Quantification of the Impact of Popularity Bias in Multi-stakeholder and Time-Aware Environments

Francisco Guíñez(✉), Javier Ruiz, and María Ignacia Sánchez

Pontificia Universidad Católica, Santiago, Chile
{fhguinez,jiruiz2,mcsanchez}@uc.cl

Abstract. Popularity bias is one of the main biases present in recommendation algorithms, which means most popular items are over-recommended by the algorithms, while items with less interactions are invisible. In this paper we analyze the impact of the popularity bias over time for different stakeholders considering different recommendation algorithms. The datasets used were Last.FM-1B and KASANDR, while the algorithms compared were ALS and BPR, with Most Popular and random recommendations used as a baseline. The main contribution of this paper provides a new way of measuring popularity in a variable way over time and analyzes which algorithms maintain a popularity bias over time that negatively impacts stakeholders. The analysis of the unfairness metrics shows that the popularity bias is not static over time and that the ALS algorithm obtains more stable results with less unfairness than BPR among different groups of stakeholders.

Keywords: Popularity bias · Popularity bias over time · Stakeholders · Multi-stakeholders · Time-aware

1 Introduction

Popularity bias is one of the main biases present in recommendation algorithms [1]. This consists in the fact that the most popular items are over-recommended by the algorithms, while items with less interactions are invisible [19]. This generates a rich-get-richer effect [24], in which recommendations increase the popularity of already popular items and do not give the less popular ones a chance to emerge. In general, efforts to decrease this bias have been placed on those who consume these recommendations, i.e., the users. However, in platforms where there are multiple stakeholder groups, it is important to consider what impact each one has, otherwise some of these groups may have incentives to stop using the platform. In order to make good and useful recommendations to users, it is important that the recommender system takes into consideration the novelty and diversity of these recommendations [13]. For example, if Netflix made only popular recommendations due to the bias of its algorithm, only the directors of the most popular movies would win. In turn, those who would be most interested

L. Boratto et al. (Eds.): BIAS 2021, CCIS 1418, pp. 78–91, 2021.
https://doi.org/10.1007/978-3-030-78818-6_8

in the platform would be those users who have more popular tastes. This would hurt those directors and users who create/consume less popular movies, which would eventually give them incentives to leave the platform.

Within this context, it is also important to consider the time in which an item is popular as a variable. That is, items that are popular today may not be popular tomorrow. Thus, users who at one time had popular tastes may eventually become users with less common tastes; and similarly, suppliers may go from being unpopular to becoming popular. While metrics have been constructed to measure the impact of popularity bias, they have not considered the time dimension of interactions.

Thus, the main contributions of this paper are:

1. Propose a way to measure the popularity of the items but considering the dynamics of time. This will also allow us to catalogue stakeholders in dynamic groups of popularity over time.
2. Perform a popularity bias analysis over time, considering the state of the art metrics on the subject, which measure the impact of the popularity bias on users and providers.

2 Related Work

If we examine the related work, it is possible to identify three main areas of research: Recommendations for multiple stakeholders, analysis of time-aware recommendations, and the presence of bias and injustice in multiple contexts.

2.1 Multistakeholders

As described above, a stakeholder is defined as a person or group of people interested in a business who are directly affected by the decisions made in it. Thus, within the context of recommendation, considering multiple stakeholders implies making recommendations taking into account the differentiated interests between the different groups and considering the needs of each one [14]. The research in this subject arises after the academy has historically focused on the interests of the end user when proposing recommending models, but the complexity of these made it necessary to recognize that they serve multiple objectives [2,15], various models have been proposed in this direction in the last decade [23,26,32,37].

Specifically, an interesting multiple approach is presented in [1], where, based on a music dataset, the quality of recommendations for both end users and composing artists are considered, in addition to subdividing these groups according to their popularity in high, medium and low.

This field of research is related to the search for new and more diverse evaluation metrics for recommending systems, because of the need to evaluate groups as well as individuals [22].

2.2 Time-Aware Recommendations

Over the years, this topic has been approached from two main perspectives, which are independent and complementary [18]. On the one hand, we have the notion that the popularity of an item may change over time, and occurs intrinsically in every context, but is also influenced by external events. Examples of this perspective are the prediction of values on a timeline through before and after interactions [36] and the incorporation of time-dependent weights that decrease the importance of old items [6,17].

