Mean–variance portfolio optimization with shrinkage estimation for recommender systems

Yuta Yasumoto[a], Yuichi Takano[b,*]

[a] *Graduate School of Science and Technology, University of Tsukuba, 1-1-1 Tennodai, Tsukuba-shi, 305-8573, Ibaraki, Japan*
[b] *Institute of Systems and Information Engineering, University of Tsukuba, 1-1-1 Tennodai, Tsukuba-shi, 305-8573, Ibaraki, Japan*

Abstract

This paper is concerned with a mean–variance portfolio optimization model with cardinality constraint for generating high-quality lists of recommendations. It is usually difficult to accurately estimate the rating covariance matrix required for mean–variance portfolio optimization because of a shortage of observed user ratings. To improve the accuracy of covariance matrix estimation, we apply shrinkage estimation methods that compute the weighted sum of the target and sample covariance matrices, and we propose two types of target matrices that work well for shrinkage estimation of the rating covariance matrix. Experimental results show that with appropriate parameter tuning, our method can improve the quality of recommendation lists produced by various collaborative filtering algorithms.

Keywords: recommendation, portfolio optimization, shrinkage estimation, collaborative filtering, diversity

1. Introduction

1.1. Background

Dramatic advances in information and communication technology have allowed us to browse and purchase a huge variety of items on e-commerce websites, but users of those websites often find it very difficult to select an

*Corresponding author

appropriate item from the plethora of choices presented to them. Recommender systems are aimed at resolving this problem of information overload by providing a personalized list of items that are unknown but attractive to each user (Adomavicius & Tuzhilin, 2005; Aggarwal et al., 2016; Bobadilla et al., 2013). In fact, effective implementations of recommender systems can be found in various online services (Lu et al., 2015), such as Amazon (Linden et al., 2003), Netflix (Bennett et al., 2007), Google News (Das et al., 2007), and YouTube (Davidson et al., 2010).

One of the most successful technologies for recommender systems is collaborative filtering (Su & Khoshgoftaar, 2009), in which user-based neighborhood models give personalized recommendations to a target user by analyzing the rating data of some other users whose preferences are similar to those of the target user (Herlocker et al., 1999). Item-based neighborhood models recommend items that are similar to a target item in terms of user rating data (Sarwar et al., 2001), and matrix factorization techniques predict user ratings by calculating the inner product of user and item factor vectors (Koren et al., 2009). It has been shown that matrix factorization techniques achieve good prediction accuracy in top-N recommendation tasks (Cremonesi et al., 2010), and deep learning methods have also been applied recently in research into recommender systems (Zhang et al., 2019).

A major challenge in researching recommender systems is the difficulty of evaluating the validity of recommendations (Herlocker et al., 2004). Although most previous studies were focused on the prediction accuracy evaluated using observed rating data, the most accurate recommendations are not always of the most use to users (McNee et al., 2006). More importantly, recommender systems are required to generate a high-quality list of recommendations (e.g., by top-N recommendation) rather than individual items, in consideration of various interaction effects among candidate items (Adamopoulos, 2013). In the recommender-system community, recommendation diversity is recognized as a key evaluation metric (Vargas & Castells, 2011), and it has been shown that user satisfaction with recommendation lists is positively correlated with the diversity of those recommendations (Ekstrand et al., 2014). In addition, some well-known recommender algorithms that are likely to recommend items that are already popular can lead to reduced sales diversity (Fleder & Hosanagar, 2009).

1.2. Related work

Several previous studies have considered the definition and evaluation of diversity, its impact on recommendation quality, and the development of diversification algorithms (Kaminskas & Bridge, 2016; Kunaver & Požrl, 2017; Logesh et al., 2020; Castells et al., 2022). Ziegler et al. (2005) dealt with decreasing the intra-list similarity of topics in recommendation lists. Lathia et al. (2010) investigated temporal diversity of recommendations to avoid repeatedly recommending the same items to a particular user. Kabutoya et al. (2013) developed a probabilistic diversification model for increasing the probability of recommending at least one relevant item to each user. Vargas et al. (2014) provided a binomial framework for defining the genre diversity of recommendations.

Various diversification algorithms have been proposed for recommender systems, including case-based reasoning systems (Smyth & McClave, 2001), Markov chain models (Javari & Jalili, 2015), and matrix factorization techniques (Gogna & Majumdar, 2017). Adomavicius & Kwon (2012) considered general-purpose item ranking algorithms that can generate diverse recommendations while maintaining a comparable level of recommendation accuracy. Multi-objective optimization algorithms have also been used to account for multiple objectives including diversity in recommender systems (Di Noia et al., 2017; Jambor & Wang, 2010; Ribeiro et al., 2015; Zheng & Wang, 2022).

Here, we address the application of financial portfolio theory (Elton et al., 2009; Kolm et al., 2014) to recommender systems. This approach was introduced into product portfolio decisions (Cardozo & Smith, 1983) and then applied to information retrieval (Wang & Zhu, 2009) and collaborative filtering (Wang, 2009). Kwon (2008) focused on the variance of user ratings for each item to increase the precision of top-N recommendation. Wang & Zhu (2009) applied mean–variance portfolio analysis to document ranking under uncertainty in information retrieval. Wang (2009) demonstrated that this ranking strategy based on mean–variance portfolio analysis can improve the recommendation performance of collaborative filtering algorithms. Shi et al. (2012) improved this ranking strategy by using the latent factor model for user ratings. Zhang & Hurley (2008) and Hurley & Zhang (2011) formulated several optimization models with cardinality constraint for selecting a list of diversified items, and they developed heuristic algorithms for the resultant binary optimization problems. Xiao et al. (2020) used the portfolio optimization model to recommend relevant services based on service risk facets.

As shown from the above, high-quality recommendation lists can be produced by applying mean–variance portfolio optimization to recommender systems. Indeed, previous studies (Wang, 2009; Zhang & Hurley, 2008; Hurley & Zhang, 2011) have shown that the quality of recommendation lists (e.g., their accuracy, diversity, and novelty) can be improved by applying portfolio optimization. However, the covariance matrix of user ratings is required for mean–variance portfolio optimization, but this is usually difficult to estimate closely because of a shortage of observed user ratings, and inadequate accuracy of covariance matrix estimation is a barrier to improving the quality of recommendation lists given by the associated portfolio optimization.

1.3. Our contribution

The goal herein is to establish a computational framework for generating high-quality lists of recommendations based on cardinality-constrained mean–variance portfolio optimization. To obtain a close estimate of the covariance matrix of user ratings, we make effective use of shrinkage estimation techniques (Ledoit & Wolf, 2003, 2004; Chen et al., 2010), which allow us to improve the accuracy of covariance matrix estimation by computing the weighted sum of the target and sample covariance matrices. We propose two types of target matrices that work well for shrinkage estimation of the rating covariance matrix: the first is a diagonal-entries-based target matrix given by moderately shrinking off-diagonal entries of the sample covariance matrix to zero; the second is a matrix-completion-based target matrix defined by computing a covariance matrix after imputing the missing ratings by means of collaborative filtering algorithms.

