewards sets that are diverse in terms of the different artist popularity bins represented and, as per the current definition, fair to different popularity bins of suppliers. Given the nature of the function, there is more benefit to selecting an artist from a popularity bin not yet having one of its artist already chosen. As soon as an artist is selected from a bin, other artists from the same bin start having diminishing gain owing to the square root function (e.g. \( \sqrt{2+1} > \sqrt{3+0} \)).

There have been numerous attempts to define fairness [25, 30], and it is unlikely that there will be a universal definition that is appropriate across all applications. We contend that our definition of group fairness is one of the many different possible fairness definitions, and the framework presented in this work is amenable to other interpretations and definitions of fairness.

3.2 Contextual Bandit Formulation
Traditional approaches to recommendations (e.g. collaborative filtering) fail at handling uncertainty of relevance of items, and are incompetent at handling new information (new users, new items). Contextual bandits have recently emerged as a viable alternative [15, 31]. We formalize the recommendation problem as a combinatorial contextual bandit problem, wherein the recommender system powering the 2-sided marketplace repeatedly interacts with consumers as follows:

1. the system observes a context \( x \sim D(x) \) over some space \( X \);
2. based on the context, the system chooses an action \( a \in A \) from the space of possible actions (i.e. sets to recommend);
3. given the context and the action, a reward \( r \in [0, 1] \) is drawn from the distribution \( D(r|x, s) \), with rewards in different
To balance the requirements of both consumers and suppliers in a 2-sided marketplace, recommender systems need to strike a balance in terms of the relevance of recommended content to its consumers, and their fairness in terms of opportunity of surfacing different suppliers. In this section, we present a number of recommendation policies wherein the system could trade-off relevance and fairness. We begin by considering only relevance, and only fairness as the optimizing criterion, and then present few interleaved policies.

4.1 Optimizing Relevance
The first policy we investigate considers only the relevance of a set for a given user. Given a collection of sets \( S \), and a given user \( u \), we leverage the embedding based representation learnt for both the user, and each set \( s \) (as described in Section 3.1) and select the most relevant set to recommend which maximizes the following:

\[
\hat{s}_u^r = \text{argmax}_{s \in S} \phi(u, s) \tag{2}
\]

It is well known that recommending relevant content has a positive impact on user satisfaction [28]; so we expect to achieve higher
user satisfaction estimates for this policy. Further, given the preliminary analysis (Section 3.3), we expect lower fairness estimates for the content recommended using this policy.

4.2 Optimizing Fairness

While optimizing for relevance of recommended content to customers can be expected to have a positive impact on customer satisfaction, its important for 2-sided marketplaces to explicitly consider fairness towards different suppliers. The second policy we investigate aims at recommending content which is equally fair towards suppliers. For a given user (u), we compute the fairness estimate for each set (s ∈ S), and recommend the set with the maximum fairness estimate. Specifically: Recommend only based on playlist fairness

\[ s_u^* = \arg\max_{s \in S_u} \psi(s) \]  

(3)

where \( \psi(s) \) is the fairness estimation function described in Section 3.1, and \( S_u \) is the collection of all sets pertinent to the user \( u \). While we anticipate a relatively lower relevance score for the content recommended under this policy, the impact on user satisfaction is hitherto unknown and is discussed in detail in Section 7.

4.3 Combining Relevance & Fairness

We depart from solely optimizing for relevance or fairness, and present the first interpolated policy, which jointly considers a set’s relevance to the user, and its fairness value. For a given user and set, we compute a combined score by a weighted combination of its relevance and fairness estimates, with the parameter \( \beta \in [0, 1] \) deciding on the importance given to each. Specifically:

\[ s_u^* = \arg\max_{s \in S_u} \left( (1 - \beta) \phi(u, s) + \beta \psi(s) \right) \]  

(4)

Varying \( \beta \) from 0 to 1, increases the importance given to relevance of the set to the user, with \( \beta = 0 \) defaulting to the fairness-only policy, and \( \beta = 1 \) defaulting to the relevance only policy. This policy allows us to understand the interplay between fairness and relevance in finer details, and understand how different differentially weighting fairness and relevance impact satisfaction. We call this the Interpolation policy.