On the other hand, there is the temporal evolution of the users; the way they interact with their environment is not constant and therefore their tastes or opinions vary depending on the temporal context [12,33]. In particular, this has motivated research for models with a time dimension. Among them, for example: models aware of the day of the week [5], models aware of the time of the day [35] and models where the importance of the interactions is penalized according to how many newer interactions with the same item the user has [25].

2.3 Bias and Unfairness

Finally, it is important to mention that both the presence of multiple stakeholders and the time dimension are sources of bias and injustice when making recommendations [7].

When we recognize stakeholder groups differentiated by popularity, it is common to keep in mind a popularity bias that is unfair to groups describing unpopular segments [4]. In addition, popularity bias has been shown to impact the evaluation of recommender systems [8,21], being important to know the objective of the recommendation to measure the real impact. In [9] various algorithms are compared in a context that needs to consider both precision and diversity of the results, so algorithms that compensate for popularity bias perform well.

There have been efforts to measure this injustice by means of knowledge graphs [16] and considering the efficiency of pareto [34], while other works have analyzed the implications that unfair models have on people [3,20]. Also, [10] proposes two ways of measuring bias, one aimed at giving equal exposure to all items and the other aimed at giving exposure according to the actual popularity of the item.

Likewise, not considering the notions of temporal evolution described in the Sect. 2.2 almost always implies the presence of a time bias that is impossible to measure [33]. Therefore, in the literature we see methods such as the one proposed in [18] which counteracts this bias by means of penalizers proportional to the measured bias, or [11] that uses variational autoencoders.

3 Datasets

In order to carry out the desired study, we used two datasets: 'LFM-1B' [28], which has the reproductions made by users to different songs in Last.FM site, and

KASANDR-train_de [30], which has records of e-commerce sales. Both datasets were pre-processed in order to adapt the amount of information to the available computing capacity. After the pre-processing, we obtained the Last.FM dataset with 316,666 interactions between a user and a specific album. Meanwhile, for KASANDR, we obtained a dataset with 520,196 interactions between a user and a specific item.

The resulting dataset of Last.fm was divided into 6 parts, each of which had interactions up to a certain point in time. The first part had interactions occurring until the end of the sixth month of 2011, the second until the end of the seventh month of 2011, the third until the end of the eighth month of 2011, and so on. Meanwhile for KASANDR, since this dataset only contains data for 14 days, the resulting dataset was divided into 6 parts, each of them separated by two days starting from day 5. A summary of the matrix of interactions associated with both datasets can be seen in the Table 1.

For simplicity, we will call the subdivision of these datasets "periods". In the case of Last.fm each period will be monthly, while in KASANDR each period will be once every two days. For each periodic dataset, a division of 80% was made for training and 20% for testing.

Table 1. Summary of interaction matrices

Last.FM dataset						
Month	7	8	9	10	11	12
# of interactions	124,869	151,657	179,953	209,423	242,800	316,666
# of users	9,573	11,101	12,631	14,093	15,716	19,129
# of suppliers (artists)	18,202	20,799	23,360	25,752	28,355	33,638
# of items (albums)	26,583	31,003	35,385	39,551	44,050	53,636
KASANDR dataset						
Day	5	7	9	11	13	15
# of interactions	184,620	251,395	326,053	390,456	462,196	522,196
# of users	9,526	12,371	15,092	17,345	19,806	24,936
# of suppliers (sellers)	620	634	643	651	659	665
# of items (products)	103,832	132,540	162,656	186,415	213,906	228,537

4 Proposed Method

Intuitively, when we talk about the popularity of a song for example, we consider that it becomes popular when many users start listening to it. Thus, most works in the area consider the number of interactions of an item as a key factor to define its popularity.

On the other hand, from a mathematical perspective, when we are facing a continuous change in time, it is natural to model the problem as a derivative, understood as the ratio of instantaneous change of a variable [31]. Common examples of the application of the derivative concept are the change of motion in time, which translates into speed, and the change of speed in time, which translates into acceleration.

Having said this and considering the state of the art, given that the objective of this work is to measure the popularity bias considering the existence of a time variable, we propose a new way of measuring the popularity of an item, which interprets popularity as the ratio of instantaneous change related to the number of interactions that an item achieves in time. Having this popularity value for each item, we will proceed to classify the set of items into three subgroups of items: populars ($\mathcal{H}^t$), moderately popular ($\mathcal{M}^t$) and of low popularity ($\mathcal{T}^t$). A similar procedure will also be carried out to group users and suppliers as stakeholders.

