idea first use parameter from papers

Mean-Variance Efficient Collaborative Filtering for Stock Recommendation

Mean-Variance Efficient Collaborative Filtering

Munki Chung

Korea Advanced Institute of Science and Technology, Daejeon, South Korea, moonki93@kaist.ac.kr

Yongjae Lee*

Ulsan National Institute of Science and Technology, Ulsan, South Korea, yongjaelee@unist.ac.kr

Woo Chang Kim*

Korea Advanced Institute of Science and Technology, Daejeon, South Korea, wkim@kaist.ac.kr

The rise of FinTech has transformed financial services onto online platforms, yet stock investment recommender systems have received limited attention compared to other industries. Personalized stock recommendations can significantly impact customer engagement and satisfaction within the industry. However, traditional investment recommendations focus on high-return stocks or highly diversified portfolios based on the modern portfolio theory, often neglecting user preferences. The former would result in unsuccessful investment because accurately predicting stock prices is almost impossible, whereas the latter would not be accepted by investors because most individuals tend to possess only a few stocks that they are interested in. On the other hand, collaborative filtering (CF) methods also may not be directly applicable to stock recommendations, because it is inappropriate to just recommend stocks that users like. The key is to optimally blend user's preference with the portfolio theory. However, research on stock recommendations within the recommender system domain remains comparatively limited, and no existing model considers both the preference of users and the risk-return characteristics of stocks. In this regard, we propose a mean-variance efficient collaborative filtering (MVECF) model for stock recommendations that consider both aspects. Our model is specifically designed to improve the pareto optimality (mean-variance efficiency) in a trade-off between the risk (variance of return) and return (mean return) by systemically handling uncertainties in stock prices. Such improvements are incorporated into the MVECF model using regularization, and the model is restructured to fit into the ordinary matrix factorization scheme to boost computational efficiency. Experiments on real-world fund holdings data show that our model can increase the mean-variance efficiency of suggested portfolios while sacrificing just a small amount of mean average precision and recall. Finally, we further show MVECF is easily applicable to the state-of-the-art graph-based ranking models.

CCS CONCEPTS • **Information systems → Collaborative filtering; • Applied computing → Economics**.

Additional Keywords and Phrases: recommender systems, stock recommendation, collaborative filtering, modern portfolio theory

ACM Reference Format:

Anonymous Author(s). 2023. Mean-Variance Efficient Collaborative Filtering for Stock Recommendation. In ACM, New York, NY, USA, 12 pages. https://www.doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

While the rise of FinTech has galvanized the participation of individual investors in financial markets [9], recommender systems for financial services have not been actively studied compared to other industries. Investment management would be the most attention-grabbing service in the finance industry. The necessity for recommender systems for investment is

* Corresponding author

incorporate investor's preference but MVO is too rational for most of customer

in no doubt since individuals are known to hold under-diversified portfolios [3]. However, recommendations of investment products are not trivial due to the high uncertainty in their price. We list *three key requirements* for stock recommendation.

First, stock recommender systems should *incorporate investor's preference*. One may simply recommend stocks that are predicted to have high returns, or recommend well-diversified portfolios constructed based on established portfolio theories. However, such recommendations would be almost irrelevant to the user's appetite, and this is why individuals still hold under-diversified portfolios even though there are a huge number of mutual funds. This is almost the opposite direction from most recommender systems for online services that have been gaining great success from subtle but user-specific recommendations. Hence, stock recommendations should reflect users' preferences that can be inferred from the users' portfolios.

Second, stock recommender systems should *improve portfolio diversification*. Direct applications of CF would probably make users' portfolios more concentrated on some risk factors or industry sectors because recommended stocks would be similar to the users' current holdings. More importantly, user preference does not necessarily lead to good investment. Therefore, stock recommendations should bring diversification benefits while satisfying the users' preferences.

Third, they should be proven in terms of *ex-post performance* evaluations. While wrong recommendations in the most of other online services would not severely harm users (e.g., waste of time, need to refund), wrong investment products would directly lead to monetary damages that cannot be simply returned. Hence, the performances of stock recommender systems should be carefully evaluated accounting for high uncertainties in stock prices.

In this paper, we develop a novel model for stock recommendation, **mean-variance efficient collaborative filtering (MVECF)**, that has all three key requirements. We utilize the regularization technique to ensure that the recommendation is made based on the user's current portfolio while increasing diversification effect by systemically handling uncertainties of stock returns, and the model is restructured to an ordinary weighted matrix factorization (WMF) form [10] to boost the computational efficiency. The proposed model is specifically designed to improve the pareto optimality in a trade-off between risk and return (i.e., the mean-variance efficiency), which is the essence of the modern portfolio theory (MPT). We further show that MVECF can be easily incorporated into state-of-the-art graph-based ranking models by applying MVECF user-item ratings to the sampling process of the ranking systems.

2 RELATED WORKS

but existing method have to trade off between performance, accuracy...