To assess the efficacy of our method, we performed computational experiments using two publicly available datasets of user ratings. We tested three collaborative filtering algorithms for rating prediction, namely, the user-based neighborhood model, the item-based neighborhood model, and the nonnegative matrix factorization model, then we evaluated the recommendation quality in terms of the accuracy (F1 score), risk (variance), and diversity of the recommendations. The experimental results for both datasets show that with appropriate parameter tuning, our portfolio optimization model based on shrinkage estimation can enhance the quality of recommendation lists produced by various collaborative filtering algorithms.

2. Mean–variance portfolio optimization for recommendation

In this section, we present our mean–variance portfolio optimization model with cardinality constraint for selecting a list of recommendations for each user. Our model is a combination of mean–variance portfolio analysis (Wang, 2009; Wang & Zhu, 2009) and the cardinality-constrained optimization model (Zhang & Hurley, 2008; Hurley & Zhang, 2011) for recommendation.

2.1. User–item rating matrix

Let U and I denote the sets of users and items, respectively. The user–item rating matrix is defined as

$$\boldsymbol{R} := (r_{ui})_{(u,i)\in U\times I} \in \mathbb{R}^{|U|\times|I|},$$

where r_{ui} is the rating given by user $u \in U$ to item $i \in I$. Note that this matrix usually has a large number of missing entries. Let $Q \subseteq U \times I$ be the set of user–item pairs with observed ratings, namely,

$$r_{ui} \text{ is } \begin{cases} \text{given} & \text{if } (u,i) \in Q, \\ \text{missing} & \text{otherwise.} \end{cases}$$

A main purpose of recommender systems is to recommend items that are undiscovered but preferred by each user. For this purpose, collaborative filtering algorithms (e.g., user/item-based neighborhood models and matrix factorization techniques) (Aggarwal et al., 2016) have been proposed for predicting unknown ratings r_{ui} for $(u,i) \notin Q$. In the next subsection, we focus on the portfolio optimization problem for selecting a list of recommendations for each user.

2.2. Objectives of portfolio optimization

Let $\boldsymbol{x} := (x_i)_{i\in I} \in \{0,1\}^{|I|}$ be a vector comprising binary decision variables for selecting recommendations; namely, $x_i = 1$ if item i is recommended, or $x_i = 0$ otherwise. Following financial portfolio theory (Elton et al., 2009; Kolm et al., 2014), we define R_{ui} as a random variable representing an unknown rating of user $u \in U$ for item $i \in I$. We then define a user utility function as the sum of user ratings for recommended items:

$$R_u(\boldsymbol{x}) := \sum_{i\in I} R_{ui} x_i \quad (u \in U).$$

The first objective to be maximized in our portfolio optimization model is the expectation of the user utility:

$$\mathbb{E}[R_u(\boldsymbol{x})] = \mathbb{E}\left[\sum_{i \in I} R_{ui} x_i\right] = \sum_{i \in I} \mu_{ui} x_i \quad (u \in U), \tag{1}$$

where $\mu_{ui} := \mathbb{E}[R_{ui}]$ is the expected rating of user $u \in U$ for item $i \in I$. These expected ratings can be predicted using collaborative filtering algorithms (i.e., $\mu_{ui} \approx \hat{r}_{ui}$), where $\hat{r}_{ui}$ is a rating predicted by a collaborative filtering algorithm for $(u,i) \notin Q$.

The second objective to be minimized for diversification of recommendations is the variance of the user utility:

$$\begin{aligned}
\mathrm{Var}[R_u(\boldsymbol{x})] &= \mathbb{E}\left[(R_u(\boldsymbol{x}) - \mathbb{E}[R_u(\boldsymbol{x})])^2\right] \\
&= \mathbb{E}\left[\left(\sum_{i \in I}(R_{ui} - \mu_{ui})x_i\right)\left(\sum_{j \in I}(R_{uj} - \mu_{uj})x_j\right)\right] \\
&= \sum_{i \in I}\sum_{j \in I} \sigma_{uij} x_i x_j \quad (u \in U),
\end{aligned} \tag{2}$$

where $\sigma_{uij} := \mathbb{E}[(R_{ui} - \mu_{ui})(R_{uj} - \mu_{uj})]$ is the covariance of the ratings of user $u \in U$ for a pair of items $(i,j) \in I \times I$. However, it is virtually impossible to estimate the rating covariance for each user $u \in U$ because the number of items rated by each user is limited. For this reason, we consider the rating covariance matrix common to all users:

$$\boldsymbol{\Sigma} := (\sigma_{ij})_{(i,j) \in I \times I} \in \mathbb{R}^{|I| \times |I|}. \tag{3}$$

2.3. Portfolio optimization model

Let $C_u \subseteq \{i \in I \mid (u,i) \notin Q\}$ be the set of candidate (unrated) items to be recommended for a target user $u \in U$. We can now formulate our cardinality-constrained mean–variance portfolio optimization model for selecting N items for each user $u \in U$ as

$$\text{maximize} \quad (1-\alpha)\sum_{i \in C_u} \hat{r}_{ui} x_i - \alpha \sum_{i \in C_u}\sum_{j \in C_u} \sigma_{ij} x_i x_j \tag{4}$$

$$\text{subject to} \quad \sum_{i \in C_u} x_i = N, \tag{5}$$

$$x_i \in \{0,1\} \quad (i \in C_u), \tag{6}$$

where $\alpha \in [0,1]$ is a hyperparameter of the risk aversion. Eq. (4) is the weighted sum for maximizing the expected utility (1) and minimizing the utility variance (2) based on the covariance matrix (3) for the target user $u \in U$. Eq. (5) is the cardinality constraint for specifying the number of recommended items. Eq. (6) lists binary decision variables for selecting recommendations.

This portfolio optimization model is a convex quadratic optimization problem with binary decision variables, which can be solved exactly using optimization solvers. Note also that this optimization model with $\alpha = 0$ is consistent with the common top-N recommendation (Cremonesi et al., 2010), which selects N items in descending order of predicted user ratings.

3. Shrinkage estimation of the rating covariance matrix

In this section, we propose shrinkage estimation methods specialized for estimating the rating covariance matrix (3).

3.1. Framework of shrinkage estimation

Let U_{ij} be a set of users who rated both items $i, j \in I$. A sample estimate of the covariance matrix (3) is then given by

$$\boldsymbol{S} := (s_{ij})_{(i,j)\in I\times I} \in \mathbb{R}^{|I|\times|I|}, \tag{7}$$

where

$$s_{ij} := \begin{cases} \dfrac{1}{|U_{ij}|} \displaystyle\sum_{u\in U_{ij}} \left(r_{ui} - \mu_i^{(j)}\right)\left(r_{uj} - \mu_j^{(i)}\right) & \text{if } U_{ij} \neq \emptyset, \\ 0 & \text{otherwise} \end{cases} \quad ((i,j) \in I \times I),$$

$$\mu_i^{(j)} := \frac{1}{|U_{ij}|} \sum_{u\in U_{ij}} r_{ui} \quad ((i,j) \in I \times I).$$

However, because the rating matrix is usually very sparse, each sample size (i.e., $|U_{ij}|$) is often insufficient (or zero) for obtaining a close estimate of the covariance matrix.