4.1 Time Popularity Metrics

We will define N_i^t as the number of interactions achieved by the item i until a period of time t. For example, for an item i, $N_i^{2018} = 1000$ will imply that until 2018 the item has achieved a number of 1000 interactions. This variable is discrete, and easily obtainable through the datasets. In our case (and as described in Sect. 3) for Last.FM dataset $t \in \{7, \ldots, 12\}$ are monthly periods from July to December in 2011. For KASANDR dataset $t \in \{5, 7, 9, 11, 13, 15\}$ are daily periods from June 2016.

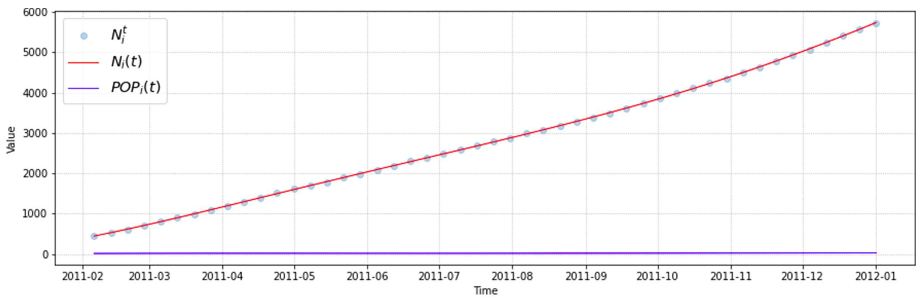
Our interest will be in approximating N_i^t with a soft function $N_i(t)$ such that $N_i(t) = N_i^t$ for all period t. There are many methods of approximation of functions with different degrees of accuracy, however, since the main focus of this project is an analysis at the bias level, a polynomial adjustment was chosen, which proves to be effective and fast without sacrificing too much accuracy. We emphasize the importance of maintaining a low computational cost in this work, since in certain cases it is necessary to adjust large amounts of data. For this purpose, the library *statsmodels* was used [29], which contains what is needed to create a non-linear model based on dataframes in a simple way, and allows for performing a polynomial regression using least squares. Another benefit of this library is that it allows for adjusting the degree d of the polynomial with which the regression is sought, which will allow to obtain coefficients to model different variants of the function when required. In this case, it was considered that $d = 3$ gave an error small enough for this context.

Once the coefficients of this regression are obtained, the function $N_i(t)$ adjusts to the variable N_i^t with an error of ε_i, which will be used to build the item popularity function:

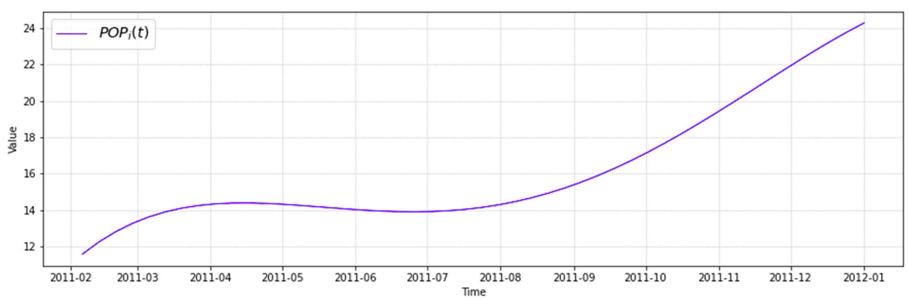
$$\textsc{Pop}_i(t) = \frac{\partial}{\partial t} N_i(t) \tag{1}$$

This popularity function will allow to calculate the popularity of an item at any instant of time, since the popularity is represented as the ratio of instantaneous change related to the number of interactions that an item achieves in time.

To illustrate this, we will take as example the album *21* from Adele, which was released on January 24, 2011. Figure 1a shows the discrete acumulated interactions N_i^t and its aproximation $N_i(t)$ using $d = 5$ to illustrate. On the other side, Fig. 1b shows the popularity of this item in time. Altough the number of interactions increase monotonely in time (Fig. 1a) from which we could say that the popularity of this item is constant, Fig. 1b shows that this is not necessarily true since the popularity changes over time. Every album has variations in the speed of growth of its interactions, no matter how small they are. Our popularity function detects these variations, which can be seen in Fig. 1b with a zoomed Y axis. The better the adjustment of $N_i(t)$, the better modeled the popularity function will be.



