Journal of Ambient Intelligence and Humanized Computing (2023) 14:10849–10865
https://doi.org/10.1007/s12652-022-04356-0

ORIGINAL RESEARCH

Distributional learning in multi-objective optimization of recommender systems

Antonio Candelieri[1] · **Andrea Ponti**[3] · **Ilaria Giordani**[3] · **Anna Bosio**[2] · **Francesco Archetti**[2]

Received: 29 October 2021 / Accepted: 18 July 2022 / Published online: 5 August 2022

Abstract
Metrics such as diversity and novelty have become important, beside accuracy, in the design of Recommender Systems (RSs), in response the increasing users' heterogeneity. Therefore, the design of RSs is now increasingly modelled as a multi-objective optimization problem (MOP) for whose solution Multi-objective evolutionary algorithms (MOEAs) have been increasingly considered. In this paper we focus on the k-top recommendation problem in which a solution is encoded as a matrix whose rows correspond to customers and column to items. The value of accuracy, novelty, and coverage for each candidate list, is evaluated as a sample and can be represented as a 3-d histogram which encodes the knowledge obtained from function evaluations. This enables to map the solution space into a space, whose elements are histograms, structured by the Wasserstein (WST) distance between histograms. The similarity between 2 users in this probabilistic space is given by the Wasserstein distance between their histograms. This enables the construction of the WST graph whose nodes are the users and the weights of the edges are the WST distance between users. The clustering of users takes then place in the WST-graph. In the optimization phase the difference between two top-k lists can be encoded as the WST distance between their 3-dimensional histograms. This enables to derive new selection operators which provide a better diversification (exploration). The new algorithm Multi-objective evolutionary optimization/Wasserstein (MOEA/WST), compared with the benchmark NSGA-II, yields better hypervolume and coverage, in particular at low generation counts.

Keywords Recommender systems · Accuracy · Coverage · Novelty · Wasserstein distance · Multi objective evolutionary optimization

1 Introduction

Recommendation systems are a key component of the toolbox for analysing data from social media. A recent survey (Balaji et al. 2021) reviews the main methods and application areas. The focus of this paper is on Collaborative Filtering (CF) based RSs, in which the basic data structure is a *matrix of ratings,* with as many rows as users and as many columns as items, and each entry is the rating provided by a user (row) to an item (column). Rating matrices are mostly sparse because many entries are unknown: the key assumption is that the unknown ratings are predictable because the known ratings are often highly correlated across various users or items.

The main driver in the development of RSs has been the accuracy of recommendations i.e., the error in the prediction on unknown ratings. Recently the need to recognize the increasing heterogeneity of users' demands has led to multiple metrics such as diversity and novelty which might conflict with each other. Generally speaking, the increase of diversity and novelty might decrease accuracy (Castells et al. 2015). Therefore, we have to manage a trade-off between different objective functions so that we are faced with multi-objective optimization problems (MOP). The best trade off between the objectives can be defined in terms of Pareto optimality. in which solutions are the elements of the Pareto set.

✉ Andrea Ponti
andrea.ponti@oaks.cloud

[1] Department of Economics, Management and Statistics, University of Milano-Bicocca, Milan, Italy

[2] Department of Computer Science, Systems and Communication, University of Milano-Bicocca, Milan, Italy

[3] OAKS Srl, Milan, Italy

We consider Multiobjective evolutionary algorithms (MOEAs): recently some papers, e.g. Li et al. (2016), Lin et al. (2018), have focused also on novelty and diversity.

In this paper we focus on the k-top recommendation problem in which a list is pro posed to each user containing his k-top rated items. A solution is encoded as a matrix whose rows correspond to customers and column to items. The objectives of the optimization are accuracy, novelty, and coverage.

One key novelty of this paper is that a solution is associated not to a scalar value of each objective function, given by the average over users/items, but to the probability distribution of its sample values.

This maps the problem into a space whose elements are the discrete probability distributions which we represent as histograms: this space is structured by a distance between histograms, namely the Wasserstein distance. Measuring the distance between distributions can be accomplished by many alternative models. A general class of distances known as f-divergences are based on the expected value of a convex function of the ratio of the two distributions. If P and Q are two probability distributions over $\mathbb{R}^d$, continuous with respect to each other, and f is a convex function such that $f(0) = 1$ the f-divergenge is given by:

$$D_f(P, Q) = \mathbb{E}_Q f\left(\frac{P}{Q}\right) \tag{1}$$

According to the choice of f the above formula yields specific distances including Kullback–Leibler distance, its symmetrized version Jensen–Shannon, Hellinger distance, total variation divergence and χ-square divergence. The main disadvantage of KL and χ-square distances with respect to Wasserstein is that they do not use information across different bins of the histograms or distributions with different binning schemes, that is different support. WST first introduced in Monge (1781) has received its modern formulation in Kantorovich (1942). WST has a very rich mathematical structure whose complexity and flexibility are analyzed in a landmark volume (Villani 2009) and, in the discrete domain in the tutorial (Solomon et al. 2014). A difficulty with WST is its computational cost which has hampered its diffusion outside the computer vision community. Recently a number of specialized computational approaches have drastically reduced the computational hurdles (Peyré and Cuturi 2019).

1.1 Related works

A related approach has been proposed in Zheng et al. (2017) for the "grey sheep" problem. Users are represented as histograms whose distance is given by their intersections and whose feature by the quantile analysis of each histogram. Ribeiro et al. (2014) analyses multi-objective Pareto efficient approaches considering accuracy of a top-k recommendation list along with novelty and coverage as objective functions. A similar approach is proposed in Li et al. (2016) where one more objective serendipity is considered.

A different approach based on Gaussian Processes has been proposed in Nguyen et al. (2014), Galuzzi et al. (2020). Another approach, also based on Gaussian processes, is proposed in Vanchinathan et al. (2014) which deals explicitly with the explore/exploit dilemma in top-k list selection. Multi armed bandits (MAB) is a classical formalism for studying the exploration/exploitation dilemma when the reward is an unknown pay-off function. Guillou et al. (2015) gathers feedback from users which is used to update the model of users' preferences. Another approach which stresses the interaction with users is Christakopoulou and Banerjee (2018). In Hejazinia et al. (2019) it is shown how different approaches, including matrix factorization, can be embedded in the MAB framework. The MAB framework can also be extended to the case of context dependent bandits (Gentile et al. 2017) and correlated arms (Wang et al. 2018a, b). A general consideration is that while MAB models are very good at capturing the interaction with users and the online learning mechanism, they meet some difficulty dealing with top-k list recommendation and with multi-objective problems.

Multi-objective problems are typically dealt with using evolutionary algorithms; Ribeiro et al. (2014) exploit the pareto efficiency and show that the suggested lists are simultaneously accurate, diverse, and novel. The same objectives are considered in Lin et al. (2018) which show that an extreme point based method can encode the prior knowledge of RSs and enhance the performance of personalized recommendation.

Two recent deterministic approaches are (Gillis et al. 2021) where the issue of distributional robust multi objective optimization is considered and Lin et al. (2019) which considers two objectives CTR (Click Through Rate) and GMV (Gross merchandise Volume) and propose a scalarization approach whose weights are automatically learnt and shown to guarantee Pareto efficiency.

A stream of research, increasingly intertwined with RSs is metric learning. After the early contributions of Weinberger and Saul (2009) metric learning has been suggested moving from the observation that the output of MF methods violates the triangle inequality between inter user distances.

The literature on WST distance is extremely large. Here we quote only few very recent papers focused on the design of recommender systems (RSs). Meng et al. (2020) shows that embedding the RS into a metric space endowed with WST distance enables an effective solution to the item cold-start recommendation. Zhao et al. (2021) propose a Wasserstein based Correlation Analysis for Cross-Domain Recommendation. The use of autoencoders and generative adversarial networks (GAN) for collaborative filtering has been recently proposed in Zhang et al. (2021), Li et al. (2020).

Wasserstein has been also proposed in the context of metric learning (Ma et al. 2020; Rakotomamonjy et al. 2018) to measure uncertainty, embeds user/item representation in a low dimensional space and comply with the triangle inequality).

Our contributions

- The proposal of a probabilistic space in which both the data model and the optimization algorithm are embedded. The elements of this space are discrete probability distributions, specifically histograms.
- A distributional representation of the rating matrix. To each user one can associate a vector whose components are the distances according to a similarity measure to all other users. This vector is synthetized as a histogram with N bins, where $N \ll m$. The value associated to each bin is the number of users whose distance from the target user falls in that bin. The histogram might be regarded as the signature of a user and the weights as user's features.
- The distance between 2 users in the probabilistic space is given by the Wasserstein distance between their histograms. Another important result is that the WST based distributional representation enables the construction of a new graph: called the WST graph whose nodes are the users and the weights of the edges are the WST distance between users. The clustering of users takes then place in the WST-graph.
- The value of each objective (accuracy, novelty, and coverage), for each candidate top k-list, is evaluated as a sample and can be represented as a histogram. This multidimensional histogram encodes the knowledge obtained from function evaluations. The difference between two lists can be encoded as the WST distance between their histograms. This enables to define a WST based selection operators in MOEAs.
- The resulting algorithm Multi-objective Evolutionary Optimization/Wasserstein (MOEA/WST) is run on each cluster and compared with the algorithmic benchmark NSGA-II. On a standard benchmark data set for RSs MOEA/WST results in better hypervolume and coverage, in particular at low generation counts.

1.2 Organization of the paper

Section 2 contains the formal definition of the problem of the ranking matrix and the basic notions of multi-objective optimization. Section 3 contains the basic definitions of the Wasserstein distance and its main computational methods. Section 4 introduces the distributional representation of users, the graph representation of the rating matrix, and the results of different clustering methods. Section 5 is devoted to the distributional representation of the objective functions. Section 6 describes the encoding of solutions and their distributional representations. Section 7 contains the description of MOEA/WST and in particular of the new genetic operators. Section 8 describes the software resources. Section 9 the computational results in terms of hypervolume and coverage of the Pareto approximation, to compare MOEA/WST and NSGA-II. Section 10 contains the conclusions and perspectives.