To remedy this situation, we focus on the shrinkage estimation techniques for covariance matrices (Ledoit & Wolf, 2003, 2004; Chen et al., 2010). Specifically, we begin by defining a target covariance matrix:

$$\boldsymbol{F} := (f_{ij})_{(i,j)\in I\times I} \in \mathbb{R}^{|I|\times|I|}.$$

We then estimate the covariance matrix to be the weighted sum of the sample and target covariance matrices:

$$\boldsymbol{\Sigma} = (1-\delta)\boldsymbol{S} + \delta \boldsymbol{F}, \tag{8}$$

where $\delta \in [0,1]$ is a hyperparameter of the shrinkage estimation. In what follows, we propose two types of target matrices that are effective for shrinkage estimation of the rating covariance matrix.

3.2. Diagonal-entries-based target matrix

Our first target matrix is given by moderately shrinking off-diagonal entries of the sample covariance matrix to zero. This amounts to using the target matrix defined by extracting diagonal entries from the sample covariance matrix (7) as follows:

$$\boldsymbol{F}^{\mathrm{DE}} := (f_{ij}^{\mathrm{DE}})_{(i,j)\in I\times I} \in \mathbb{R}^{|I|\times|I|}, \tag{9}$$

where

$$f_{ij}^{\mathrm{DE}} := \begin{cases} s_{ij} & \text{if } i=j, \\ 0 & \text{otherwise} \end{cases} \quad ((i,j)\in I\times I).$$

The rating covariance matrix is then estimated by substituting the target matrix $\boldsymbol{F} = \boldsymbol{F}^{\mathrm{DE}}$ to the weighted sum (8).

3.3. Matrix-completion-based target matrix

Our second target matrix is defined by computing a covariance matrix after completion of the rating matrix. Specifically, we first impute the missing ratings r_{ui} for $(u,i) \notin Q$ by means of collaborative filtering algorithms, thereby giving a complete version of the rating matrix without missing entries.

By using this complete version of the rating matrix, we calculate the target covariance matrix:

$$\boldsymbol{F}^{\mathrm{MC}} := (f_{ij}^{\mathrm{MC}})_{(i,j)\in I\times I} \in \mathbb{R}^{|I|\times|I|}, \tag{10}$$

where

$$\begin{aligned} f_{ij}^{\mathrm{MC}} &:= \frac{1}{|U|}\sum_{u\in U}(r_{ui}-\mu_i)(r_{uj}-\mu_j) \quad ((i,j)\in I\times I), \\ \mu_i &:= \frac{1}{|U|}\sum_{u\in U} r_{ui} \quad (i\in I). \end{aligned}$$

After that, the rating covariance matrix is estimated by substituting the target matrix $\boldsymbol{F} = \boldsymbol{F}^{\mathrm{MC}}$ to the weighted sum (8).

4. Experimental results

In this section, we report experimental results to assess the efficacy of our method for generating a list of recommendations for each user.

4.1. Experimental design

We used two publicly available datasets of user ratings, namely, the MovieLens[1] and BookCrossing[2] datasets. The MovieLens 100K dataset consists of 100 000 ratings (about 6.3% of all the user–item pairs) of 1682 movies from 943 users on a scale of 1 to 5, with each user having rated 20 or more movies. This dataset was randomly split into training (60%) and testing (40%) datasets of observed ratings. The BookCrossing dataset consists of 1 149 780 ratings of 271 379 books from 278 858 users on a scale of 0 to 10. We deleted the implicit zero-valued ratings and then created 10 datasets by randomly extracting 1000 users who had rated 20 or more books as in the MovieLens dataset; each of those datasets (consisting on average of 56 206 ratings [about 0.20% of all the user–item pairs] of 28 701 books from 1000 users) was then split randomly into training (60%) and testing (40%) datasets of observed ratings. Note that the error bars in Figs. 4–6 represent the standard errors calculated in the 10 trials.

We used the following collaborative filtering algorithms for rating prediction (Aggarwal et al., 2016):

UserN: the user-based neighborhood model;

ItemN: the item-based neighborhood model;

NMF: the nonnegative matrix factorization model.

We implemented these algorithms using the Python `Surprise` library[3] (Hug, 2020) for building and analyzing recommender systems. We used the Pearson correlation coefficient and the cosine similarity as similarity metrics for the user- and item-based neighborhood models, respectively. For both neighborhood models, we set the neighborhood size to five. For the matrix factorization model, we set the number of factors to 100, the learning rate to 0.1, and the regularization weight to 0.01.

[1] `https://grouplens.org/datasets/movielens/`
[2] `http://www2.informatik.uni-freiburg.de/~cziegler/BX/`
[3] `http://surpriselib.com/`

We evaluate the performance of the portfolio optimization model (4)–(6) that uses the following methods for estimating the rating covariance matrix:

DiagEnt(α, δ): our shrinkage estimation method (8) using the diagonal-entries-based target matrix (9);

MatComp(α, δ): our shrinkage estimation method (8) using the matrix-completion-based target matrix (10).

Here, $\alpha \in \{-1.0, -0.8, \dots, 1.0\}$ is the risk aversion parameter, and $\delta \in \{0.00, 0.25, \dots, 1.00\}$ is the matrix shrinkage parameter. Following previous studies (Wang, 2009; Shi et al., 2012), we tested negative values of the risk aversion parameter. Recall that $\alpha = 0$ corresponds to the common top-N recommendation (Cremonesi et al., 2010), which selects N items in descending order of predicted user ratings. The binary optimization problem (4)–(6) was solved using the optimization solver Gurobi Optimizer[4]. For the matrix-completion-based method, missing ratings were imputed using the item-based neighborhood model for collaborative filtering.

In the experiments, we considered the task of top-N recommendation with $N = 5$. To prevent a large increase in the computation time, we chose the top 50 items in the predicted ratings for each user $u \in U$ as a set of candidate items (i.e., $|C_u| = 50$) from those that were not rated by user u in the training dataset. Next, we solved the portfolio optimization model (4)–(6) to calculate $I_u^N := \{i \in C_u \mid x_i = 1\}$, a set of N items to be recommended to user u.

Herein, we evaluate the recommendation quality in terms of the accuracy, risk, and diversity of the recommendations. Let I_u^{Rel} be a set of relevant items that are highly rated by each user $u \in U$ in the testing dataset. Following Adomavicius & Kwon (2012), the relevant items are defined as those obtaining ratings of not less than 3.5 and 7.0 in the MovieLens and BookCrossing testing datasets, respectively.

The recommendation accuracy is evaluated by comparing the recommended item set I_u^N and the relevant item set I_u^{Rel}. We use the (average) F1 score as a measure of recommendation accuracy:

$$\textbf{F1 score} := \frac{1}{|U|} \sum_{u \in U} \frac{2 \cdot \mathrm{Recall}_u \cdot \mathrm{Precision}_u}{\mathrm{Recall}_u + \mathrm{Precision}_u}, \tag{11}$$

[4]`https://www.gurobi.com/`

where the recall and precision for each user are defined as

$$\text{Recall}_u := \frac{|I_u^N \cap I_u^{\text{Rel}}|}{|I_u^{\text{Rel}}|}, \quad \text{Precision}_u := \frac{|I_u^N \cap I_u^{\text{Rel}}|}{|I_u^N|} \quad (u \in U).$$