(a)



(b)

Fig. 1. N_i^t, $N_i(t)$ and $\text{P}\textsc{op}_i(t)$ illustration for the album item *21* from Adele.

It should be noted that, depending on the dataset, other measures of popularity could be proposed. For example, instead of considering the number of interactions accumulated by an item N_i^t, you could consider the number of user interactions achieved by this item B_i^t. That is, for each user, a 1 is added to the variable B_i^t if an interaction was made with the item i until the period t. Other approaches could be to divide these variables by the number of total users, so as to capture the growth of the item within a specific context. All these variants were implemented and tested, but their results were similar to (1), so the latter was finally chosen given its low computational time compared to the others.

Sequential recommendations will be made. In particular, recommendations will be generated in each period using as periods those defined in Sect. 3.

4.2 Dynamic Grouping of Items and Stakeholders

Once these popularity functions $\textsc{Pop}_i(t)$ for all i items have been obtained, the dynamic grouping of items and stakeholders (users and suppliers) is carried out. To make this grouping in the items, as in [1], we used the Pareto Principle [27]. Given a period of time t, the items are ordered from most to least popular according to $\textsc{Pop}_i(t)$. The first 20% of these items will represent group $\mathcal{H}^t$. The last 20% of the items will belong to the $\mathcal{T}^t$ group. The remaining items will belong to $\mathcal{M}^t$.

With regard to the grouping of users and suppliers, we will proceed to calculate the simple average of popularity of the items that have been listened to or created, respectively. A similar procedure will then be carried out for grouping, using the same percentages of cuts as for item grouping. Let $\mathrm{W}_u(t)$ be the average level of popularity interactions by the user u over time t, and $\mathrm{P}_a(t)$ be the popularity of the supplier a over time t, then:

$$\mathrm{W}_u(t) = \frac{\sum_{i \in E_u^t} \textsc{Pop}_i(t)}{|E_u^t|} \quad (2) \qquad \mathrm{P}_a(t) = \frac{\sum_{i \in C_a^t} \textsc{Pop}_i(t)}{|C_a^t|} \quad (3)$$

Where E_u^t are the items that the user u interacted with until time t, and C_a^t are the items offered by the supplier a until time t. We will call the groups derived from this procedure as $\mathcal{U}_1^t$, $\mathcal{U}_2^t$, $\mathcal{U}_3^t$ for the users and $\mathcal{A}_1^t$, $\mathcal{A}_2^t$, $\mathcal{A}_3^t$ for the suppliers, where the larger the number, the more popular the group.

4.3 Measuring the Unfairness of Stakeholder Recommendations

In order to measure the impact of the recommendations on the different stakeholders, the Item Popularity Deviation (IPD), Supplier Popularity Deviation (SPD) and User Popularity Deviation (UPD) measures will be taken, as proposed by [1][1], but adapted to a version that considers temporality. On the other hand, to measure the coverage of items that do not belong to the popular group, the well-known metrics of Aggregate Diversity (Agg-Div) and Long Tail Coverage (LC) will be used, in addition to the metric Average Percentage of Long-tail Items (APL) proposed by [1]. That being said, the following variables are defined:

- ℓ_u^t: List of item recommendations delivered by an algorithm to a user u in time t.
- L^t: Set of recommended items in the time period t.

[1] With respect to UPD, a small modification in the way it is calculated will be considered, but it follows the same idea proposed by [1].

- $V(i)$: Function that returns the supplier of the item i
- $\mathcal{U}^t = \{\mathcal{U}_1^t, \mathcal{U}_2^t, \mathcal{U}_3^t\}$: Set with the popularity groups of users for a time t.
- U^t: Group of all users on the platform, up to the time t.
- $\mathcal{A}^t = \{\mathcal{A}_1^t, \mathcal{A}_2^t, \mathcal{A}_3^t\}$: Sets with the popularity groups of suppliers for a time t.
- $\mathcal{I}^t = \{\mathcal{H}^t, \mathcal{M}^t, \mathcal{T}^t\}$: Set with the popularity groups of the items for a time t.
- E_u^t: List of items that user u interacted with before time t.
- n: Number of recommendations.