2 The problem definition and background information

2.1 The problem definition

The basic notion is:

- The set of users $U = \{u_i\}_{i=1,\dots,M}$, where M is the number of users.
- The set of items $O = \{o_j\}_{j=1,\dots,N}$,, where N is the number of items.

Each user expresses its judgement, or rating, $r \in X$, where typical rating values can be binary or integers from a given range. The set of all the ratings given by the users on the items can be represented as a partially specified matrix $R \in \mathbb{R}^{N\times M}$, where its entries r_{ij} express the possible ratings of user u_i for item o_j. Usually, each user rates only a small number of items, thus the matrix R is sparse.

Two types of Collaborative Filtering (CF) methods are commonly used to solve it: the memory-based methods and model-based methods. The memory-based methods, or neighbourhood-based algorithms are based on the fact that similar users display similar patterns of rating behaviour (user-based) or similar items receive similar ratings (item-based). These methods rely usually on k-nearest neighborus (kNN) algorithms, are simple to implement, and yield recommendations easy to explain. Clearly the choice of the distance and the value k can be considered as hyperparameters and their values impact substantially the performance of the method.

The model-based methods, since the Netflix contest, use mostly matrix factorization where users and items can be represented by latent factors in a low dimensional space. Many variants of MF have been recently proposed and MF has become a de-facto cornerstone in the construction of RS. The focus of this paper is not on matrix filling but on the multi-objective optimization problem in k-top recommendation.

An important evolution of MF is given in Indyk et al. (2019) which propose a learning based approach and Wang et al. (2018a, b) and Zhang et al. (2018).

We used the basic implementation of k-NN in Surprise with the cosine similarity.

2.2 Multi-objective optimization

Multiobjective optimization problem (MOP) can be stated as follows:

$$\min_{x \in \Omega \subseteq \mathbb{R}^d} F(x) = \left(f_1(x), \dots, f_m(x)\right) \tag{2}$$

Pareto rationality is the theoretical framework to analyse multi objective optimization problems where m objective functions $f_1(x), \dots, f_m(x)$, where $f_i(x) : \rightarrow \mathbb{R}$ are to be simultaneously optimized in the search space $\Omega \subseteq \mathbb{R}^d$. Here we use x to be compliant with the typical Pareto analysis's notation, clearly in this study x is a sensor placement s.

Let $u, v \in \mathbb{R}^m$ u is said to dominate v if and only if $u_i \geq v_i \forall i = 1, \dots, n$ and $u_j > v_j$ for at least one index j to refer to the vector of all objectives evaluated at a location x. The goal in multi-objective optimization is to identify the Pareto frontier of $f(x)$. A point x^* is pareto optimal for the problem in Eq. (1) if there is no point x such that $F(x)$ dominate $F(x^*)$. This implies that any improvement in a Pareto optimal point in one objective leads to a deterioration in another. The set of all Pareto optimal points is the Pareto set and the set of all Pareto optimal objective vectors is the Pareto front (PF). The interest in finding locations x having the associated $F(x)$ on the Pareto frontier is clear: all of them represent efficient trade-offs between conflicting objectives and are the only ones, according to the Pareto rationality, to be considered by the decision maker. The issue of the quality evaluation of Pareto solutions sets is the key issue in multi objective optimization. The quality evaluation of Pareto solutions sets is the key issue in multi objective optimization.

To measure the progress of the optimization a natural and widely used metric is the hypervolume indicator that measures the objective space between a non-dominated set and a predefined reference vector. An example of Pareto frontier, along with the reference point to compute the hypervolume, is reported in Fig. 1.

A good approximation of the Pareto set will result into a high hypervolume value; thus, hypervolume is a reasonable measure for evaluating the quality of the optimization process.

The grey shaded area is the original hypervolume: a new point A improves the approximation to the exact Pareto front and increases the hypervolume by the blue shaded area. The improvement of the hypervolume can also be used in the selection of the new point as in Daulton et al. (2020). This

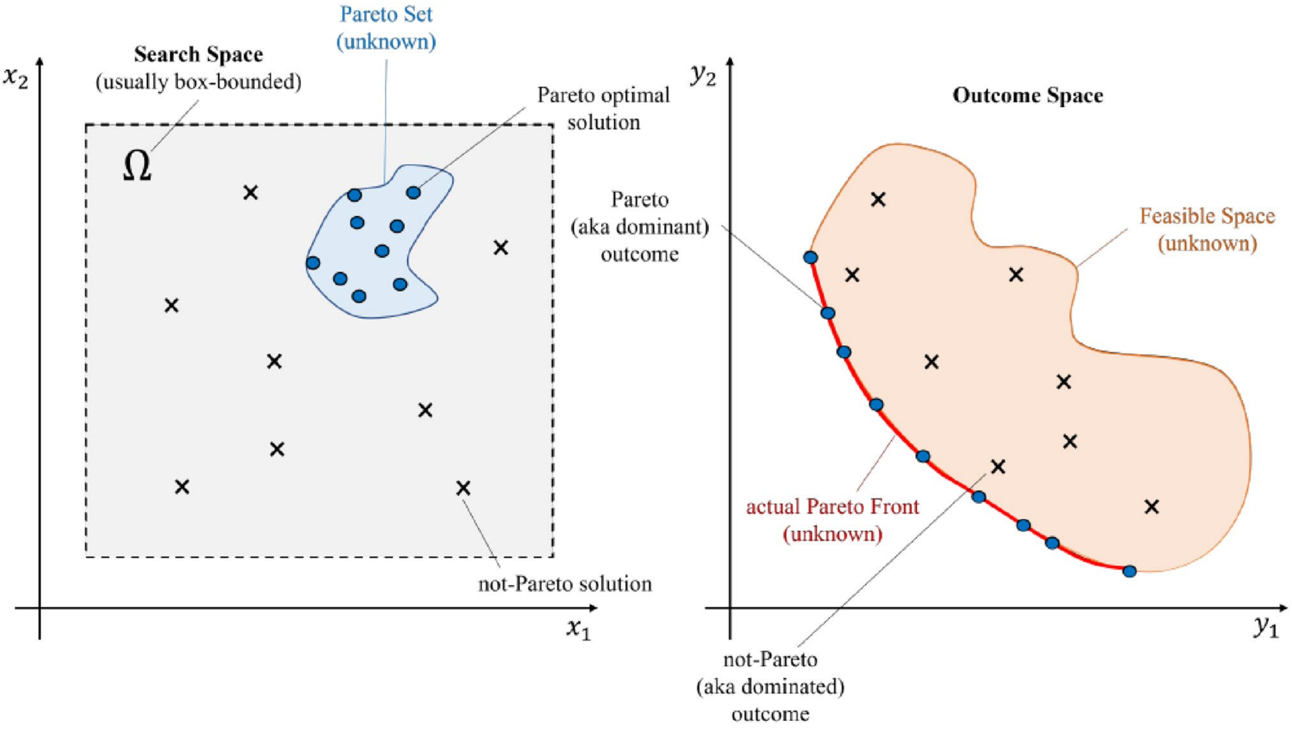


Fig. 1 An example of Pareto frontier (left), with the associated hypervolume (right), for two minimization objectives

approach has been further developed using the gradient of the expected hypervolume improvement to speed up the selection process and also in the selection of the new point.

Another metric to compare different approximations of the Pareto front is the C-metric, also called coverage. Let A and B be two approximations of the PF, $C(A, B)$ gives the fraction of solutions in B that are dominated by at least one solution in A. Hence, $C(A, B) = 1$ means that all solutions in B are dominated by at least one solution in A while $C(A, B) = 0$ implies that no solution in B is dominated by a solution in A.

2.3 Structure of the overall approach

Phase A: data representation and analysis

Step 1: Distributional representation of each user as a histogram whose values depend on the distribution of (user, user) similarity values. This step can be considered as the embedding of each user in a lower dimensional space whose elements are the histograms.

Step 2: The distance between histograms is computed as the Wasserstein distance (a.k.a. optimal transport distance) and the space of histograms, endowed with the Wasserstein metric, is called the Wasserstein space.

Step 3: The rating matrix can be represented as two alternative graphs:

i. The usual cosine graph where each edge is weighted by the cosine similarity of the two end-point users of the edge.
ii. The WST graph where each edge is weighted by the WST distance of the two histograms associated to the two end-point users of the edge.

Each graph can be clustered by any standard method like k-means or spectral clustering. In the computational results, spectral clustering has been used.

Phase B: optimization

Step 4: Formulation of the multi-objective optimization problem.

The candidate top-k list is encoded as a matrix. The values of the objective functions (accuracy, novelty and coverage) are represented as multivariate histograms.

Step 5: Multi-objective evolutionary algorithm.

The selection operation is built upon the WST distance between histograms associated to two candidate solutions.

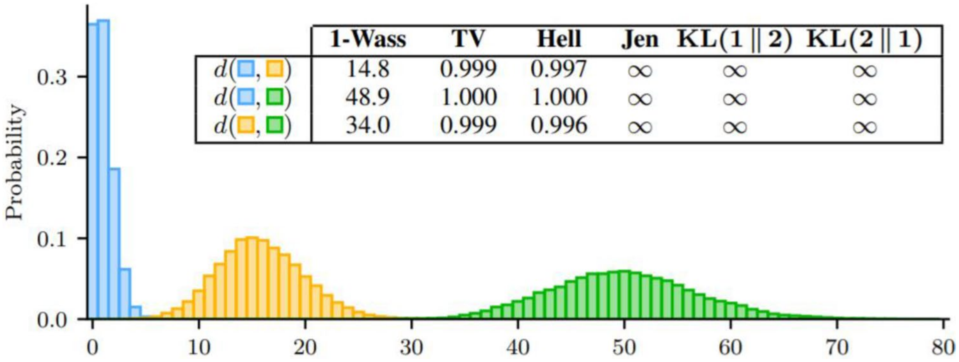
To measure the performance of the algorithm over each cluster computed in Step 3, hypervolume and coverage of the approximate pareto sets have been used.