We seek to maintain a certain level of recommendation accuracy for all users. For this reason, we analyze the variance of the number of recommended relevant items (i.e., $|I_u^N \cap I_u^{\text{Rel}}|$) as a risk measure of recommendations:

$$\textbf{Variance} := \frac{1}{|U|} \sum_{u \in U} (|I_u^N \cap I_u^{\text{Rel}}| - m)^2, \tag{12}$$

where

$$m := \frac{1}{|U|} \sum_{u \in U} |I_u^N \cap I_u^{\text{Rel}}|.$$

Finally, as did Adomavicius & Kwon (2012), we use the total number of different recommended items across all users as a measure of recommendation diversity:

$$\textbf{Diversity} := \left| \bigcup_{u \in U} I_u^N \right|. \tag{13}$$

4.2. Results for the MovieLens dataset

Fig. 1 shows the F1 score (11) as a measure of recommendation accuracy for the MovieLens dataset. With both shrinkage estimation methods, the F1 scores increased with the matrix shrinkage parameter δ, meaning that our target covariance matrices were effective in improving the recommendation accuracy. With $\delta = 1.0$, the diagonal-entries-based method often outperformed the matrix-completion-based method slightly in the F1 score. Moreover, the diagonal-entries-based method with $\delta = 1.0$ for the risk aversion parameter $\alpha \in \{0.2, 0.4, 0.6, 0.8\}$ often attained higher F1 scores than those for the common top-N recommendation ($\alpha = 0$).

Fig. 2 shows the variance (12) as a risk measure of recommendations for the MovieLens dataset. With both shrinkage estimation methods, the variance tended to decrease with the risk aversion parameter α, and the associated decrease rate increased with decreasing matrix shrinkage parameter δ. When α was positive, the matrix-completion-based method always attained smaller variances than those for the common top-N recommendation

($\alpha = 0$), whereas the diagonal-entries-based method with large δ sometimes failed to decrease the variance.

Fig. 3 shows the diversity (13) (i.e., the total number of different recommended items) for the MovieLens dataset. With both shrinkage estimation methods, the diversity increased sharply with decreasing matrix shrinkage parameter δ.

These results show that with appropriate parameter values (e.g., $(\alpha, \delta) = (0.4, 1.0)$), our shrinkage estimation methods can improve both the F1 score and variance simultaneously for the MovieLens dataset. Moreover, our methods can also improve the diversity with low values of the matrix shrinkage parameter δ.

4.3. Results for the BookCrossing dataset

Fig. 4 shows the F1 score (11) as a measure of recommendation accuracy for the BookCrossing dataset. As in Fig. 1 for the MovieLens dataset, the F1 scores with both shrinkage estimation methods increased with increasing matrix shrinkage parameter δ. Notably, both methods with $\delta = 1.0$ for positive values of the risk aversion parameter α substantially outperformed the common top-N recommendation ($\alpha = 0$). Unlike in Fig. 1 for the MovieLens dataset, the matrix-completion-based method slightly outperformed the diagonal-entries-based method in the F1 score when $\delta = 1.0$. Additionally, the F1 score increased monotonically with the risk aversion parameter α only for the nonnegative matrix factorization model (NMF).

Fig. 5 shows the variance (12) as a risk measure of recommendations for the BookCrossing dataset. As in Fig. 2 for the MovieLens dataset, both shrinkage estimation methods with positive α often attained smaller variances than those for the common top-N recommendation ($\alpha = 0$). When the risk aversion parameter α was positive, the variance decreased with decreasing δ for the diagonal-entries-based method and with increasing δ for the matrix-completion-based method. The decrease in the variance was larger with the matrix-completion-based method than with the diagonal-entries-based method, and the variance decreased monotonically with the risk aversion parameter α only for the nonnegative matrix factorization model (NMF).

Fig. 6 shows the diversity (13) (i.e., the total number of different recommended items) for the BookCrossing dataset. Unlike in Fig. 3 for the MovieLens dataset, the diversity provided by both shrinkage estimation methods increased monotonically with the risk aversion parameter α for the neighborhood models (UserN and ItemN), whereas the increase in the diversity was limited for the nonnegative matrix factorization model (NMF).

These results confirm that our shrinkage estimation methods with positive values of the risk aversion parameter α can often improve the F1 score, variance, and diversity simultaneously for the BookCrossing dataset.

5. Conclusion

This paper presented a mean–variance portfolio optimization model with cardinality constraint for generating a high-quality list of recommendations for each user. This portfolio optimization model can be regarded as a combination of mean–variance portfolio analysis (Wang, 2009; Wang & Zhu, 2009) and the cardinality-constrained optimization model (Zhang & Hurley, 2008; Hurley & Zhang, 2011) for recommendation. However, it is usually difficult to accurately estimate the rating covariance matrix required for mean–variance portfolio optimization because of a shortage of observed user ratings. To improve the accuracy of covariance matrix estimation, we applied shrinkage estimation methods that compute the weighted sum of the target and sample covariance matrices. Moreover, we proposed two types of target matrices that work well for shrinkage estimation of the rating covariance matrix.

We conducted computational experiments using the two publicly available datasets of user ratings. For the MovieLens dataset, our method with appropriate tuning of the risk aversion and matrix shrinkage parameters improved both the F1 score and variance simultaneously, and it improved the diversity with low values of the matrix shrinkage parameter. For the BookCrossing dataset, our method with positive values of the risk aversion parameter often improved the F1 score, variance, and diversity simultaneously. This study opens up new possibilities for applying financial portfolio theory to recommender systems. Importantly, our method has the potential to improve the recommendation quality of various collaborative filtering algorithms (e.g., user/item-based neighborhood models and matrix factorization techniques).

A future research direction will be to devise a method for estimating the rating covariance given that it differs from user to user. Other directions will be to design scalable algorithms (Bertsimas & Cory-Wright, 2022; Kobayashi et al., 2021, 2023) for solving our cardinality-constrained portfolio optimization problems for recommendation, and to improve the recommendation quality further by exploiting implicit user feedback via clickstream data (Iwanaga et al., 2019).

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP21K04526.

References

Adamopoulos, P. (2013). Beyond rating prediction accuracy: On new perspectives in recommender systems. In *Proceedings of the 7th ACM Conference on Recommender Systems* RecSys '13 (p. 459–462). New York, NY, USA: Association for Computing Machinery. doi:10.1145/2507157.2508073.

Adomavicius, G., & Kwon, Y. (2012). Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, *24*, 896–911. doi:10.1109/TKDE.2011.15.

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, *17*, 734–749. doi:10.1109/TKDE.2005.99.

Aggarwal, C. C. et al. (2016). *Recommender systems: The textbook*. Springer. doi:10.1007/978-3-319-29659-3.

Bennett, J., Lanning, S. et al. (2007). The netflix prize. In *Proceedings of KDD Cup and Workshop*. New York.

Bertsimas, D., & Cory-Wright, R. (2022). A scalable algorithm for sparse portfolio selection. *INFORMS Journal on Computing*, *34*, 1489–1511. doi:10.1287/ijoc.2021.1127.

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, *46*, 109–132. doi:10.1016/j.knosys.2013.03.012.

Cardozo, R. N., & Smith, D. K. (1983). Applying financial portfolio theory to product portfolio decisions: An empirical study. *Journal of Marketing*, *47*, 110–119. doi:10.1177/002224298304700211.

Castells, P., Hurley, N., & Vargas, S. (2022). Novelty and diversity in recommender systems. In F. Ricci, L. Rokach, & B. Shapira (Eds.), *Recommender Systems Handbook* (pp. 603–646). New York, NY: Springer US. doi:10.1007/978-1-0716-2197-4_16.