4.4 Probabilistic Policy

When combining relevance and fairness, the above policy deterministically combines the estimates via a weighted combination. An alternative is to consider a probabilistic approach of recommendation (probPolicy), wherein the weighting factor (\( \beta \)) deciding on whether to recommend content based on fairness or on relevance. Specifically:

\[ s_u^* = \begin{cases} \arg\max_{s \in S_u} \psi(s) & \text{if } p < \beta \\ \arg\max_{s \in S} \phi(u, s) & \text{otherwise} \end{cases} \]

where \( p \in [0, 1] \) is a randomly generated number, and \( \beta \in [0, 1] \) controls the probabilistic focus on relevance and fairness. Lower values of \( \beta \) favor fair sets being recommended and higher values of \( \beta \) favor recommendation of more relevant sets.

4.5 Guaranteed Relevance

Often, system designers are wary of negatively impacting user satisfaction, and hence prefer showing relevant content, and consequently avoid risky variants which might harm user satisfaction. To address this concern, we develop a policy which guarantees a certain minimum amount of relevance, following which the model has the freedom to show content based on any criterion, including fairness. Specifically,

\[ s_u^* = \arg\max_{s \in S_u} \psi(s) \]  

(5)

s.t. \( \phi(s, u) \geq \beta \)

where the constraint \( \phi(s, u) \geq \beta \) ensures that the minimum relevance of the recommended content is \( \beta \). The value of \( \beta \) guarantees the level of relevance of the recommended content, and the policy selects the set that maximizes fairness from among the set of relevant content. It is important to note that this policy is different to the combination and probabilistic policies presented above, since when combining relevance and fairness, even with \( \beta = 0.8 \), the model gives 0.8 weight to the relevance score, and the resulting recommended set might have a relevance value less than 0.8. The current guaranteed relevance policy overcomes such cases, by ensuring a minimum value of relevance.

While the policies considered so far consider user’s interest features while computing relevance, they ignore user level attributes towards fairness. We next propose a personalized policy which considers the extent to which any given user is tolerant to content which is fair towards suppliers.

5 ADAPTIVE POLICY

We conjecture that users have varying extent of sensitivity towards fair content, with some users only interested in a particular group of suppliers, while others being more flexible around the distribution of suppliers exposed in the recommended content. Such user level disposition towards fairness motivates the need to develop a user-affinity aware recommendation policy which computes user’s affinity towards different types of content (especially fair sets), and adaptively leverages such user-level affinity to recommend content. We next describe the computation of user level affinity (Section 5.1) and present the affinity aware recommender policy (Section 5.2).

5.1 User Fairness Affinity

To compute user’s propensity to like fair content, we leverage historic interaction logs to compute how users behave differently to relevant and fair recommended sets. For each user, for the scope of this section, we assume access to their satisfaction metrics, and compute the difference in user satisfaction when recommended relevant content, versus when recommended fair content. More specifically, we compute fairness affinity as:

\[ \xi_u = \frac{1}{|N_{\phi(u, s) > \alpha_1} \psi(u, s)|} \sum_{\phi(u, s) > \alpha_1} \psi(u, s) \]  

(6)

where \( N_{\psi(u, s) > \alpha_1} \) represents the number of sets which satisfy the appropriate constraint \( \psi(u, s) > \alpha_1 \). We consider the difference in historic user satisfaction with relevant content (\( \sum_{\phi(u, s) > \alpha_2} \psi(u, s) \)), and fair content respectively (\( \sum_{\psi(u, s) > \alpha_2} \psi(u, s) \)) with \( \alpha_1 \) and \( \alpha_2 \) deciding the specific relevance and fairness thresholds. A lower value of \( \xi_u \) indicates that the user is not very tolerant towards
fairness content, with a large dip in satisfaction being observed when moving towards fair recommendations instead of relevant recommendations. Similarly, higher values of $\xi_u$ indicate that the user satisfaction does not change with relevant and fair content, and that the user is more tolerant towards fair content. We next present a recommendation policy which leverages the user affinity towards fair content.