3 The Wasserstein distance

Measuring the distance between distributions can be accomplished by many alternative models. Most used distances are Kullback–Leibler distance, Hellinger distance, total variation divergence and χ-square divergence and Jensen-Shannon, which is a symmetrized version of KL. The main disadvantage of KL and χ-square to measure the distance between histograms is that they account only for the correspondence between bins of the same index and do not use information across bins or distributions with different binning schemes, that is different support.

This is clearly shown in Fig. 2 (Öcal et al. 2019), which compares the values of 5 distances Total Variation (TV), Hellinger, Jensen-Shannon and Kullback–Leibler (KL) between 3 histograms. It's apparent that WST not only shows a better discrimination but also an intuitive and natural perception of the distances.

WST is a cross binning distance and is not affected by different binning schemes. Moreover, WST matches naturally the perceptual notion of nearness and similarity. Moreover, Wasserstein embedding can be easily generalized to multi-objective problems considering $\boldsymbol{m}$-dimensional histograms:



	1-Wass	TV	Hell	Jen	KL(1 ‖ 2)	KL(2 ‖ 1)
d(🟦, 🟨)	14.8	0.999	0.997	∞	∞	∞
d(🟦, 🟩)	48.9	1.000	1.000	∞	∞	∞
d(🟨, 🟩)	34.0	0.999	0.996	∞	∞	∞

Fig. 2 Three histograms and their distances (Öcal et al. 2019)

3.1 Basic definitions

The WST distance between continuous probability distributions is:

$$W_p\left(P^{(1)},P^{(2)}\right) = \left(\inf_{\gamma\in\Gamma\left(P^{(1)},P^{(2)}\right)} \int_{X\times X} d\left(x^{(1)},x^{(2)}\right)^p d\gamma\left(x^{(1)},x^{(2)}\right)\right)^{\frac{1}{p}} \quad (3)$$

where $d(x^{(1)},x^{(2)})$ is also called ground distance (usually it is the Euclidean norm), $\Gamma\left(P^{(1)},P^{(2)}\right)$ denotes the set of all joint distributions $\gamma(x^{(1)},x^{(2)})$ whose marginals are respectively $P^{(1)}$ and $P^{(2)}$, and $p>1$ is an index. The Wasserstein distance is also called the Earth Mover Distance (EMD). The EMD is the minimum energy cost of moving and transforming a pile of sand in the shape of $P^{(1)}$ to the shape of $P^{(2)}$. The cost is quantified by the amount of sand moved times the moving distance $d(x^{(1)},x^{(2)})$.The EMD then is the cost of the optimal transport plan.

There are some specific cases, very relevant in applications, where WST can be written in an explicit form. Let $\widehat{P}^{(1)}$ and $\widehat{P}^{(2)}$ be the cumulative distribution for one-dimensional distributions $P^{(1)}$ and $P^{(2)}$ on the real line and $\left(\widehat{P}^{(1)}\right)^{-1}$ and $\left(\widehat{P}^{(2)}\right)^{-1}$ be their quantile functions.

$$W_p\left(P^{(1)},P^{(2)}\right) = \left(\int_0^1 \left|\left(\hat{P}^{(1)}\right)^{-1}\left(x^{(1)}\right) - \left(\hat{P}^{(2)}\right)^{-1}\left(x^{(2)}\right)\right|^p dx\right)^{\frac{1}{p}} \quad (4)$$

Let's now consider the case of a discrete distribution P specified by a set of support points x_i with $i=1,\dots,m$ and their associated probabilities w_i such that $\sum_{i=1}^m w_i = 1$ with $w_i \geq 0$ and $x_i \in M$ for $i=1,\dots,m$.

Usually, $M=\mathbb{R}^d$ is the d-dimensional Euclidean space with the l_p norm and x_i are called the support vectors. M can also be a symbolic set provided with a symbol-to-symbol similarity. P can also be written using the notation:

$$P(x) = \sum_{i=1}^m w_i \delta\left(x - x_i\right) \quad (5)$$

where $\delta(\cdot)$ is the Kronecker delta.

The WST distance between two distributions $P^{(1)} = \left\{w_i^{(1)}, x_i^{(1)}\right\}$ with $i=1,\dots,m_1$ and $P^{(2)} = \left\{w_i^{(2)}, x_i^{(2)}\right\}$ with $i=1,\dots,m_2$ is obtained by solving the following linear program:

$$W\left(P^{(1)},P^{(2)}\right) = \min_{\gamma_{ij}\in\mathbb{R}^+} \sum_{i\in I_1, j\in I_2} \gamma_{ij} d\left(x_i^{(1)}, x_j^{(2)}\right) \quad (6)$$

The cost of transport between $x_i^{(1)}$ and $x_j^{(2)}$, $d\left(x_i^{(1)},x_j^{(2)}\right)$, is defined by the p-th power of the norm $\|x_i^{(1)}, x_j^{(2)}\|$ (usually the Euclidean distance).

We define two index sets $I_1 = \left\{1,\dots,m_1\right\}$ and I_2 likewise, such that:

$$\sum_{i\in I_1} \gamma_{ij} = w_j^{(2)}, \forall j \in I_2 \quad (7)$$

$$\sum_{j\in I_2} \gamma_{ij} = w_i^{(1)}, \forall i \in I_1 \quad (8)$$

Equations (7) and (8) represent the in-flow and out-flow constraint, respectively. The terms γ_{ij} are called matching weights between support points $x_i^{(1)}$ and $x_j^{(2)}$ or the optimal coupling for $P^{(1)}$ and $P^{(2)}$.

The discrete version of the WST distance is usually called Earth Mover's Distance (EMD). For instance, when measuring the distance between grey scale images, the histogram weights are given by the pixel values and the coordinates by the pixel positions. Another way to look at the computation of the EMD is as a network flow problem. In the specific case of histograms, the entries γ_{ij} denote how much of the bin i has to be moved to bin j.

The basic computation of OT between 2 discrete distributions involves solving a network flow problem whose computation scales typically cubic in the sizes of the measure. There are 2 lines of work to reduce the time complexity of OT, simple ground costs can lead to simpler computations. In the general case it is shows to be equivalent to a min-flow algorithm of quadratic computational complexity and, in specific cases, to linear. The computation of EMD turns out to be the solution of a minimum cost flow problem on a bi-partite graph where the bins of $P^{(1)}$ are the source nodes and the bins of $P^{(2)}$ are the sinks while the edges between sources and sinks are the transportation costs. In the case of one-dimensional histograms, the computation of WST reduces to the comparison of two 1-dimensional histograms which can be performed by a simple sorting and the application of the following Eq. (11).

$$W_p\left(P^{(1)},P^{(2)}\right) = \left(\frac{1}{n}\sum_{i}^{n} \left|x_i^{(1)*} - x_i^{(2)*}\right|^p\right)^{\frac{1}{p}} \quad (9)$$

where $x_i^{(1)*}$ and $x_i^{(2)*}$ are the sorted samples.To clarify the relation between Euclidean and WST distance we consider histograms seen as probability vectors of length d, belonging to the probability simplex:

$$\Sigma_d = \left\{ u \in \mathbb{R}_+^d \,|\, \sum_{i=1}^d u_i = 1 \right\} \quad (10)$$

Assume that we wish to compare images of 10 × 10 = 100 pixels and that these pixels can only take values in a range of 4 possible colours, dark red (dR), light red (lR), dark blue (dB) and light blue (lB): each image can therefore be associated to a histogram of 4 colours.

The Euclidean distance (the *l2* norm of the difference of two vectors) computes the distance between ***a*** and ***b*** by comparing for each given index *i* their coordinates a_i and b_i one at a time. The Euclidean distance between ***a*** and ***b*** is 66, between ***a*** and ***c*** is 69, and between ***b*** and ***c*** is 77. For the Manhattan distance (the *l1* norm of the difference of two vectors) of three histograms, we obtain that it is 120 between ***a*** and ***b*** as well as ***a*** and ***c*** and it is 114 between ***b*** and ***c***.

Already in Aitchison (1982) and in Le and Cuturi (2015) it was remarked that the information reflected in histograms lies more in the relative value of their coordinates rather than on their absolute value.

This observation matches with the reasonable assumption that dark and light red have more in common than dark red and dark blue supporting the intuition that ***c*** should be closer to ***a*** than it is to ***b***.

The Wasserstein distance implements this intuition by carrying out an optimization procedure to compute a distance between histograms.

The transport distance is defined as the lowest cost one could possibly find by considering all possible transport plans from ***a*** to ***b***. For ***a***, ***b***, and ***c***, we obtain Table 1.

We can see that the transport distance agrees with our initial intuition that ***a*** is closer to ***c*** than ***b*** by considering a metric computed on features. The Manhattan distance does not discriminate because it assigns the same value to the pairs (***a***, ***b***) and (***a***, ***c***). The Euclidean distance does not discriminate because it results respectively 66.83 between ***a*** and ***b***, and 77.24 between ***b*** and ***c***, and 72 between ***a*** and ***c***.

Table 1 Distances between the histograms in Fig. 3

	$\updownarrow_1$	$\updownarrow_2$	$\mathcal{W}_1$	$\mathcal{W}_2$
(a, b)	120.00	66.83	125.00	134
(b, c)	114.00	77.24	59.00	75
(a, c)	120.00	69.22	72.00	68

WST distance is also a full metric in that satisfies also the triangle inequality.

3.2 Computational issues

There are some particular cases, very relevant in applications, where WST can be written in an explicit form. Let F and G be the cumulative distribution for one-dimensional distributions f and g on the real line and F^{-1} and G^{-1} be their quantile functions.

$$W_p(f,g) = \left(\int_0^1 \left| F^{-1}(x) - G^{-1}(x) \right|^p dx \right)^{\frac{1}{p}} \tag{11}$$

Then the computation of WST reduces to the comparison of two 1-dimensional histograms which can be performed by a simple sorting and the application of Eq. (11).

$$W_p(f,g) = \left(\frac{1}{n} \sum_i^n \left| x_i^* - y_i^* \right|^p \right)^{\frac{1}{p}} \tag{12}$$

where x_i^* and y_i^* are the sorted samples. In this paper $p = 1$. The output of the optimization problem is the distance WST and the optimal transport map, where M_{ij} is the metric cost matrix defining the cost of moving mass from x_i to y_j and $\gamma_{ij} \in \mathbb{R}^+$.