2.1 Collaborative Filtering

CF is the most popular approach in modern recommendation systems because of its efficiency in utilizing the entire user-item preference history [10]. Recently, graph-based ranking models like NCF, NGCF, LightGCN, UltraGCN, and HCCF [8, 23, 7, 13, 26] have become state-of-the-art for implicit feedback data.

This work may look similar to a stream of research to increase the diversity and novelty of recommendations. Diversity focuses on recommending dissimilar items, and novelty seeks items that are dissimilar to the items in the user's past experiences. These involve greedy re-ranking [30, 28, 22] and directly optimizing multi-objectives approaches [19, 18, 11, 24, 25]. However, the notion of recommendation diversity and novelty is quite different from portfolio diversification. Portfolio diversification considers the trade-off between minimizing risk and maximizing return of the entire portfolio.

2.2 Stock Recommendations

Previous studies on stock recommender systems can be categorized into three. First, purely item-based recommendation. These studies try to find stocks that would have high returns in the future by analyzing item similarities between stocks

[27, 21, 29, 4]. This approach has a quite different perspective from conventional recommender systems given that most methods in this category do not utilize user information. In addition, accurately predicting stock returns is almost impossible, because signals are dominated by noises in financial markets.

Second, recommendations based on user-item information. These studies suggest ways to measure similarities between stocks or equity funds, and then apply existing CF methods [15, 1, 2]. However, simply buying stocks that are held by similar investors may make a portfolio more exposed to a certain type of risk (or a sector). Hence, we incorporate the portfolio theory to ensure proper diversification within the CF framework.

Third, recommendations based on user-item information and then diversification. For example, [16] and [20] recommend stocks based on the second approach and determine weights of stocks to reduce portfolio risk. However, adjusting portfolio weights after choosing stocks would have limited effects. On the other hand, [5] and [6] carefully analyze user similarity and recommend a portfolio according to the user's preference. This approach may be able to provide well-diversified portfolios to investors, but this has the same problem with existing mutual funds that investors prefer possessing a few stocks that they are interested in.

3 PRELIMINARIES

3.1 Weighted Matrix Factorization

c_ui is weight term for implicit score (sparse matrix)

Consider m users and n stocks (items). Let the binary variable representing whether the user u holds the stock i is y_{ui}. In matrix form, they can be represented as the user-item interaction matrix $\mathrm{Y} \in \mathbb{R}^{m \times n}$. WMF decomposes Y into the user embedding matrix $\mathrm{P} \in \mathbb{R}^{m \times l}$ and the item embedding matrix $Q \in \mathbb{R}^{n \times l}$ with l number of latent factors as follows.

$$\min \sum_{u,i} c_{ui}(y_{ui} - p_u^T q_i)^2 + \lambda \sum_u \|p_u\|^2 + \lambda \sum_i \|q_i\|^2 \qquad (1)$$

Here, $y_{ui} \in \mathbb{R}$ is an element of Y, and $p_u^T \in \mathbb{R}^{1 \times l}$ and $q_i^T \in \mathbb{R}^{1 \times l}$ are row vectors of P and Q, respectively. $c_{ui} \in \mathbb{R}$ is a hyperparameter indicating the confidence level about observation y_{ui}, which becomes large when $y_{ui} = 1$ and small when $y_{ui} = 0$, and λ is a hyperparameter for L_2 regularization. $\hat{y}_{ui}$, the estimated preference of user u to item i is $p_u^T q_i$. [10] proposed an efficient alternating least squares (ALS) algorithm for solving Equation (1).

3.2 Modern Portfolio Theory (MPT)

Markowitz [14] was the first to mathematically define and analyze the risk and return of financial investments. The return of a risky asset was regarded as a random variable and the expected return was defined as its mean value and the risk was defined as its standard deviation. Then, n risky assets can be described by their return vector $r \in \mathbb{R}^n$ with mean $\mu \in \mathbb{R}^n$ and covariance matrix $\Sigma \in \mathbb{R}^{n \times n}$. A portfolio of n risky assets can be represented as a weight vector $w \in \mathbb{R}^n$, which should sum to one (i.e., $\sum_i w_i = 1$), and it's expected return and risk can be expressed as $\mu^T w$ and $w^T \Sigma w$, respectively.

$$\min_{\{w:\, \sum w = 1,\ w \geq 0\}} \frac{\gamma}{2} w^T \Sigma w - \mu^T w \qquad (2)$$

plus (more error) minus (reduce error)

In [14], a convex quadratic programming problem (Equation (2)) was proposed to find a pareto optimal portfolio between minimizing the risk and maximizing the expected return. It is called the mean-variance (MV) optimization, and the resulting optimal portfolios are called MV efficient portfolios. The term 'efficient' emphasizes the pareto optimality of the solution from the trade-off between risk and return, where γ is the parameter that represents the risk aversion of the investor. [14] is the foundation of the modern portfolio theory (MPT), and it is widely used in practice as well [12].