5.2 Affinity aware Recommender

Given each user’s affinity towards fair content, we propose an adaptive policy for recommending sets, which recommends only relevant sets to users who have a low affinity score, and fair content to users who have a high affinity score. While there are multiple ways of implementing this adaptive policy, we consider two formulations. First, we begin by a simple extreme case formulation wherein we optimize for relevance for users with negative affinity scores, and optimize for fairness for users with a positive score. Specifically, we define Adaptive - I as:

$$s_u^* = \begin{cases} \text{argmax}_{s \in S_u} \psi(s) & \text{if } \xi_u \geq 0 \\ \text{argmax}_{s \in S} \phi(u, s) & \text{if } \xi_u < 0 \end{cases}$$

Additionally, we consider an adaptive variant of the interpolation policy, which goes beyond positive/negative bifurcation of affinity scores, and uses its exact estimate to select the recommended set. To this end, we normalize the user affinity scores across all users using the standard deviation of its distribution, and re-weight the score of each candidate set treating the normalized affinity score as a weighting parameter. Specifically, we define Adaptive - II as:

$$s_u^* = \text{argmax}_{s \in S_u} \left( (1 - \hat{\xi}_u) \phi(u, s) + \frac{\hat{\xi}_u}{\sigma} \psi(s) \right)$$

with $\hat{\xi}_u$ denoting the standard deviation normalized value of the user affinity estimate.

While the above mentioned policies allow us to trade-off relevance of a set to a customer and fairness of a set towards different suppliers, the goal of this paper is to investigate their impact on user satisfaction. While relevance and fairness are easier to compute offline, their impact on user satisfaction is indeed hard to measure since most metrics that estimate satisfaction depend on user feedback, and are hard to estimate offline. The next section outlines a counterfactual estimation approach to compute user satisfaction metrics.

6 UNBIASED ESTIMATION OF USER SATISFACTION

Estimating user satisfaction is a non-trivial problem, with most solutions based on running a controlled experiment (i.e. an A/B test). However, these experiments usually require non-trivial engineering resources and are time-consuming. Furthermore, they impact live users, and can have a detrimental effect on user retention and experience. To this end, we adopt the use of counterfactual estimation techniques for unbiased offline evaluation of user satisfaction in recommendation setting. Compared to A/B tests, offline evaluation allows multiple models to be evaluated on the same log, without the need to be run online. Effectively, this technique makes it possible to run many A/B tests simultaneously, leading to substantial increase in experimentation agility. Indeed, trying various recommendation policies around trading-off relevance to achieve higher fairness can have a negative impact on user experience, and thus, the ability to make offline estimates of user satisfaction is strongly desired. To the best of our knowledge, this work is among the first to advocate the use of counterfactual techniques as a subroutine for evaluation of trade-off policies in recommendation settings.

6.1 Unbiased Estimator

Most satisfaction metrics are computed from user feedback, which in case of contextual bandits is embodied as the reward signals observed. An important observation in contextual bandits is that, only rewards of chosen actions are observed. For offline policy evaluation, such partial observability raises a related difficulty. Data in a contextual bandit is often in the form of $(x, a, r_a)$, where $a$ is the action chosen for context $x$ when collecting the data, and $r_a$ is the corresponding reward. If this data is used to evaluate a policy $\pi$, which chooses a different action $\pi(x) \neq a$, then we simply do not have the reward signal to evaluate the policy in that context.

In this section, the recommender system is modeled as a stochastic policy $\pi$ that specifies a conditional distribution $\pi(a|x)$ (a deterministic policy is a special case). The value of a policy $\pi$, denoted $V(\pi)$, is defined as the expected reward when following $\pi$:

$$V(\pi) = \mathbb{E}_{x \sim D} \mathbb{E}_{a \sim \pi(.|x)} \mathbb{E}_{r \sim D(., x, a)} [r]$$

To compute user satisfaction metrics, we rely on off-policy evaluation under the contextual bandit setting described above. Off-policy evaluation enables estimation of the value of a new policy $\pi$, called the target policy, using the logged data. For the logged data, we rely on collecting randomized data which has been shown to be crucial for drawing valid causal conclusions [22]. To collect the randomized data, for each user interaction, we observe the context $x \sim D(x)$, and a random action $a \in A$ is drawn according to a uniform distribution over the action space, and the corresponding reward $r_a$ and probability mass $p_a$ are logged. The probabilities $p_a$ are usually called propensity scores and the data thus collected exploration data, owing to the exploration of all actions with a non-zero probability. This process helps us in collecting a dataset of the form $(x, a, r_a, p_a)$.