To the term $W_p(f,g)$ we add the entropic regularizer given by the entropy of the matrix $\Gamma = [\gamma_{ij}]$.

$$H(\Gamma) = \sum_{ij} \gamma_{ij} \log \gamma_{ij} \tag{13}$$

$$\begin{aligned} W(f,g) = & \min_{\gamma_{ij} \in \mathbb{R}^{n \times n'}} \sum_{i=1}^{n} \sum_{j=1}^{n'} \left(\gamma_{ij} M_{ij} + \lambda \gamma_{ij} \log \gamma_{ij} \right) \\ & s.t. \sum_{j=1}^{n'} \gamma_{ij} = f_i, \sum_{i=1}^{n} \gamma_{ij} = g_j, \gamma_{ij} \geq 0 \end{aligned} \tag{14}$$

The additional computational cost of MOEA/WST over MOEA is due to the computation of the WST distance.

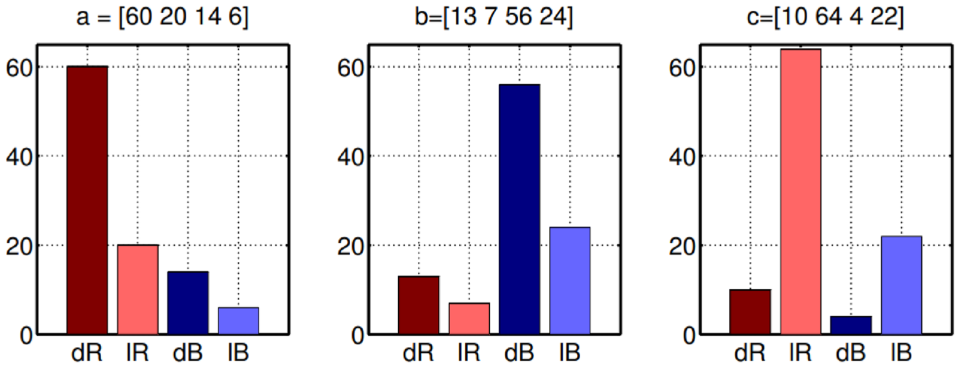


Fig. 3 The above picture is taken from Cuturi and Avis (2014)

The formula (11) that we have used to construct $G_w = (V_w, E_w)$ does no longer work for $d > 1$. To speed up the search we could consider approximate solutions as suggested in Backurs et al. (2020), Atasu and Mittelholzer (2019). In this paper we have solved the full optimal transport problem given by Eq. (5) using the library Python Optimal Transport (POT) (Flamary et al. 2021) and specifically extending to 3d the OT networks simplex solver.

Here we consider the computational cost of OT. Assuming that each objective function is quantized in 10 bins, we have a 3D "image" of 1000 bins. The transport matrix $\gamma_{ij} \in \Gamma$ has n=1 M entries, the linear program has a worst-case complexity, using interior point methods, of *3n log n*. This can be reduced in several specific cases, notably for univariate distributions using Eq. (7). This is clearly displayed by the behaviour of the wall clock time. Another solution, offered in the OPT package, is the Sinkhorn regularization.

If we define $K_{ij} = e^{-\frac{d_{ij}}{\alpha}}$ where d_{ij} represents the cost of moving from x_i to y_j and λ is a multiplicative coefficient in the entropic regularizer $H(\Gamma)$. Then it can be shown that the solution γ^*_{ij} can be expressed as $u_i K_{ij} v_j$ where u and v are unknown vectors. This result means that instead of optimizing over the $n \times n'$ values of the full matrix $[\gamma_{ij}]$ we have to optimize over $n + n'$ values u and v (i.e., Eqs. (15) and (16) would result in 2000 variables instead of 1 M).

$$\sum_j T_{ij} = s_i \Rightarrow u_i \sum_j K_{ij} v_j = s_i \Rightarrow u_i = \frac{s_i}{\sum_j K_{ij} v_j} \tag{15}$$

$$\sum_i T_{ij} = d_j \Rightarrow v_j \sum_i K_{ij} u_i = d_j \Rightarrow v_j = \frac{d_j}{\sum_i K_{ij} u_i} \tag{16}$$

Alternating the estimation of u and v the algorithm will reach the correct values. This result is the basis of the most effective approximate algorithms which are analyzed in Peyré and Cuturi (2019).

Starting from the consideration that the variables in w are more important that the matching weights, approximate solvers have been proposed, specifically Sinkhorn solvers. Here it is just important to remark that they allow to manage the trade-off between accuracy and computational cost through a regularization hyperparameter. Entropic regularization enables scalable computations, but large values of the regularization parameter lambda in Eq. (13) could induce an undesirable smoothing effect while low values not only reduce the scalability but might induce several numeric instabilities.

4 Graph representation of the rating matrix

The graph representation of rating matrix is very useful both for natural interpretation and as enabler of many graph analysis tools.

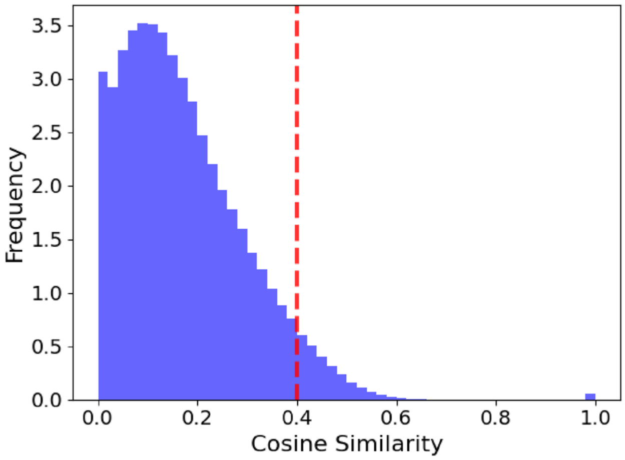


Fig. 4 Distribution of the Cosine similarity between all pair of nodes

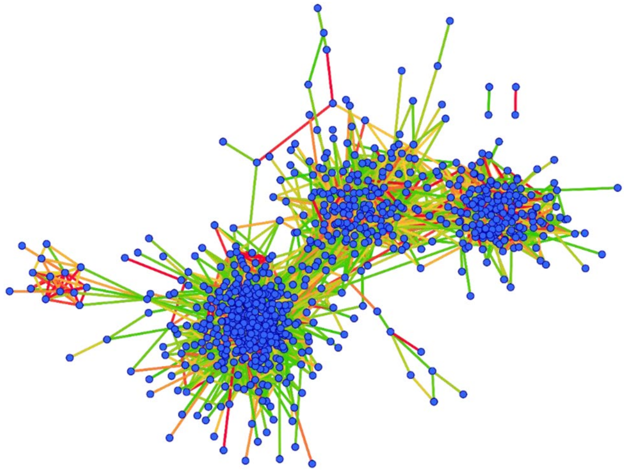


Fig. 5 Cosine graph

4.1 Cosine graph

The first graph representation of the rating matrix is directly from k-NN method. Two users/items (vertices) i and j are connected by an edge if their distance (given by a similarity measure like cosine or Pearson) is among the $k$smallest distances from i to other users/items.

A more general representation is given by the cosine graph. Figure 4 shows the distribution of cosine similarity between all pairs of users.

The cosine similarity between individual users, can be used to build a graph $G_c = (V_c, E_c)$, in which two nodes (users) are linked together if their similarity is above a given threshold (the dotted red line in Fig. 4); then $V_c = \{u_i\}_{i=1,\ldots,M}$ is the set of users and $E_c = \{(i,j) : \cos(u_i, u_j) > \tau\}$ is the set of edges. Each edge of this graph is then weighted based on the similarity $\cos\left(u_i, u_j\right)$. Figure 5 displays the resulting graph, in which the edge color represents its weight. The nodes connected by a red edge are the most similar, while the ones connected by a green edge are the most different.

4.2 Users as histograms, the feature graph and its clustering

Another graph representation is through the association to each user u_i of a one-dimensional histogram $h(u_i)$: the bins are the equi-subdivisions of the interval $[0, 1]$ for cosine similarity (and $[-1, 1]$ for Pearson correlation) and the weights are the fraction of users whose cosine similarity falls in each bin; the same representation can be item driven. In Fig. 6 three users' histograms are shown.

According to this representation each user is described by a signature, feature vector, given by the bins and the associated weights. The elements in this feature space are probabilistic distributions. Whose distance is measured by WST distance. Figure 6 shows the distribution of the Wasserstein distance averaged over all the pairs of nodes.

This distributional representation of the users enables construction of another graph (Fig. 7) in which the nodes are the users that are linked if their distributions of similarity are close enough, i.e., their Wasserstein distance is below a given threshold (the red dotted line in Fig. 7). Let $G_w = (V_w, E_w)$ be the Wasserstein graph, then $V = \left\{h(u_i)\right\}_{i=1,\dots,M}$ is the set of users represented as histograms and $E_w = \left\{(i,j) : WST\left(h\left(u_i\right), h\left(u_j\right)\right) < \tau\right\}$ is the set of edges. The edge (i,j) is then weighted based on the Wasserstein distance $WST\left(h\left(u_i\right), h\left(u_j\right)\right)$ between the similarity distributions of nodes i and j (see Fig. 8).

The same procedure could be used to construct the item-base Wasserstein graph.