Chen, Y., Wiesel, A., Eldar, Y. C., & Hero, A. O. (2010). Shrinkage algorithms for MMSE covariance estimation. *IEEE Transactions on Signal Processing*, *58*, 5016–5029. doi:10.1109/TSP.2010.2053029.

Cremonesi, P., Koren, Y., & Turrin, R. (2010). Performance of recommender algorithms on top-N recommendation tasks. In *Proceedings of the Fourth ACM Conference on Recommender Systems* RecSys '10 (p. 39–46). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1864708.1864721.

Das, A. S., Datar, M., Garg, A., & Rajaram, S. (2007). Google news personalization: Scalable online collaborative filtering. In *Proceedings of the 16th International Conference on World Wide Web* WWW '07 (p. 271–280). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1242572.1242610.

Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., & Sampath, D. (2010). The youtube video recommendation system. In *Proceedings of the Fourth ACM Conference on Recommender Systems* RecSys '10 (p. 293–296). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1864708.1864770.

Di Noia, T., Rosati, J., Tomeo, P., & Sciascio, E. D. (2017). Adaptive multi-attribute diversity for recommender systems. *Information Sciences*, *382*, 234–253. doi:10.1016/j.ins.2016.11.015.

Ekstrand, M. D., Harper, F. M., Willemsen, M. C., & Konstan, J. A. (2014). User perception of differences in recommender algorithms. In *Proceedings of the 8th ACM Conference on Recommender Systems* RecSys '14 (p. 161–168). New York, NY, USA: Association for Computing Machinery. doi:10.1145/2645710.2645737.

Elton, E. J., Gruber, M. J., Brown, S. J., & Goetzmann, W. N. (2009). *Modern portfolio theory and investment analysis*. John Wiley & Sons.

Fleder, D., & Hosanagar, K. (2009). Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management Science*, *55*, 697–712. doi:10.1287/mnsc.1080.0974.

Gogna, A., & Majumdar, A. (2017). Diablo: Optimization based design for improving diversity in recommender system. *Information Sciences*, *378*, 59–74. doi:10.1016/j.ins.2016.10.043.

Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* SIGIR '99 (p. 230–237). New York, NY, USA: Association for Computing Machinery. doi:10.1145/312624.312682.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, *22*, 5–53. doi:10.1145/963770.963772.

Hug, N. (2020). Surprise: A python library for recommender systems. *Journal of Open Source Software*, *5*, 2174. doi:10.21105/joss.02174.

Hurley, N., & Zhang, M. (2011). Novelty and diversity in top-N recommendation – Analysis and evaluation. *ACM Transactions on Internet Technology*, *10*. doi:10.1145/1944339.1944341.

Iwanaga, J., Nishimura, N., Sukegawa, N., & Takano, Y. (2019). Improving collaborative filtering recommendations by estimating user preferences from clickstream data. *Electronic Commerce Research and Applications*, *37*, 100877. doi:10.1016/j.elerap.2019.100877.

Jambor, T., & Wang, J. (2010). Optimizing multiple objectives in collaborative filtering. In *Proceedings of the Fourth ACM Conference on Recommender Systems* RecSys '10 (p. 55–62). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1864708.1864723.

Javari, A., & Jalili, M. (2015). A probabilistic model to resolve diversity—Accuracy challenge of recommendation systems. *Knowledge and Information Systems*, *44*, 609–627. doi:10.1007/s10115-014-0779-2.

Kabutoya, Y., Iwata, T., Toda, H., & Kitagawa, H. (2013). A probabilistic model for diversifying recommendation lists. In Y. Ishikawa, J. Li, W. Wang, R. Zhang, & W. Zhang (Eds.), *Web Technologies and Applications* (pp. 348–359). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-37401-2_36.

Kaminskas, M., & Bridge, D. (2016). Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems*, *7*. doi:10.1145/2926720.

Kobayashi, K., Takano, Y., & Nakata, K. (2021). Bilevel cutting-plane algorithm for cardinality-constrained mean-cvar portfolio optimization. *Journal of Global Optimization*, *81*, 493–528. doi:10.1007/s10898-021-01048-5.

Kobayashi, K., Takano, Y., & Nakata, K. (2023). Cardinality-constrained distributionally robust portfolio optimization. *European Journal of Operational Research*, *309*, 1173–1182. doi:10.1016/j.ejor.2023.01.037.

Kolm, P. N., Tütüncü, R., & Fabozzi, F. J. (2014). 60 years of portfolio optimization: Practical challenges and current trends. *European Journal of Operational Research*, *234*, 356–371. doi:10.1016/j.ejor.2013.10.060.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, *42*, 30–37. doi:10.1109/MC.2009.263.

Kunaver, M., & Požrl, T. (2017). Diversity in recommender systems—A survey. *Knowledge-Based Systems*, *123*, 154–162. doi:10.1016/j.knosys.2017.02.009.

Kwon, Y. (2008). Improving top-N recommendation techniques using rating variance. In *Proceedings of the 2008 ACM Conference on Recommender Systems* RecSys '08 (p. 307–310). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1454008.1454059.

Lathia, N., Hailes, S., Capra, L., & Amatriain, X. (2010). Temporal diversity in recommender systems. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (p. 210–217). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1835449.1835486.

Ledoit, O., & Wolf, M. (2003). Improved estimation of the covariance matrix of stock returns with an application to portfolio selection. *Journal of Empirical Finance*, *10*, 603–621. doi:10.1016/S0927-5398(03)00007-0.

Ledoit, O., & Wolf, M. (2004). Honey, I shrunk the sample covariance matrix. *The Journal of Portfolio Management*, *30*, 110–119. doi:10.3905/jpm.2004.110.

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, *7*, 76–80. doi:10.1109/MIC.2003.1167344.

Logesh, R., Subramaniyaswamy, V., Vijayakumar, V., Gao, X.-Z., & Wang, G.-G. (2020). Hybrid bio-inspired user clustering for the generation of diversified recommendations. *Neural Computing and Applications*, *32*, 2487–2506. doi:10.1007/s00521-019-04128-6.

Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments: A survey. *Decision Support Systems*, *74*, 12–32. doi:10.1016/j.dss.2015.03.008.

McNee, S. M., Riedl, J., & Konstan, J. A. (2006). Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems* CHI EA '06 (p. 1097–1101). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1125451.1125659.

Ribeiro, M. T., Ziviani, N., Moura, E. S. D., Hata, I., Lacerda, A., & Veloso, A. (2015). Multiobjective pareto-efficient approaches for recommender systems. *ACM Transactions on Intelligent Systems and Technology*, *5*. doi:10.1145/2629350.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web* WWW '01 (p. 285–295). New York, NY, USA: Association for Computing Machinery. doi:10.1145/371920.372071.

Shi, Y., Zhao, X., Wang, J., Larson, M., & Hanjalic, A. (2012). Adaptive diversification of recommendation results via latent factor portfolio. In *Proceedings of the 35th International ACM SIGIR Conference* *on Research and Development in Information Retrieval* SIGIR '12 (p. 175–184). New York, NY, USA: Association for Computing Machinery. doi:10.1145/2348283.2348310.