Using the exploration data, we rely on the inverse propensity score (IPS) estimator [12], which is provably unbiased. The IPS estimator re-weights the exploration data according to ratios of action probabilities under the target and exploration policy. The target policies are the different recommendation policies proposed in Sections 4 & 5. To evaluate a target policy (say $\pi$) using the logged exploration data, we define the IPS estimator as follows:

$$\hat{V}_{\text{offline}}(\pi) = \sum_{x, a, r_a, p_a} r_a \frac{\pi(x) = a}{p_a}$$

where $\mathbb{1}(\pi(x) = a)$ is the set indicator function which evaluates to 1 if the action selected by the target policy matches the exploration data collection policy, else it evaluates to 0.
The key observation of the estimator is that, for any context \( x \), if one chooses action \( a \) randomly according to the uniform distribution, then the user satisfaction metric can be computed as:

\[
    r_\pi(x) = \mathbb{E}_a \left[ \frac{r_a(\pi(x) = a)}{p_a} \right]
\]

With this equality, one can show the unbiasedness of the offline estimator [15]:

\[
    \mathbb{E} \left[ \hat{V}_{offline}(\pi) \right] = V(\pi)
\]

for any \( \pi \) provided that every \( p_a \) is non-zero. In other words, as long as we can randomize action selection, we can construct an unbiased estimate for any policy without even running it on users. This benefit is highly desirable, since the offline evaluator allows one to simulate many A/B tests in a fast and inexpensive way, and compute user satisfaction estimates.

### 6.2 Verification of Randomized Data & Propensity Scores

An important prerequisite for the unbiasedness guarantee of the IPS estimator is that the propensities scores are all non-zero, i.e. \( p_a > 0 \ \forall a \). We employ a uniform distribution over the action space; thus, for each user, we select a set to recommend in uniformly random fashion from among the pool of sets pre-selected for the user. Thus, \( p_a = \frac{1}{K_u} \), where \( K_u \) is the number of sets in the pool for the user \( u \).

To verify the propensity scores, we perform two simple tests:

1. **Arithmetic Mean Test**: we compare the number of times a particular action \( a \) appears in the data to the expected number of occurrences conditioned on the logged propensity scores. We noticed that the gap is not significant, which indicates no errors in the randomized data collection process.

2. **Harmonic Mean Test**: Following insights from Li et al. [14], we confirm the following equality:

\[
    \mathbb{E}_a \left[ \frac{1}{p_a} + \frac{1}{1 - p_a} \right] = 2
\]

We compared the mean of the above random variable from the data, and verified its closeness to the expected value, 2.

The above checks confirm sanity of the random exploration data we collected, which enables us to trust the unbiasedness of the user satisfaction estimates.

### 7 EMPIRICAL EVALUATION

We present detailed results on how relevance and fairness impact user satisfaction.

#### 7.1 Dataset

We use logged feedback data and live production traffic from an online music streaming service to evaluate different recommendation policies. The exploration data gathering AB test was run for 2 weeks period in November 2017, wherein we collected data for a random collection of over 400K users, their interactions with over 5000 sets (playlists). In total, the dataset comprised of tracks from a total of over 49K artists, which gives us a good mix of both consumers (users) and suppliers (artists).

<table>
<thead>
<tr>
<th>Recommendation Policy</th>
<th>User Satisfaction Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Fairness</td>
<td>0.420</td>
</tr>
<tr>
<td>Only Relevance</td>
<td>0.650</td>
</tr>
<tr>
<td>Adaptive - I</td>
<td>0.709</td>
</tr>
<tr>
<td>Adaptive - II</td>
<td>0.729</td>
</tr>
</tbody>
</table>

**Table 1**: User satisfaction estimates for a subset of recommendation policies considered.

#### 7.2 Comparing Different Trade-off Policies

We begin by considering the first two recommendation policies: Optimizing Relevant, and optimizing Fairness, and present user satisfaction estimates for the two policies in Table 1. The results confirm our initial hypothesis: recommending relevant content has a positive impact on customer satisfaction, while recommending fair content harms user satisfaction. We observe that a relative decline of 35% in satisfaction on moving from a focus on relevance to a focus on fairness. These initial results further motivate the need for interpolating between relevance and fairness.