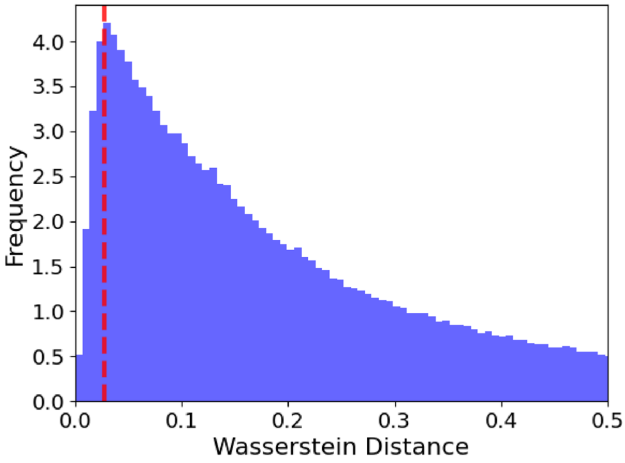


Fig. 7 Distribution of the Wasserstein distance between all pair of nodes. To enable a better visualization, we omitted in the figure pair of nodes whose WST is greater than 0.5

5 Objective functions

For the problem of finding the optimal top-L recommendation list, three conflicting objectives are considered, i.e., accuracy, coverage, and novelty. The distributional representation of these three metrics enables the definition of an information space that allows the use of MOEA/WST In particular, each recommendation list matrix can be represented by a three dimensional histogram as shown in Fig. 9.

5.1 Rating score

To each recommendation list, it is possible to assign a score that represents how "good" the items recommended to users are. This score is based on the sum of the ratings given by the users to the recommended items and is given by the following equation:

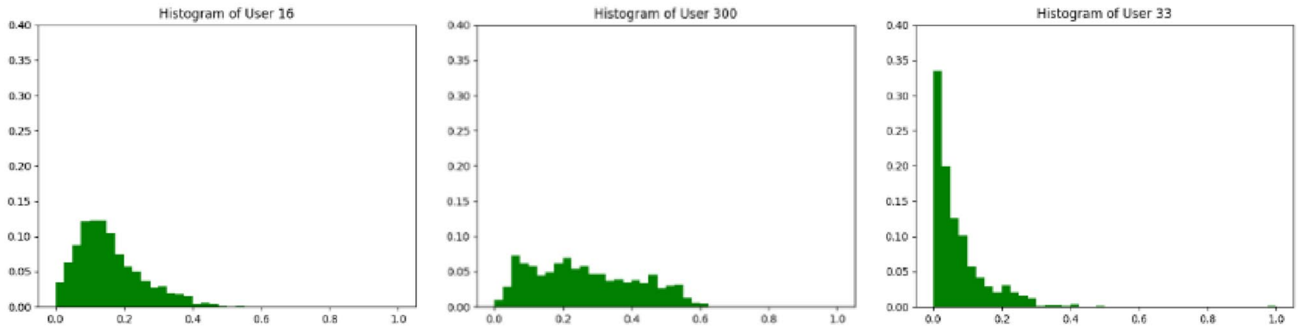


Fig. 6 Cosine similarity histograms with a bin length of 0.025

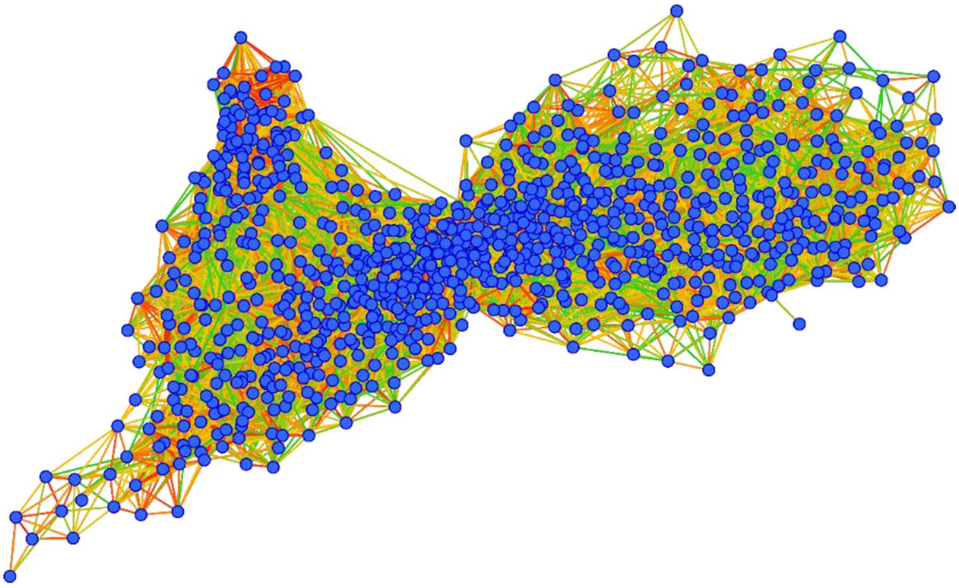
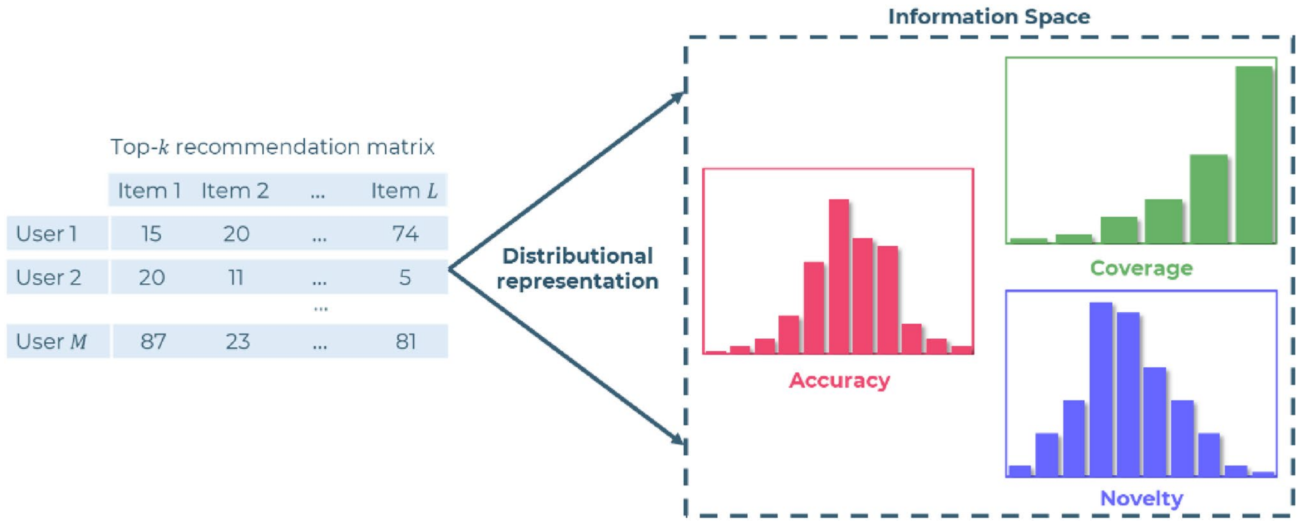


Fig. 8 Distributional Wasserstein graph



Top-k recommendation matrix

	Item 1	Item 2	...	Item L
User 1	15	20	...	74
User 2	20	11	...	5
			...	
User M	87	23	...	81

Fig. 9 Representation of the recommendation list

$$score = \frac{1}{M \cdot L} \sum_{u_i \in U} \sum_{o_j \in S_L(u_i)} r(u_i, o_j) \tag{17}$$

where $r(u_i, o_j)$ is the rating given by user u_i to item o_j. Maximize this score ensures that the recommendation list of each user contains only items that user has given a high rating.

This particular definition of *score* admits a distributional representation. The distribution is given by the values of accuracy of each user as in Eq. (16). This distribution can be represented by a histogram (Fig. 10) in which the support points $k_1, \dots, k_{N_a}$ correspond to accuracy values, and the weights w_{k_i} with $i = 1, \dots, N_a$ represent the fraction of users with a certain value of *score* $w_{k_i} = \frac{1}{M}\left|\left\{ u_i : \sum_{o_j \in S_L(u_i)} r(u_i, o_j) \in \left[k_i, k_{i+1}\right) \right\}\right|$.

One problem with Collaborative Filtering recommendation is the "popularity bias": (Abdollahpouri et al. 2019) popular items are being recommended too frequently while most of the items do not get attention. This is one reason why in recent research papers, the *score* is considered together with other two objectives, coverage and novelty.

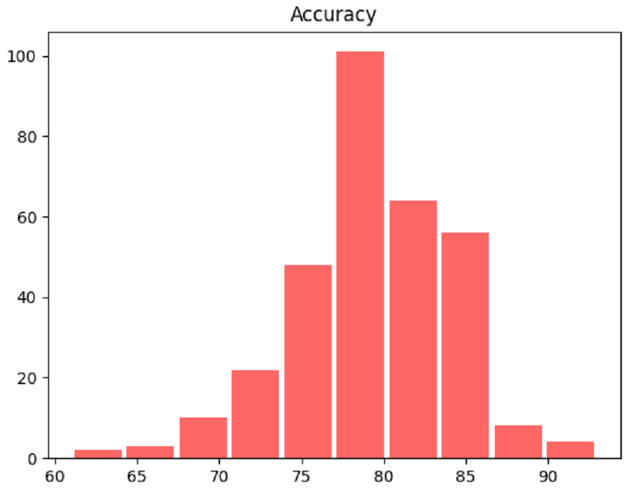


Fig. 10 Distributional representation of accuracy

5.2 Coverage

A recommender system is expected to provide M recommendation lists. Each list corresponds to a user and consists of L items. The coverage of the recommendation list is defined as the number of different items in all users' top-L lists.

$$coverage = \frac{1}{N}\left|\bigcup_{u_i\in U} S_L\left(u_i\right)\right| \tag{18}$$

The objective function *coverage* is averaged over the number of items N. Coverage reflects the diversity of recommendation. A larger value of coverage is better because more choices are provided to the users.

Also the coverage admits a distributional representation. The distribution is given by the ratio between the non-duplicated items in the recommendation list and the total number of items for each user, i.e., the coverage of the user recommendation list $S_L(u)$.

This distribution can be represented by a histogram (Fig. 11) in which the support points are the values of coverage $k_1, \dots, k_{N_c}$, and the weights w_{k_i} with $i = 1, \dots, N_c$ represent the fraction of users with a certain value of coverage $w_{k_i} = \frac{1}{M}\left|\left\{u_i : \left|S_L\left(u_i\right)\right| \in \left[k_i, k_{i+1}\right)\right\}\right|$.