Smyth, B., & McClave, P. (2001). Similarity vs. diversity. In D. W. Aha, & I. Watson (Eds.), *Case-Based Reasoning Research and Development* (pp. 347–361). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/3-540-44593-5_25.

Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, *2009*. doi:10.1155/2009/421425.

Vargas, S., Baltrunas, L., Karatzoglou, A., & Castells, P. (2014). Coverage, redundancy and size-awareness in genre diversity for recommender systems. In *Proceedings of the 8th ACM Conference on Recommender Systems* RecSys '14 (p. 209–216). New York, NY, USA: Association for Computing Machinery. doi:10.1145/2645710.2645743.

Vargas, S., & Castells, P. (2011). Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the Fifth ACM Conference on Recommender Systems* RecSys '11 (p. 109–116). New York, NY, USA: Association for Computing Machinery. doi:10.1145/2043932.2043955.

Wang, J. (2009). Mean-variance analysis: A new document ranking theory in information retrieval. In M. Boughanem, C. Berrut, J. Mothe, & C. Soule-Dupuy (Eds.), *Advances in Information Retrieval* (pp. 4–16). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-00958-7_4.

Wang, J., & Zhu, J. (2009). Portfolio theory of information retrieval. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* SIGIR '09 (p. 115–122). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1571941.1571963.

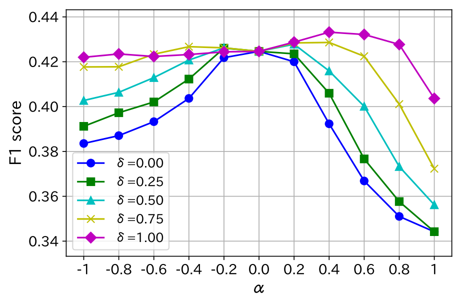
Xiao, Y., Pei, Q., Yao, L., & Wang, X. (2020). Recrisk: An enhanced recommendation model with multi-facet risk control. *Expert Systems with Applications*, *158*, 113561. doi:10.1016/j.eswa.2020.113561.

Zhang, M., & Hurley, N. (2008). Avoiding monotony: Improving the diversity of recommendation lists. In *Proceedings of the 2008 ACM Conference on Recommender Systems* RecSys '08 (p. 123–130). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1454008.1454030.

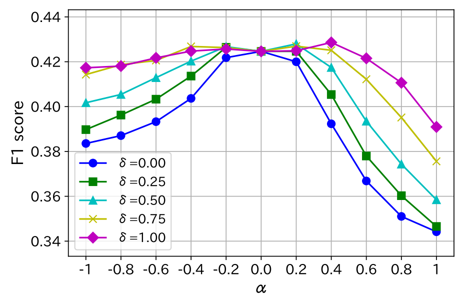
Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, *52*. doi:10.1145/3285029.

Zheng, Y., & Wang, D. X. (2022). A survey of recommender systems with multi-objective optimization. *Neurocomputing*, *474*, 141–153. doi:10.1016/j.neucom.2021.11.041.

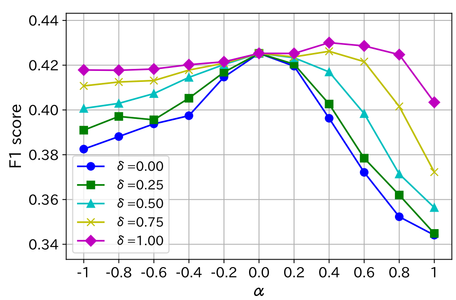
Ziegler, C.-N., McNee, S. M., Konstan, J. A., & Lausen, G. (2005). Improving recommendation lists through topic diversification. In *Proceedings of the 14th International Conference on World Wide Web* WWW '05 (p. 22–32). New York, NY, USA: Association for Computing Machinery. doi:10.1145/1060745.1060754.



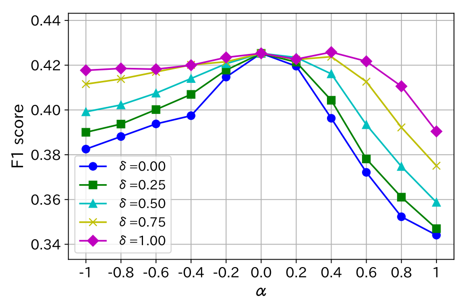
(a) UserN, DiagEnt(α, δ)



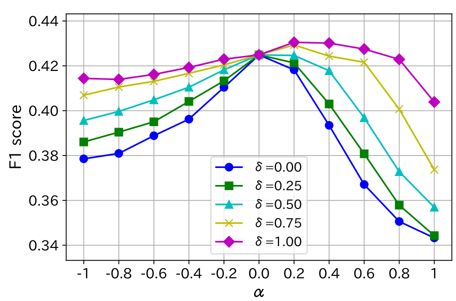
(b) UserN, MatComp(α, δ)



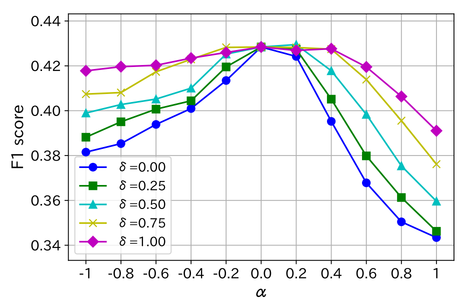
(c) ItemN, DiagEnt(α, δ)



(d) ItemN, MatComp(α, δ)

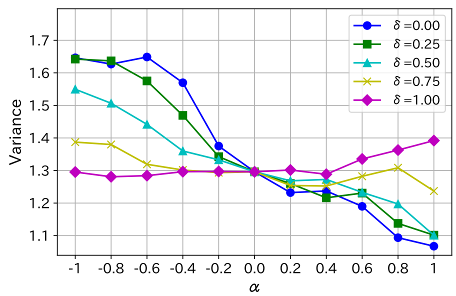


(e) NMF, DiagEnt(α, δ)

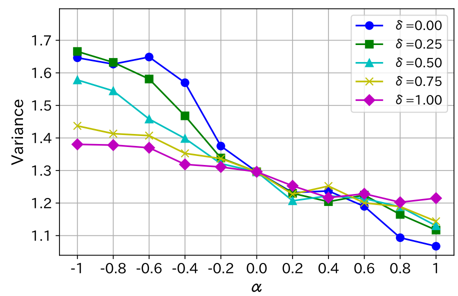


(f) NMF, MatComp(α, δ)

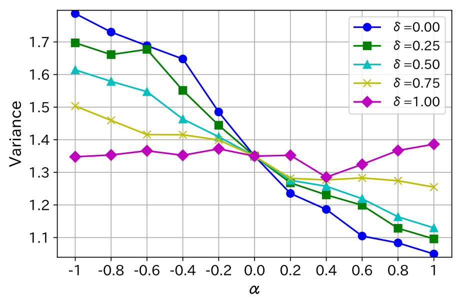
Figure 1: F1 score for the MovieLens dataset.