$$APL = \frac{1}{|U^t|} \sum_{u \in U^t} \frac{|\{i, i \in (\ell_u^t \cap (\mathcal{M}^t \cup \mathcal{T}^t))\}|}{|\ell_u^t|} \quad (4)$$

$$UPD = \frac{\sum_{g \in \mathcal{U}^t} |UPD(g)|}{|\mathcal{U}^t|} \quad (6)$$

$$SPD = \frac{\sum_{s \in \mathcal{A}^t} |SPD(s)|}{|\mathcal{A}^t|} \quad (5)$$

$$IPD = \frac{\sum_{c \in \mathcal{I}^t} |IPD(c)|}{|\mathcal{I}^t|} \quad (7)$$

Here, we calculate $SPD(s)$ and $IPD(c)$ as proposed in [1], with the difference that the popularity groups, instead of being defined statically by the number of interactions, were defined according to the proposed new popularity metric, which considers a time-varying subdivision. In addition, we also considered the variables associated with the recommendations given to a user and the interactions of a user in a variable way in time. On the other hand, we considered a slight variation of the formula for $UPD(g)$ with respect to what was proposed by [1], but it was decided to maintain the same idea proposed to calculate SPD and IPD and to average over the user popularity groups the subtraction between the proportion of recommendations achieved by a group and the proportion of interactions achieved by that same group, that is:

$$UPD(g) = qu(g) - pu(g) \quad (8)$$

$$qu(g) = \frac{\sum_{u \in U^t} \sum_{j \in \ell_u^t} \mathbb{1}(V(j) \in s)}{n \times |U^t|} \quad (9)$$

$$pu(g) = \frac{\sum_{u \in U^t} \sum_{j \in E_u^t} \mathbb{1}(V(j) \in s)}{|E_u^t|} \quad (10)$$

For these last three metrics, lower values mean that there is a smaller average difference between the proportion of the recommended and the proportion of the actual interactions achieved per popularity group, so the algorithm would be fair to the different popularity groups.

5 Experimented Recommender Systems

In order to find an ideal configuration for the recommendation algorithms and to enable them to do their job better, different combinations of values for their hyperparameters were tested on a cross validation of four folds.

The hyperparameters studied, both for ALS and BPR, were the number of latent factors (50, 100 and 200) and the regularization parameter (0.01 and 0.1). In addition, the learning rate was varied for BPR (0.001, 0.01 and 0.1).

First, for Last.FM dataset, the decision was made to make 5 recommendations per user, since a smaller number does not allow us to adequately analyze the capacity of the algorithms to recommend less popular items due to most popular options monopolize all the recommendations. On the other hand, a higher number of recommendations would not be representative of the context to be studied, since very few users have interacted with more than 5 different items, adding noise to the metrics.

Second, in the case of KASANDR dataset, we decided to make 5 recommendations also based on what was said in [30].

Then, for each specific hyperparameter configuration, MAP@5 and nDCG@5 were calculated for each period. Finally, the average of the metrics for each set of parameters was obtained in order to select the one that delivers a higher value.

For Last.FM dataset, the chosen parameters for ALS were 50 latent factors and 0.1 as a regularization parameter. With this configuration higher values were obtained in both MAP@5 and nDCG@5. Meanwhile for BPR, the chosen parameters were 100 latent factors, 0.01 as a regularization parameter and 0.01 as a learning rate parameter.

For KASANDR dataset, the chosen parameters for ALS were 200 latent factors and 0.1 as a regularization parameter, with higher values obtained in MAP@5 and nDCG@5. Meanwhile for BPR, the chosen parameters were 200 latent factors, 0.01 as a regularization parameter and 0.001 as a learning rate parameter.

6 Results

Once the best hyperparameter configurations were defined, recommendations were made for each user in each monthly (Last.FM) and daily (KASANDR) dataset, resulting in six groups of recommendations for each one. Then, for these recommendations the unfairness was measured using the metrics described in Sect. 4.3. In addition, recommendations were made with the Most Popular and Random algorithms to have a baseline of values for the studied metrics, since the first algorithm should deliver the highest values of popularity bias, while the second should illustrate a moderate unfairness. It is important to note that Random should not have completely fair values, since it negatively affects the recommendations of the most popular items. The results obtained can be reviewed in Fig. 2 and 3.

As displayed in both Figures, except for Agg-div and LC, which tend to be more linear, all the metrics have uneven variations as they move forward in time, showing that the bias is not static, that is, it can both increase and decrease over time.