5.3 Novelty

This objective is based on the degree d_j of an item o_j that is the number of times it has been rated. Then, the self-information (Zhou et al. 2010) of the item o_j is given by:

$$N_j = \log_2 \frac{M}{d_j} \tag{19}$$

The novelty is then defined as the average self-information of all the items in the recommendation lists of each user:

$$Novelty = \frac{1}{M}\sum_{u_i\in U}\sum_{j\in S_L(u_i)}\frac{N_j}{L} \tag{20}$$

It's important to note that novelty also admits a distributional representation. The distribution is given by the values $\sum_{i\in S_u}\frac{N_i}{L}$ for each user u.

This distribution can be represented by a histogram (Fig. 12) in which the support points $k_1, \dots, k_{N_n}$ are the values of novelty, and the weights w_{k_i} with $i = 1, \dots, N_n$ represent the number of users with a certain value of novelty $w_k = \frac{1}{M}\left|\left\{u_i : \sum_{j\in S_L(u_i)}\frac{N_j}{L} \in [k, k+1)\right\}\right|$.

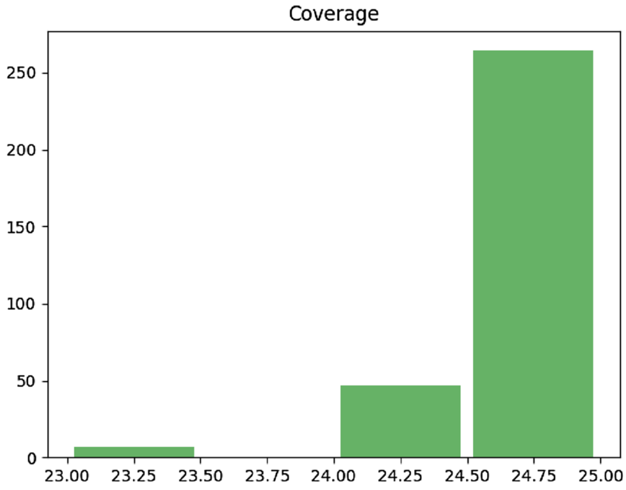


Fig. 11 Histogram representation of coverage

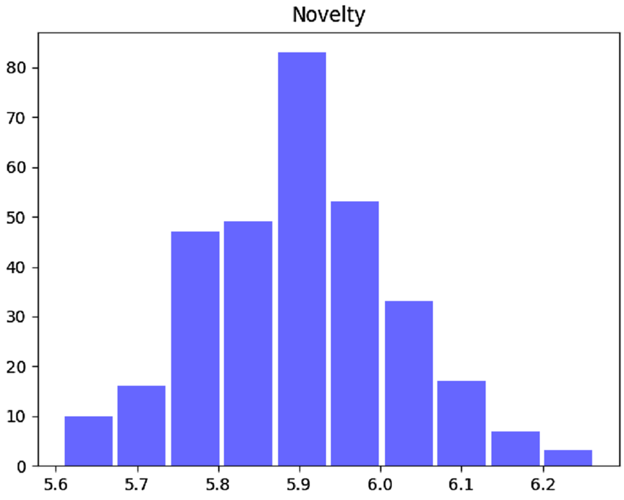
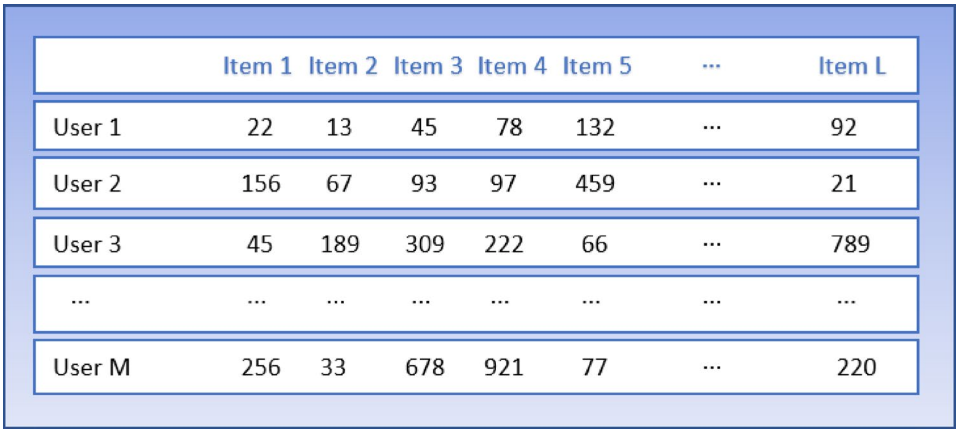


Fig. 12 Distributional representation of novelty



	Item 1	Item 2	Item 3	Item 4	Item 5	···	Item L
User 1	22	13	45	78	132	···	92
User 2	156	67	93	97	459	···	21
User 3	45	189	309	222	66	···	789
···	···	···	···	···	···	···	···
User M	256	33	678	921	77	···	220

Fig. 13 Matrix encoding of a top-L recommendation list

The Eqs. (19) and (20) could be interpreted as relating the novelty to the number of items originally unrated that are recommended to users, in the recommendation lists.

5.4 Encoding of solutions and their representation

A solution is represented as a matrix whose rows are the users in the cluster and the columns represent for each user the top-L items (see Fig. 13).

The multi-objective problem is

$$\max_{x\in\Omega\subseteq\mathbb{R}^d} F(x) = \left(f_1(x), f_2(x), f_3(x)\right)$$

where x is given by the entries of the matrix shown before. The codomain of each objective function $f_i(x)$ is subdivided in n_i bins.

The distributional representation of the three previously defined objective can be viewed as a three-dimensional histogram. For each recommendation list S_L the support points of this histogram are the values of accuracy along the x-axis, the values of coverage along the y-axis and the values of novelty along the z-axis; the weights represent the fraction of users whose values of accuracy, coverage and novelty fall in a specific range. These distributions constitute the space on which the MOEA/WST algorithm is based. Therefore, it uses the Wasserstein distance computed by formula (6), (7), (8) to compare the histograms associated to different top-L recommendation lists. In our problem f_1 is the accuracy given by (16), f_2 is the coverage given by (17) and f_3 is the novelty, given by (19).

6 The description of MOEA/WST

The multi-objective evolutionary algorithm used for the solution of problem (1) is based on the Python framework Pymoo (Blank and Deb 2020). This section is focused on analyzing how the mathematics presented in the previous sections enables the construction of a new genetic operator and how it can be embedded in the general workflow of Pymoo.

The set of non-dominated solutions is called the Pareto set and its image in the objective space the Pareto Front (PF). For a Pareto optimal solution, the value of one objective cannot be improved unless at least one of the other objectives is negatively affected.

In order to approximate the Pareto set MOEAs we follow a strategy based on non-dominated sorting genetic algorithms which is implemented in Pymoo by the algorithm NSGA-II.

The general structure is as follows. The blue modules are basically the same as in Pymoo. The red one "selection" is a contribution of this paper and is analyzed in detail in 6.1.

The algorithm begins with a random selection of a number of individuals among which we select the non-dominated individuals as elements of first approximation of the Pareto set. Then MOEA/WST performs the following steps: (i) selection, (ii) crossover and (iii) mutation.

Pymoo offers different solutions for the above points. For the computations Simulated Binary Crossover with η = 3 and Inverse Mutation have been used. The element of novelty of the proposed algorithm with respect to NSGA-II is the selection operator based on the Wasserstein distance analyzed in Sect. 6.1. The new algorithm is accordingly called MOEA/WST. To evaluate the performance of the MOEAs the hypervolume metric is generally adopted given by the volume of the portion of the objective space enclosed by a reference point and the approximate Pareto front.

6.1 Selection

In order to select the pairs of parents to be mated using the crossover operation, we have introduced a problem specific selection method that takes place into the Wasserstein space. In this paper we used the Wasserstein distance between the histograms corresponding to the recommendation lists F_i and M_i.

First, we randomly sample from the actual Pareto set two pairs of individuals (F_1, M_1) and (F_2, M_2). Then we choose the pair $\left(F_i, M_i\right)$ as the parents of the new offspring, where $i = \arg\max_{i\in\{1,2\}} WST(F_i, M_i)$. This favours exploration and diversification in the *Wasserstein space* (see Fig. 14).

7 Computational results

In this section, the computational results over the MovieLens dataset are reported. First, the two algorithms, NSGA-II and MOEA/WST, have been tested on the clusters resulting from the graph in Fig. 4. The optimization has been run for each cluster. Figure 15 shows the Hypervolume over generations of NSGA-II (red) and MOEA/WST (blue). Since multiple runs of the algorithms are performed, the charts display mean and standard deviation of the target metric.

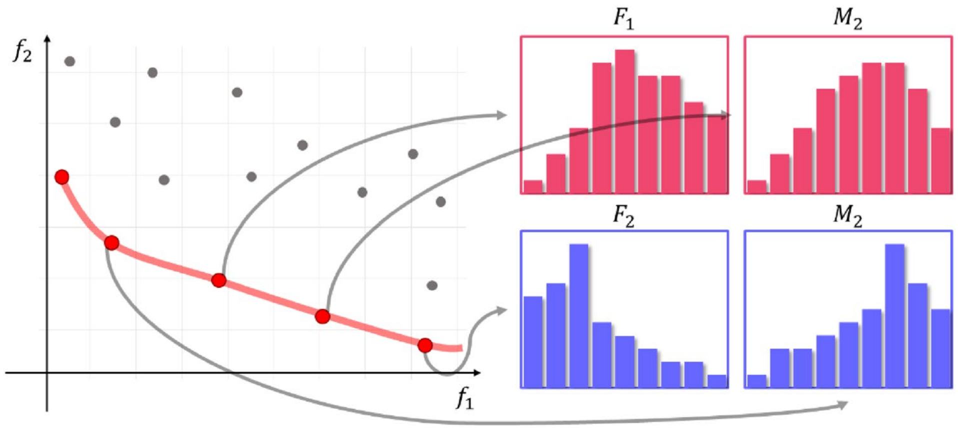


Fig. 14 The Wasserstein-based selection operator

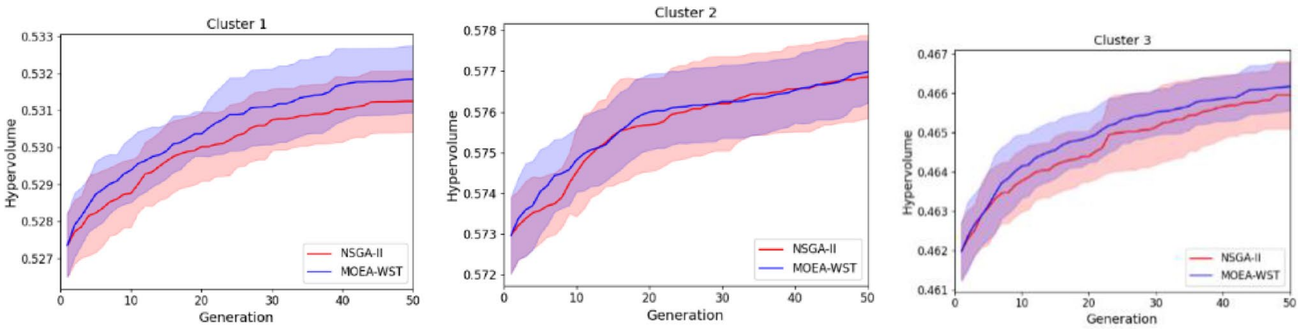


Fig. 15 Hypervolume over generations of the three different clusters identified by clustering the cosine graph