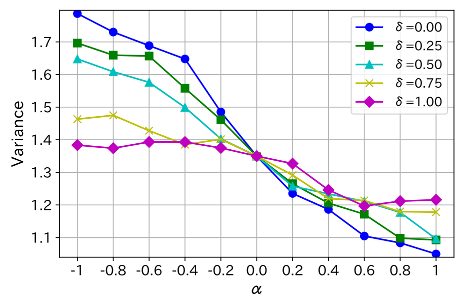
(a) UserN, DiagEnt(α, δ)



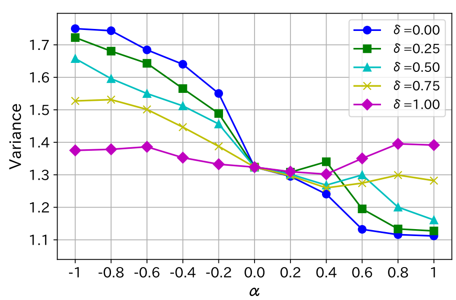
(b) UserN, MatComp(α, δ)



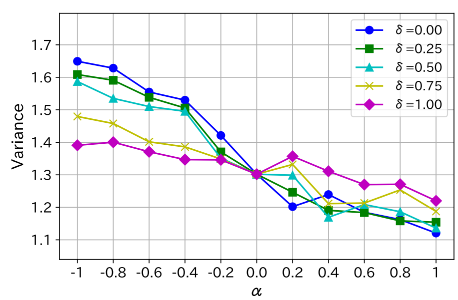
(c) ItemN, DiagEnt(α, δ)



(d) ItemN, MatComp(α, δ)

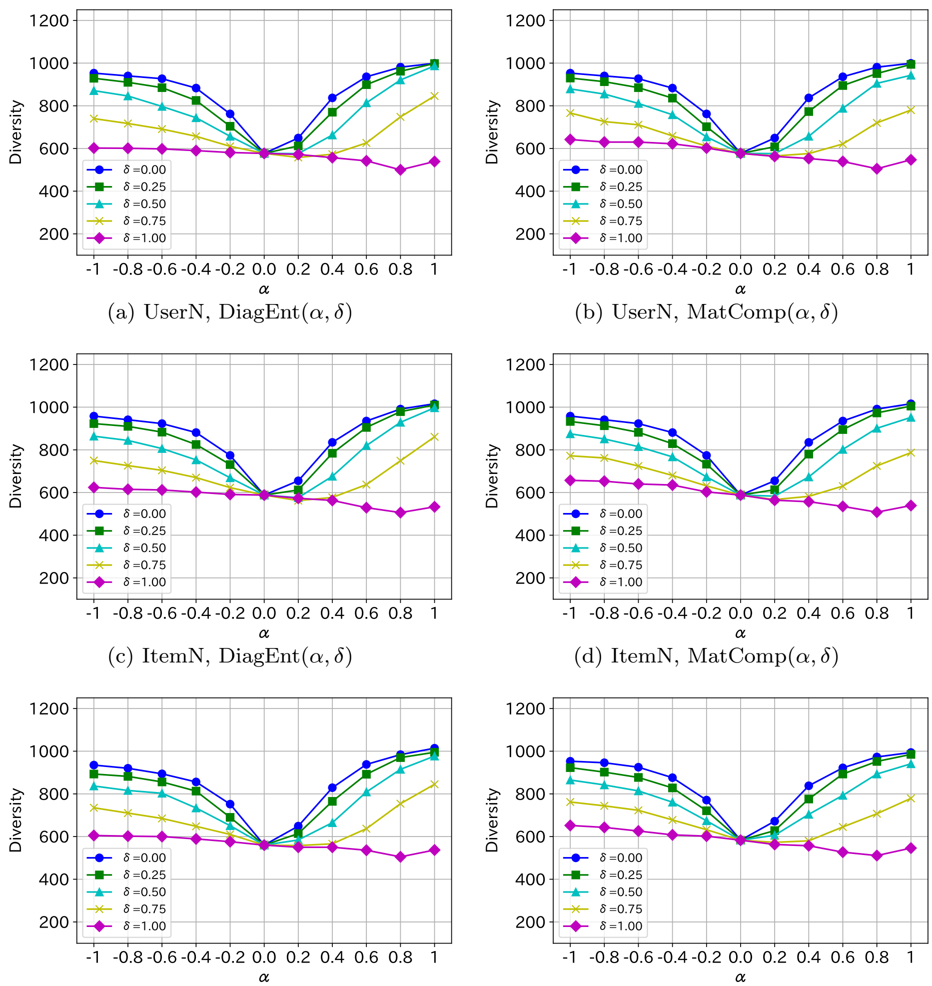


(e) NMF, DiagEnt(α, δ)



(f) NMF, MatComp(α, δ)

Figure 2: Variance for the MovieLens dataset.



(a) UserN, DiagEnt(α, δ)

(b) UserN, MatComp(α, δ)

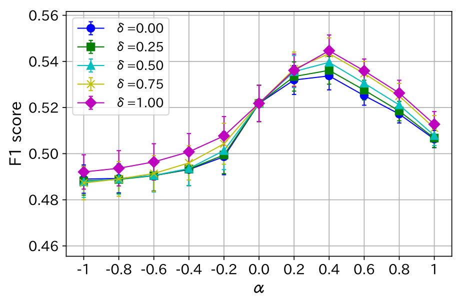
(c) ItemN, DiagEnt(α, δ)

(d) ItemN, MatComp(α, δ)

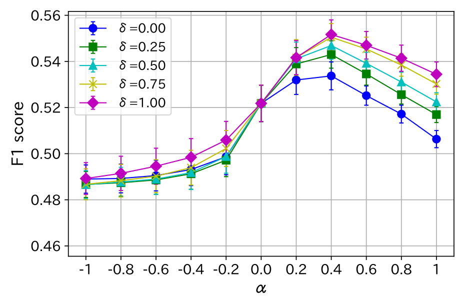
(e) NMF, DiagEnt(α, δ)

(f) NMF, MatComp(α, δ)

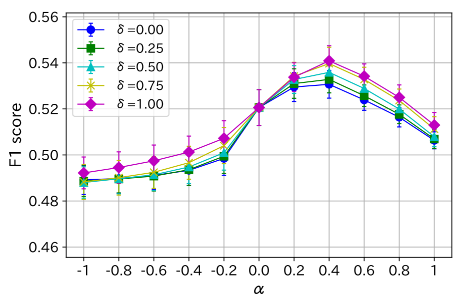
Figure 3: Diversity for the MovieLens dataset.



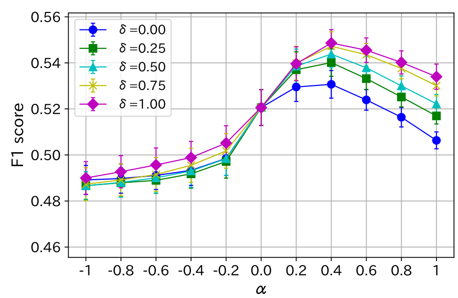
(a) UserN, DiagEnt(α, δ)



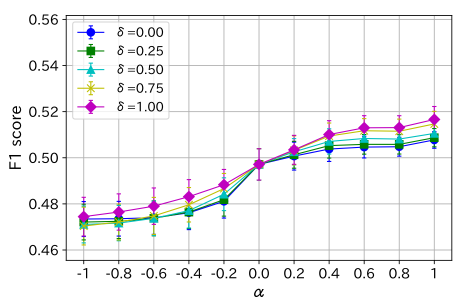
(b) UserN, MatComp(α, δ)