In addition, the ALS recommendation algorithm manages to overcome BPR in Agg-div, LC, IPD, SPD and UPD metrics in most of the epochs for both

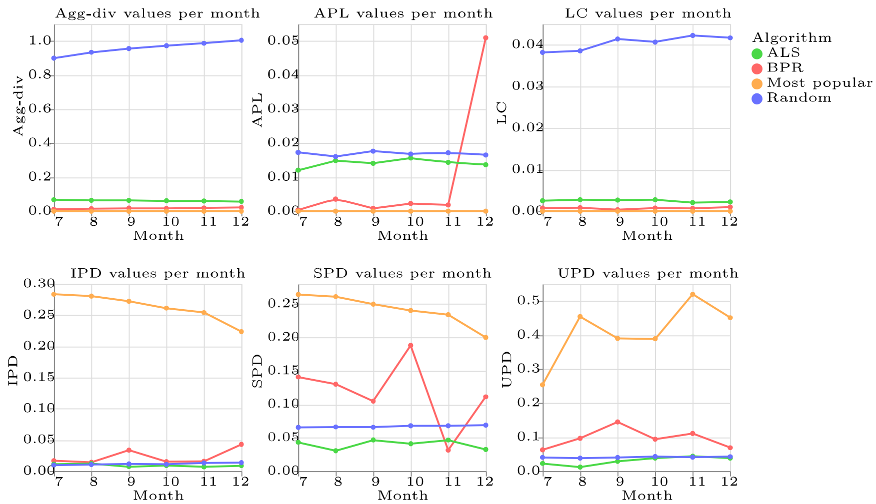


Fig. 2. Unfairness metric results for Last.FM dataset

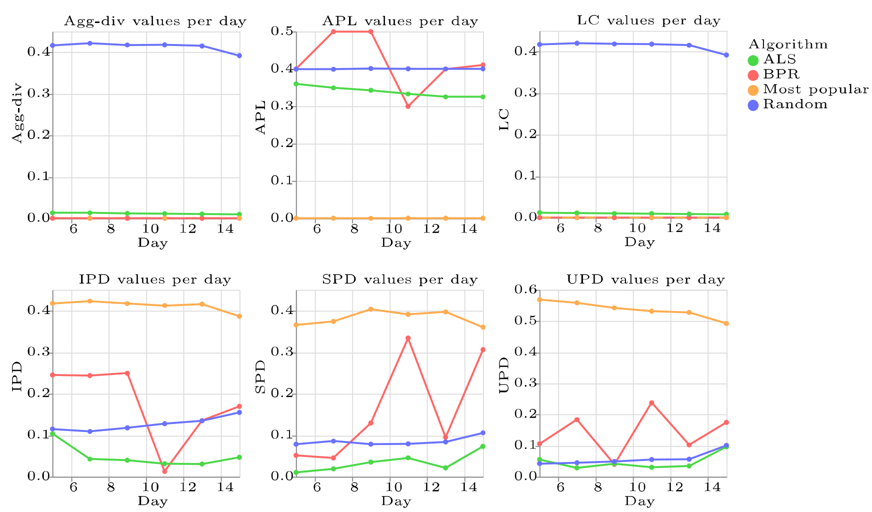


Fig. 3. Unfairness metric results for KASANDR dataset

dataset. Moreover, for APL metric, the best algorithm depends on the epoch and the dataset. This performance of ALS means that it is less unfair than BPR when recommending unpopular items.

Another aspect worth noting is that, when it comes to IPD, SPD and UPD, which are metrics that measure unfairness, both algorithms perform much better than the Most Popular algorithm, which naturally turns out to be very unfair in prioritizing popular items.

It is interesting to note that ALS is the algorithm that manages to maintain better metrics over time with respect to Agg-Div, LC, IPD, APL and SPD. This means that, in general, ALS manages to give higher priority in recommendations to less popular items compared to BPR from a time-aware perspective.

With respect to the unfairness of recommendations among popularity groups, the UPD, SPD and IPD metrics show fairly constant values over time in the case of ALS, which means that unfairness is maintained constantly. On the other hand, these metrics have non-linear variations from one month to another for BPR, which shows that the injustice of this algorithm may vary over time.