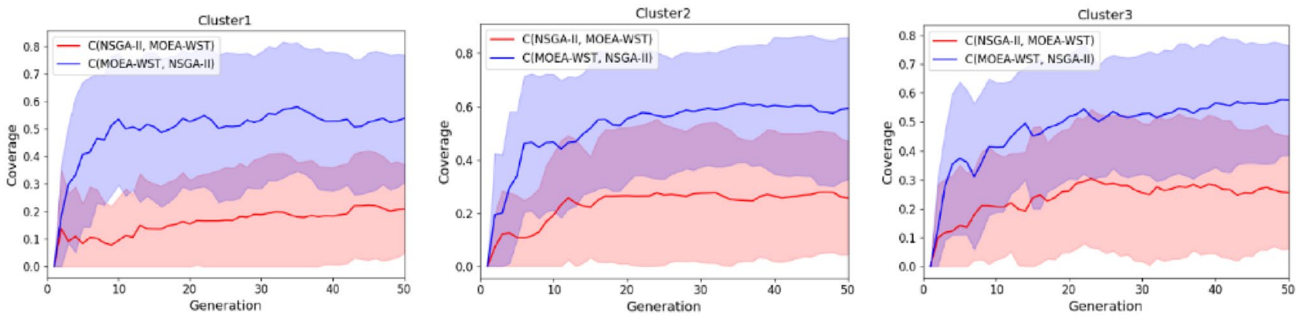


Fig. 16 Coverage over generations of the three different clusters identified by clustering the cosine graph

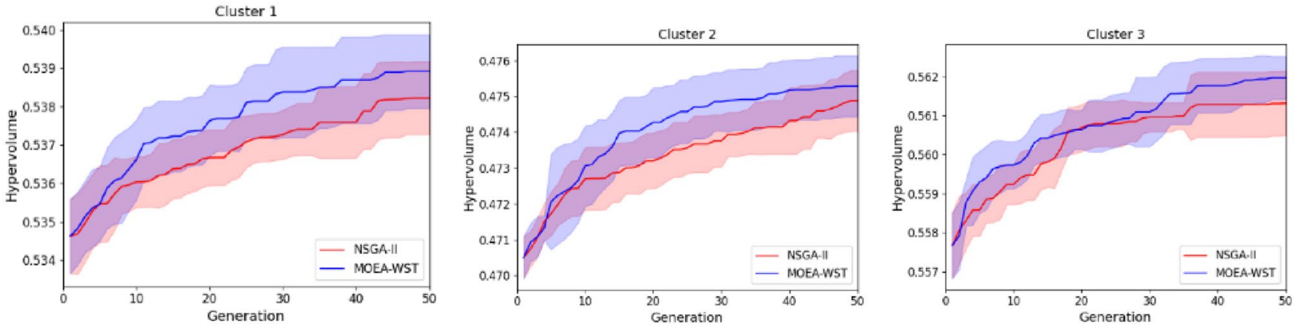


Fig. 17 Hypervolume over generations of the three different clusters identified by clustering the Wasserstein graph

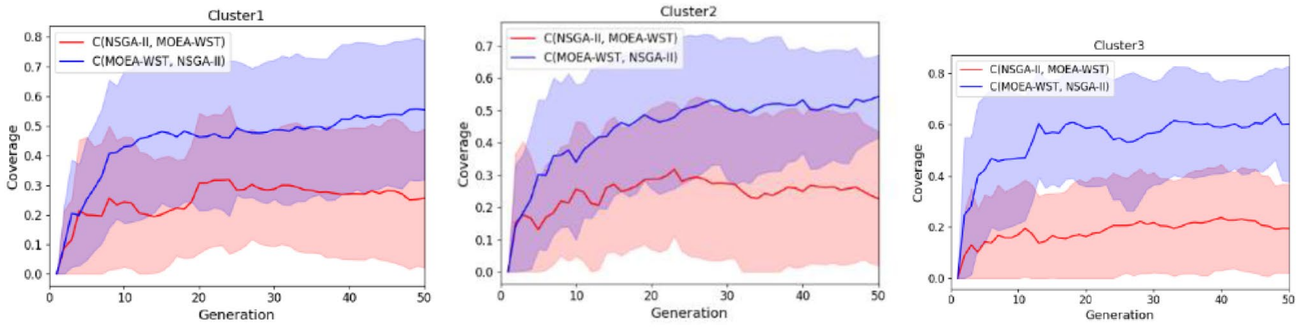


Fig. 18 Coverage over generations of the three different clusters identified by clustering the Wasserstein graph

The hypervolume curve of MOEA/WST is better than NSGA. Also, in terms of the *C*-metrics, defined in Sect. 2, and shown in Fig. 16, MOEA/WST is the better performer. Then the two algorithms have been used, with the same settings, on the Wasserstein graph in Fig. 7. The results, respectively for hypervolume and *C*-metrics are reported respectively in Figs. 17 and 18.

Again, both the hypervolume curve and the *C*-metric of MOEA/WST is better than NSGA-II. It is also important to note that, using the clusters over the Wasserstein graph, both algorithms perform better than using the cosine graph.

8 Limitations and discussion

The main limitation in the application of the Wasserstein distance is its computational cost for multivariate histograms. The straightforward utilization of Eqs. (6), (7) and (8) can become prohibitive. A first solution is to use approximate solvers as those quoted in Sect. 3.2. Another solution, offered in Python Optimal Transport (POT) is the Sinkhorn regularization which has the effect of reducing drastically the computational cost introducing a regularization term. Handling the regularization parameter is not easy and requires a delicate balance between accuracy and numerical stability due to ill conditioning. Another limitation is that the cost matrix is usually assumed in a problem agnostic way: a better solution is to use algorithms that can learn the ground metric using only a training set of labelled histograms as proposed in Cuturi and Avis (2014) and Heitz et al. (2021).

Given the growing importance of WST distance in fields like imaging, the generation of adversarial network among others approximate algorithms are being proposed. Beugnot et al. (2021) introduce before the linear solver a pre-processing step in which they summarize a smaller number of representative samples therefore solving a smaller linear problem. A different approach has been recently suggested in Si et al. (2020) where they provide insights as to why, despite the curse of dimensionality, the WST distance enjoys favourable empirical performance.

The computational overhead related to the computation of the distance is expectedly significant because the support of the histograms is 3-dimensional. As it can be gleaned from Fig. 17, the value 0.538 of hypervolume is reached by MOEA/WST in 25 generations versus the 50 generations required by NSGA-II. Each generation requires 10 function evaluations which brings the advantage of MOEA/WST over NSGA-II to 250 function evaluations. The wall-clock time from these specific computations is

80 s for NSGA-II and 220 s for MOEA/WST of which 140 s are due to the computation of Wasserstein distances which must be computed for each individual. If c is the time in seconds required for each function evaluations, the balance between the two algorithm is given by $250c = 140$. This means that for $c > 0.56$ s MOEA/WST is better also in terms of wall-clock time. This computation has been performed for MovieLens 100 k. In a conservative assumption the value c scales linearly with the number of users which means that MovieLens 1 M would give an advantage to MOEA/WST.

9 Conclusions and perspectives

The main conclusion is that embedding in a probabilistic space both the data model and the optimization algorithm in the design of RSs is a novel and potentially useful approach. The elements of this space are histograms which capture in a low dimensional space using a limited number of features the interactions between users and items. The Wasserstein distance is particularly suitable as measure of similarity between histograms. The Wasserstein distance also enables a novel graph representation of the rating matrix: the nodes are the users, and the weights of the edges are the WST distance between users.

The second key result is that a candidate top k-list is represented as a multidimensional histogram which encodes the knowledge obtained from the evaluation of all the objectives function evaluations. The difference between 2 lists can be encoded as the WST distance between their histograms. This enables to define a WST based selection operators in MOEA.

The resulting algorithm MOEA/WST results in better hypervolume and coverage than NSGA-II. The advantage is larger at low generation counts meaning that MOEA/WST is particularly useful when the evaluation of the objectives is even moderately expensive. Moreover, comparing histograms through the Wasserstein distance enables a better visualization. This potential has been largely realized for images but has general applications. As we have remarked before, WST between histograms or point clouds, considered as instances of probability measures, is defined as the smallest cost required to move one measure to another. Because the Wasserstein distance is geodesic when the ground metric is geodesic Optimal transport can be used to compute interpolation between two measures, which is the shortest path in the probability simplex that connects the two measures as end-points. The interpolation process describes a series of intermediate measures during the transport process. We have not considered in this paper the use of the Fréchet mean, also called Wasserstein barycenter, of a set of measures. This is shape-preserving and offers a better synthesis of the set of distributions than the Euclidean distance. The barycenters can be seen as centroids in a clustering process which can be structured as the standard k-means using the Wasserstein distance between elements. The Wasserstein distance, thanks to its linear programming basis, has emerged as a main tool in economic analysis (Galichon 2021) where the issues of labour economics, trade and derivatives pricing have been considered.