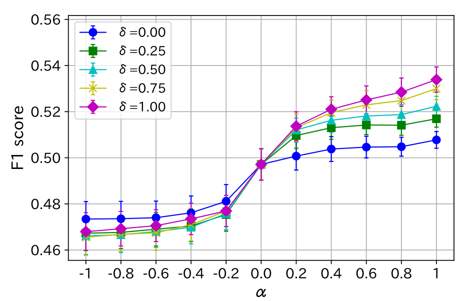
(c) ItemN, DiagEnt(α, δ)



(d) ItemN, MatComp(α, δ)

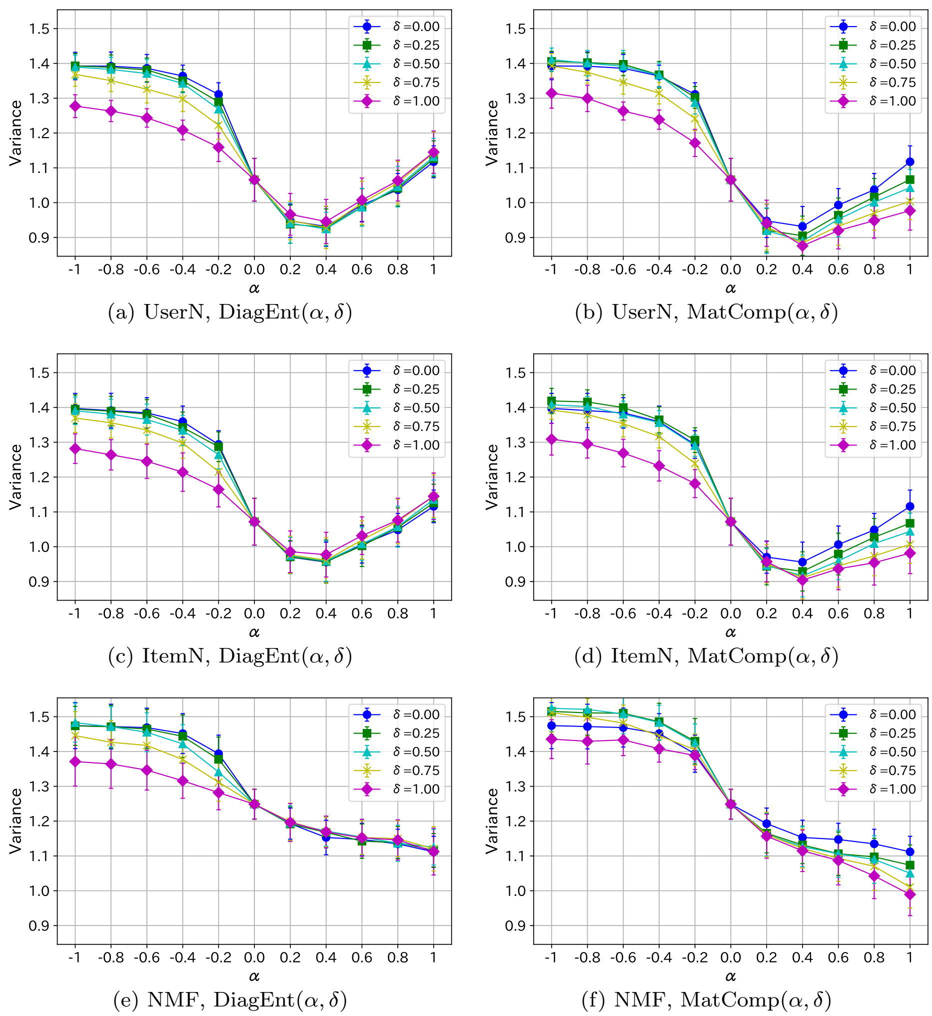


(e) NMF, DiagEnt(α, δ)



(f) NMF, MatComp(α, δ)

Figure 4: F1 score for the BookCrossing dataset.



(a) UserN, DiagEnt(α, δ)

(b) UserN, MatComp(α, δ)

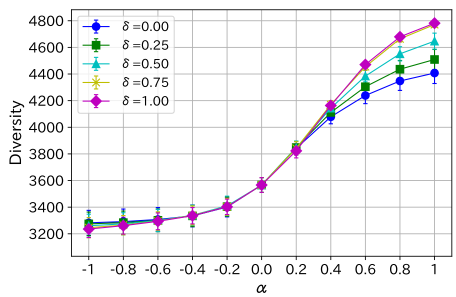
(c) ItemN, DiagEnt(α, δ)

(d) ItemN, MatComp(α, δ)

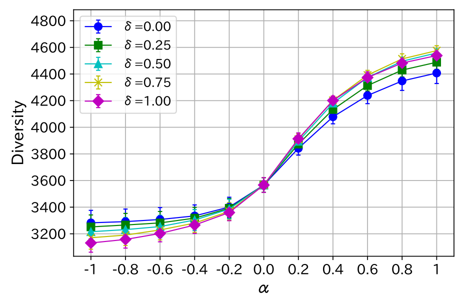
(e) NMF, DiagEnt(α, δ)

(f) NMF, MatComp(α, δ)

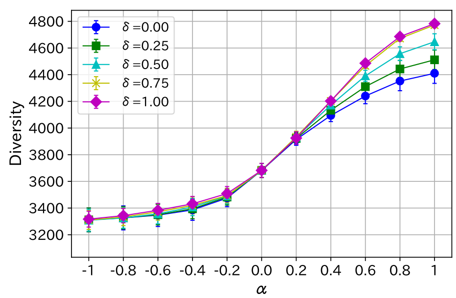
Figure 5: Variance for the BookCrossing dataset.



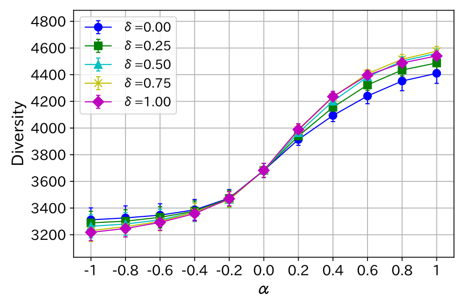
(a) UserN, DiagEnt(α, δ)



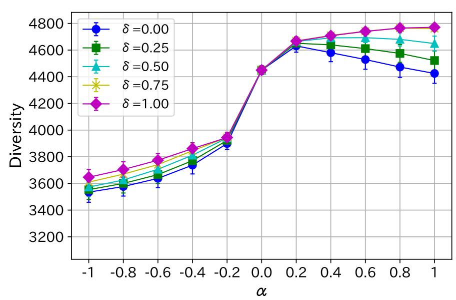
(b) UserN, MatComp(α, δ)



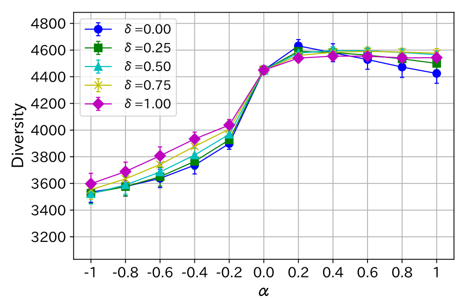
(c) ItemN, DiagEnt(α, δ)



(d) ItemN, MatComp(α, δ)



(e) NMF, DiagEnt(α, δ)



(f) NMF, MatComp(α, δ)

Figure 6: Diversity for the BookCrossing dataset.