7 Conclusions and Future Work

As the objective of this work was to explore and make a first approach to the analysis of popularity bias in algorithms considering a temporal dimension, it would be interesting to address the following tasks in the future:

- Due to the computational demands of the number of items, we decided to abstract the problem to monthly and daily time points, which translates into an adjustment error that could be important. Thus, in the future the adjustment of each function $N_i(t)$ could be better and more accurate. For example, a greater number of points could be considered to obtain a better adjustment, or an adjustment error of $\varepsilon_i = \varepsilon \ \forall i$ could be set and the number of degrees of the polynomial d_i could be increased until achieving a value less than or equal to this error.
- Given that the focus was on temporal analysis of popularity bias, temporary recommendations were made using a basic algorithm. In the future, more sophisticated and less costly methods could be used in making them. Furthermore, it would be interesting to analyze how other algorithms besides ALS and BPR behave with respect to their temporal popularity bias.
- On the other hand, it would be interesting to analyze in detail the possibilities provided by the item popularity metric based on the derivative. By pre-computing the popularity of the items over time, it is possible to know at any time which items belong to the groups $\mathcal{H}^t$, $\mathcal{M}^t$ and $\mathcal{T}^t$. With this it is possible to build an algorithm that allows rewarding recommendations from less popular groups. Naturally, in order to control this benefit, an adjustment parameter α would be required, which would allow weighting the importance given to this aspect. It would be interesting to analyze how such a metric behaves in contrast to conventional metrics to measure coverage and variability of recommendations.

The results presented in Sect. 6 demonstrated that the popularity bias is not static in time. This highlights the need to build time-conscious recommendations, since timeless analyses do not give a complete picture of the problem to be addressed.

A dataset of musical interactions and e-commerce interactions were used as object of study, with which recommendations were made considering a sequence of instants in time. With the results obtained, we concluded that ALS is less unfair than BPR when recommending unpopular items, since ALS is able to maintain lower and more consistent metrics of injustice over time.

The main difficulties arose from the high computational cost of estimating the popularity functions for each item, which was overcome by subsampling the information. This decision did not greatly affect the results of this research and a similar analysis can be carried out in the future with better hardware.

References

1. Abdollahpouri, H.: Popularity bias in recommendation: a multi-stakeholder perspective. Ph.D. thesis (2020). https://doi.org/10.13140/RG.2.2.18789.63202
2. Abdollahpouri, H., et al.: Multistakeholder recommendation: survey and research directions. User Model. User-Adap. Inter. **30**(1), 127–158 (2020)
3. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The impact of popularity bias on fairness and calibration in recommendation. arXiv e-prints arXiv:1910.05755 (2019)
4. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity bias in recommendation, August 2019
5. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 217–253. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_7
6. Anelli, V.W., Di Noia, T., Di Sciascio, E., Ragone, A., Trotta, J.: Local popularity and time in top-N recommendation. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) ECIR 2019. LNCS, vol. 11437, pp. 861–868. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-15712-8_63
7. Baeza-Yates, R.: Bias in search and recommender systems. In: Fourteenth ACM Conference on Recommender Systems, p. 2 (2020)
8. Bellogín, A., Castells, P., Cantador, I.: Statistical biases in information retrieval metrics for recommender systems. Inf. Retrieval J. **20**(6), 604–634 (2017)
9. Boratto, L., Fenu, G., Marras, M.: The effect of algorithmic bias on recommender systems for massive open online courses. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) ECIR 2019. LNCS, vol. 11437, pp. 457–472. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-15712-8_30
10. Boratto, L., Fenu, G., Marras, M.: Connecting user and item perspectives in popularity debiasing for collaborative recommendation. Inf. Process. Manage. **58**(1) (2021). https://doi.org/10.1016/j.ipm.2020.102387
11. Borges, R., Stefanidis, K.: On mitigating popularity bias in recommendations via variational autoencoders (2021)
12. Campos, P.G., Díez, F., Cantador, I.: Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. User Model. User-Adap. Inter. **24**(1–2), 67–119 (2014)