The main conclusion of this paper is that the Wasserstein distance is a convenient way to describe complex or high dimensional objects, allowing to reparametrize them so as to work with a reduced number of features. These results have implications, beyond the domain of RSs, for those situations as for instance computer experiments or simulation/optimization, in which associating a distribution to the inputs might be more effective than merely comparing a set of parametric features such as the mean or the higher moments. Indeed, the WST distance takes the whole distribution into account and matches naturally the perceptual notions of nearness and similarity.

Acknowledgements The contribution of DEMS Data Science Lab for supporting this work by providing computational resources (DEMS—Department of Economics, Management and Statistics) is gratefully acknowledged.

Author contributions All authors contributed equally to the paper.

Funding Open access funding provided by Università degli Studi di Milano - Bicocca within the CRUI-CARE Agreement.

Data availability statement All the data related to this paper are available in MovieLens.

Declarations

Conflicts of interest The authors declare no conflict of interest.

References

Abdollahpouri H, Mansoury M, Burke R, Mobasher B (2019) The unfairness of popularity bias in recommendation. arXiv preprint arXiv:1907.13286

Aitchison J (1982) The statistical analysis of compositional data. J R Stat Soc: Series B (Methodologic) 44(2):139–160

Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: International conference on machine learning. PMLR, pp 214–223

Atasu K, Mittelholzer T (2019) Linear-complexity data-parallel earth mover's distance approximations. In: International Conference on machine learning. PMLR, pp 364–373

Backurs A, Dong Y, Indyk P, Razenshteyn I, Wagner T (2020) Scalable nearest neighbour search for optimal transport. In: International Conference on machine learning, vol 119. PMLR, pp 497–506

Balaji TK, Annavarapu CSR, Bablani A (2021) Machine learning algorithms for social media analysis: a survey. Comput Sci Rev 40:100395

Beugnot G, Genevay A, Greenewald K, Solomon J (2021) Improving approximate optimal transport distances using quantization. In: de Campos CP, Maathuis MH, Quaeghebeur E (eds) Uncertainty in artificial intelligence, vol 161. AUAI Press, pp 290–300

Blank J, Deb K (2020) Pymoo: Multi-objective optimization in python. IEEE Access 8:89497–89509

Bonneel N, Peyré G, Cuturi M (2016) Wasserstein barycentric coordinates: histogram regression using optimal transport. ACM Trans Graph 35(4):71–81

Castells P, Hurley NJ, Vargas S (2015) Novelty and diversity in recommender systems. In: Ricci F, Rokach L, Shapira B (eds) Recommender systems handbook. Springer, Boston, pp 881–918

Christakopoulou K, Banerjee A (2018) Learning to interact with users: A collaborative-bandit approach. In: Proceedings of the 2018 SIAM International Conference on Data Mining, vol 2018. Society for Industrial and Applied Mathematics, pp 612–620

Cuturi M, Avis D (2014) Ground metric learning. J Mach Learn Res 15(1):533–564

Daulton S, Balandat M, Bakshy E (2020) Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization. arXiv preprint arXiv:2006.05078

Flamary R, Courty N, Gramfort A, Alaya MZ, Boisbunon A, ChambonVayer ST (2021) Pot: Python optimal transport. J Mach Learn Res 22(78):1–8

Galichon A (2021) The unreasonable effectiveness of optimal transport in economics. arXiv preprint arXiv:2107.04700

Galuzzi BG, Giordani I, Candelieri A, Perego R, Archetti F (2020) Hyperparameter optimization for recommender systems through Bayesian optimization. CMS 17(4):495–515

Gentile C, Li S, Kar P, Karatzoglou A, Zappella G, Etrue E (2017) On context-dependent clustering of bandits. In: International Conference on machine learning, vol 70. PMLR, pp 1253–1262

Gillis N, Leplat V, Tan V (2021) Distributionally robust and multiobjective nonnegative matrix factorization. IEEE Trans Pattern Anal Mach Intell 44:4052–4064

Guillou F, Gaudel R, Preux P (2015) Collaborative filtering as a multi-armed bandit. In: NIPS'15 Workshop: Machine Learning for eCommerce

Heitz M, Bonneel N, Coeurjolly D, Cuturi M, Peyré G (2021) Ground metric learning on graphs. J Math Imaginf vis 63(1):89–107

Hejazinia M, Eastman K, Ye S, Amirabadi A, Divvela R (2019) Accelerated learning from recommender systems using multi-armed bandit. arXiv preprint arXiv:1908.06158

Indyk P, Vakilian A, Yuan Y (2019) Learning-based low-rank approximations. Adv Neural Inf Process Syst 32:7400–7410

Kantorovich L (1942) On the transfer of masses (in Russian). In: Doklady Akademii Nauk. pp 227–229

Le T, Cuturi M (2015) Adaptive Euclidean maps for histograms: generalized Aitchison embeddings. Mach Learn 99(2):169–187

Li B, Qian C, Li J, Tang K, Yao X (2016) Search based recommender system using many-objective evolutionary algorithm. In: 2016 IEEE Congress on Evolutionary Computation (CEC), vol 2016. IEEE, pp 120–126

Li R, Qian F, Du X, Zhao S, Zhang Y (2020) A collaborative filtering recommendation framework based on Wasserstein GAN. J Phys Conf Ser 1864(1):012057

Lin Q, Wang X, Hu B, Ma L, Chen F, Li J, Coello Coello CA (2018) Multiobjective personalized recommendation algorithm using extreme point guided evolutionary computation. Complexity 2018:1716352–1–1716352–18

Lin X, Zhen HL, Li Z, Zhang QF, Kwong S (2019) Pareto multi-task learning. Adv Neural Inf Process Syst 32:12060–12070

Ma C, Ma L, Zhang Y, Tang R, Liu X, Coates M (2020).Probabilistic metric learning with adaptive margin for top-k recommendation. In: Proceedings of the 26th ACM SIGKDD International Conference on knowledge discovery & data mining, pp 1036–1044

Meng Y, Yan X, Liu W, Wu H, Cheng J (2020) Wasserstein collaborative filtering for item cold-start recommendation. In: Proceedings of the 28th ACM Conference on user modeling, adaptation and personalization, vol 2020, pp 318–322

Monge G (1781) Mémoire sur la théorie des déblais et des remblais. Histoire de l'Académie Royale des Sciences de Paris

Nguyen TV, Karatzoglou A, Baltrunas L (2014) Gaussian process factorization machines for context-aware recommendations. In: Proceedings of the 37th international ACM SIGIR Conference on research & development in information retrieval, vol 1, pp 63–72

Öcal K, Grima R, Sanguinetti G (2019) Parameter estimation for biochemical reaction networks using Wasserstein distances. J Phys A Math Theor 53(3):034002

Peyré G, Cuturi M (2019) Computational optimal transport: with applications to data science. Found Trends® Mach Learn 11(5–6):355–607

Ponti A, Candelieri A, Archetti F (2021a) A new evolutionary approach to optimal sensor placement in water distribution networks. Water 13(12):1625

Ponti A, Candelieri A, Archetti F (2021b) A Wasserstein distance based multiobjective evolutionary algorithm for the risk aware optimization of sensor placement. Intell Syst Appl 10:200047

Rakotomamonjy A, Traoré A, Berar M, Flamary R, Courty N (2018) Distance measure machines. arXiv preprint arXiv:1803.00250

Ribeiro MT, Ziviani N, Moura ESD, Hata I, Lacerda A, Veloso A (2014) Multiobjective pareto-efficient approaches for recommender systems. ACM Trans Intellt Syst Technol (TIST) 5(4):1–20

Si N, Blanchet J, Ghosh S, Squillante M (2020) Quantifying the empirical Wasserstein distance to a set of measures: beating the curse of dimensionality. Adv Neural Inf Process Syst 33:21260–21270

Solomon J, Rustamov R, Guibas L, Butscher A (2014) Wasserstein propagation for semi-supervised learning. In: International Conference on machine learning, vol 32. PMLR, pp 306–314

Vanchinathan HP, Nikolic I, De Bona F, Krause A (2014) Explore-exploit in top-n recommender systems via Gaussian processes. In: Proceedings of the 8th ACM Conference on Recommender systems, vol 2014, pp 225–232

Villani C (2009) Optimal transport: old and new, vol 338. Springer, Berlin, p 23

Weinberger KQ, Saul LK (2009) Distance metric learning for large margin nearest neighbor classification. J Mach Learn Res 10(2):207–244

Wang C, Liu Q, Wu R, Chen E, Liu C, Huang X, Huang Z (2018a) Confidence-aware matrix factorization for recommender systems. In: Proceedings of the AAAI Conference on artificial intelligence, Vol. 32, No. 1, pp 434–442

Wang Q, Zeng C, Zhou W, Li T, Iyengar SS, Shwartz L, Grabarnik GY (2018b) Online interactive collaborative filtering using multi-armed bandit with dependent arms. IEEE Trans Knowl Data Eng 31(8):1569–1580

Zhang S, Yao L, Tay Y, Xu X, Zhang X, Zhu L (2018) Metric factorization: recommendation beyond matrix factorization. arXiv preprint arXiv:1802.04606

Zhang X, Zhong J, Liu K (2021) Wasserstein autoencoders for collaborative filtering. Neural Comput Appl 33(7):2793–2802

Zhao Z, Nie J, Wang C, Huang L (2021) Sliced Wasserstein based canonical correlation analysis for cross-domain recommendation. Pattern Recogn Lett 150:33–39

Zheng Y, Agnani M, Singh M (2017). Identification of grey sheep users by histogram intersection in recommender systems. In: International Conference on advanced data mining and applications, vol 10604. Springer, Cham, pp 148–161

Zhou T, Kuscsik Z, Liu JG, Medo M, Wakeling JR, Zhang YC (2010) Solving the apparent diversity-accuracy dilemma of recommender systems. Proc Natl Acad Sci 107(10):4511–4515

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.