We next evaluate the interpolated Fairness and Relevance policy presented in Section 4.3. We vary the \( \beta \) parameter from 0 to 1, to differentially weight relevance and fairness estimates for each set and present the impact on user satisfaction for all such interpolated cases in Figure 3 (left). We observe a gradual improvement in user satisfaction metric when we move from \( \beta = 0 \) to \( \beta = 1 \), with the lowest user satisfaction estimate of 0.420 on the most fair policy, which increases by ~ 10% to 0.46 for a policy which equally weights relevance and fairness when computing the final score for each recommended set. Further, we observe a sharp increase in user satisfaction when we increase the importance given to relevance to 0.7 and beyond. A net gain in user satisfaction of ~ 40% is observe over the most fair policy with \( \beta \approx 0.7 \). Finally, we do not observe significant improvements in user satisfaction beyond \( \beta \approx 0.8 \), which suggests that recommender systems could easily give 20% importance to fairness of sets without severe impact on user satisfaction.

We observe a similar incremental trend and range of user satisfaction estimates for the probabilistic policy of recommendation presented in Section 4.4. The probabilistic policy makes a probabilistic choice based on the value of the \( \beta \) parameter to either recommend a set based on its fairness score or relevance score. As shown in Figure 3 (middle), we observe a linear increase in satisfaction as we enable the algorithm to pick the most relevant set to be recommended. Different from the interpolated policy, we observe a slightly lower estimate of user satisfaction for the probabilistic policy, which is expected since the interpolated policy values relevance in a deterministic manner, while in this policy, even for higher values of \( \beta \) (say) 0.9, the system may decide to show fair content in 10% of cases, which would bring down the user satisfaction estimate.

#### 7.3 Guaranteeing Relevance

Having guarantees on the relevance of recommended sets to users is often desirable for system designers as it ensures that user experience is not substantially degraded. To this end, the recommendation policy proposed in Section 4.5 optimizes for fairness with guarantees on relevance. Figure 3 (right) presents user satisfaction...
Given the trade-off between relevance and fairness and their impact on user satisfaction in the results presented so far, we take a holistic look at the cost-benefit offered by the different recommendation policies. Table 3 presents a detailed analysis of the percentage gain in fairness, relevance and satisfaction observed for the different policies. To compute the percentage loss in fairness for any policy, we compare the mean fairness of the sets recommended by that policy, and compute how much it differs from the best achievable fairness estimate, i.e., we compute its percentage different with respect to the satisfaction score obtained in the Optimizing Relevance policy (Section 4.2). Finally, we compute the gain in satisfaction by computing the difference with respect to the satisfaction score obtained in the Optimizing Relevance policy (Section 4.2).

The results presented in Table 3 enable us to select cases wherein the loss observed in fairness and relevance is minimized while maximizing the gains in satisfaction. While the interpolated policy suffers from low loss in relevance and satisfaction, it severely impacts fairness with losses ranging from 42 to 64%. For the probabilistic policy, we observe balancing fairness and relevance with \( \beta = 0.7 \).
We present a computational framework to understand the interplay between consumer relevance, supplier fairness and present a counterfactual estimation framework to estimate their impact on consumer satisfaction. We propose a number of policies to jointly optimize for fairness and relevance. We conjecture that appropriate definitions of fairness, and satisfaction in different marketplaces would enable system designers to better understand the interplay between the different factors. In future, we intend to consider variance controlled offline estimators and conduct A/B tests to validate our findings. We also envision future research towards developing and evaluating sophisticated recommendation techniques for joint optimization of fairness, relevance and satisfaction.

8 CONCLUSION

We present a computational framework to understand the interplay between consumer relevance, supplier fairness and present a counterfactual estimation framework to estimate their impact on consumer satisfaction. We propose a number of policies to jointly optimize for fairness and relevance. We conjecture that appropriate definitions of fairness, and satisfaction in different marketplaces would enable system designers to better understand the interplay between the different factors. In future, we intend to consider variance controlled offline estimators and conduct A/B tests to validate our findings. We also envision future research towards developing and evaluating sophisticated recommendation techniques for joint optimization of fairness, relevance and satisfaction.

REFERENCES