13. Castells, P., Hurley, N.J., Vargas, S.: Novelty and diversity in recommender systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook, pp. 881–918. Springer, Boston (2015). https://doi.org/10.1007/978-1-4899-7637-6_26
14. Chelliah, M., Zheng, Y., Sarkar, S.: Recommendation for multi-stakeholders and through neural review mining. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 2979–2981 (2019)
15. Ekstrand, M.D., et al.: All the cool kids, how do they fit in?: popularity and demographic biases in recommender evaluation and effectiveness. In: Friedler, S.A., Wilson, C. (eds.) Proceedings of the 1st Conference on Fairness, Accountability and Transparency. Proceedings of Machine Learning Research, vol. 81, pp. 172–186. PMLR, New York, 23–24 February 2018. http://proceedings.mlr.press/v81/ekstrand18b.html
16. Fu, Z., et al.: Fairness-aware explainable recommendation over knowledge graphs. arXiv preprint arXiv:2006.02046 (2020)
17. Garg, D., Gupta, P., Malhotra, P., Vig, L., Shroff, G.: Sequence and time aware neighborhood for session-based recommendations: STAN. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1069–1072 (2019)
18. Koren, Y.: Collaborative filtering with temporal dynamics. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 447–456 (2009)
19. Kowald, D., Schedl, M., Lex, E.: The unfairness of popularity bias in music recommendation: a reproducibility study. In: Jose, J.M., et al. (eds.) ECIR 2020. LNCS, vol. 12036, pp. 35–42. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45442-5_5
20. Liu, W., Guo, J., Sonboli, N., Burke, R., Zhang, S.: Personalized fairness-aware re-ranking for microlending. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp. 467–471 (2019)
21. Mena-Maldonado, E., Cañamares, R., Castells, P., Ren, Y., Sanderson, M.: Agreement and disagreement between true and false-positive metrics in recommender systems evaluation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 841–850 (2020)
22. Morik, M., Singh, A., Hong, J., Joachims, T.: Controlling fairness and bias in dynamic learning-to-rank. arXiv preprint arXiv:2005.14713 (2020)
23. Nguyen, P., Dines, J., Krasnodebski, J.: A multi-objective learning to re-rank approach to optimize online marketplaces for multiple stakeholders. arXiv preprint arXiv:1708.00651 (2017)
24. Nikolov, D., Lalmas, M., Flammini, A., Menczer, F.: Quantifying biases in online information exposure. J. Am. Soc. Inf. Sci. **70**(3), 218–229 (2019). https://doi.org/10.1002/asi.24121
25. Pavlovski, M., et al.: Time-aware user embeddings as a service. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 3194–3202 (2020)
26. Rodriguez, M., Posse, C., Zhang, E.: Multiple objective optimization in recommender systems. In: Proceedings of the Sixth ACM Conference on Recommender Systems, pp. 11–18 (2012)
27. Sanders, R.: The pareto principle: its use and abuse. J. Serv. Mark. **1**, 37–40 (1987). https://doi.org/10.1108/eb024706
28. Schedl, M.: The LFM-1B dataset for music retrieval and recommendation. In: ICMR (2016). https://doi.org/10.1145/2911996.2912004

29. Seabold, S., Perktold, J.: Statsmodels: econometric and statistical modeling with python. In: 9th Python in Science Conference (2010)
30. Sidana, S., Laclau, C., Amini, M.R., Vandelle, G., Bois-Crettez, A.: KASANDR: a large-scale dataset with implicit feedback for recommendation. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1245–1248 (2017)
31. Stewart, J.: Calculus: Early Transcendentals. Cengage Learning (2010)
32. Wang, S., Gong, M., Li, H., Yang, J.: Multi-objective optimization for long tail recommendation. Knowl.-Based Syst. **104**, 145–155 (2016)
33. Xiang, L., Yang, Q.: Time-dependent models in collaborative filtering based recommender system. In: 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology. vol. 1, pp. 450–457. IEEE (2009)
34. Xiao, L., Min, Z., Yongfeng, Z., Zhaoquan, G., Yiqun, L., Shaoping, M.: Fairness-aware group recommendation with pareto-efficiency. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 107–115 (2017)
35. Yuan, Q., Cong, G., Ma, Z., Sun, A., Thalmann, N.: Time-aware point-of-interest recommendation. In: Proceedings of the 36nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 363–372 (2013)
36. Zhang, Y., Zheng, Z., Lyu, M.R.: WSPred: a time-aware personalized qos prediction framework for web services. In: 2011 IEEE 22nd International Symposium on Software Reliability Engineering, pp. 210–219 (2011)
37. Zheng, Y., Pu, A.: Utility-based multi-stakeholder recommendations by multi-objective optimization. In: 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), pp. 128–135. IEEE (2018)