An evaluation metric for measuring portfolio efficiency is proposed by [17] which is called the Sharpe ratio. It is one of the most widely used performance measures in investment management. It is defined as a ratio between the expected return and the risk (return standard deviation) of the portfolio as $SR(w) = \frac{\mu^T w}{\sqrt{w^T \Sigma w}}$.

4 METHODOLOGY

The common goal of all recommendation systems is to recommend items that are likely to be selected by users in the future. Most recommender systems achieve this by recommending the top k items in the predicted ratings $\hat{y}$. In addition to that, the goal of this work is to generate $\hat{y}$ so that when the users accept the top k recommendations and add those items to their current portfolios y_u, the resulting portfolios will become more MV efficient (measured in the Sharpe ratio). Furthermore, our model should be able to adjust within two kinds of trade-off: 1) between recommendation performance and MV efficiency, 2) between portfolio risk and expected return.

4.1 Mean-Variance Regularization

In this section, we develop the mean-variance efficient collaborative filtering (MVECF) method, which is a novel WMF model with regularization on the MV efficiency. The estimated item ratings (holdings) of user u is $\hat{y}_u = Qp_u$, and we consider this as the user's final portfolio. The expected return and the risk of the portfolio would be written as $\mu^T \hat{y}_u$ and $\hat{y}_u^T \Sigma \hat{y}_u$, respectively. We regularize 'risk – return (i.e., $\frac{\gamma}{2}\hat{y}_u^T \Sigma \hat{y}_u - \mu^T \hat{y}_u$)' of the user's portfolio, which is the objective function of the MV optimization problem in (2), to the loss function of WMF. Then, the proposed model would recommend stocks based on CF while trying to minimize risk and maximize return. The resulting formulation is given in (3). λ_{MV} is a hyperparameter that controls the trade-off between the traditional recommendation performance and the MV efficiency of recommended portfolios.

$$\min_{P,Q} \sum_{u,i} c_{ui}(y_{ui} - \hat{y}_{ui})^2 + \lambda \sum_u \|p_u\|^2 + \lambda \sum_i \|q_i\|^2 + \lambda_{MV} \sum_u \left(\frac{\gamma}{2}\hat{y}_u^T \Sigma \hat{y}_u - \mu^T \hat{y}_u\right) \quad (3)$$

If we rewrite the last MV regularization term of (3) in an elementwise expression with μ_i (elements of μ), σ_i^2 and σ_{ij} (diagonal and off-diagonal elements of Σ), Equation (3) can be rewritten as Equation (4).

plus sign is plus error
minus sign is minus error

$$\min_{P,Q} \sum_{u,i} \left[c_{ui}(y_{ui} - \hat{y}_{ui})^2 + \lambda_{MV}\frac{\gamma}{2}\left(\hat{y}_{ui}^2\sigma_i^2 + \hat{y}_{ui}\sum_{j:j\neq i}\hat{y}_{uj}\sigma_{ij}\right) - \lambda_{MV}\hat{y}_{ui}\mu_i\right] + \lambda \sum_u \|p_u\|^2 + \lambda \sum_i \|q_i\|^2 \quad (4)$$

We can see from Equation (4) that the MV regularization would lower the rating of items with high variance (from $\hat{y}_{ui}^2\sigma_i^2$), lower the rating of items with high covariance with user holdings (from $\hat{y}_{ui}\sum_{j:j\neq i}\hat{y}_{uj}\sigma_{ij}$), and raise the rating of items with high expected returns (from $-\hat{y}_{ui}\mu_i$). Hence, it exactly delivers the desired effects.

4.2 Restructuring MVECF into Ordinary WMF Form

Here, we further increase the computational efficiency of MVECF by restructuring it into an ordinary WMF form (as in Equation (1)) so that we can train it using the ALS algorithm developed by [10]. ALS is known to converge much faster than SGD, and also SGD is highly sensitive to the choice of learning rate.

The trick is quite simple. In the MV regularization term in Equation (3), we change $\hat{y}_{ui}\sum_{j:j\neq i}\hat{y}_{uj}\sigma_{ij}$ into $\hat{y}_{ui}\sum_{j:j\neq i} y_{uj}\sigma_{ij}/|y_u|$. Then, (3) can be rewritten as

$$\min \sum_{u,i}\left[\left(c_{ui} + \frac{\gamma}{2}\lambda_{MV}\sigma_i^2\right)\hat{y}_{ui}^2 - \left(2c_{ui}y_{ui} - \frac{\gamma}{2}\lambda_{MV}\sum_{j:j\neq i}\frac{y_{uj}\sigma_{ij}}{|y_u|} + \lambda_{MV}\mu_i\right)\hat{y}_{ui} + c_{ui}y_{ui}^2\right] + \lambda\left(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2\right) \quad (5)$$

This trick has a nice theoretical property as well. Note that regularizing with $\hat{y}_{ui} \sum_{j:j\neq i} \hat{y}_{uj}\sigma_{ij}$ would reduce the rating of items that have large covariances with 'predicted' user holdings. On the other hand, regularizing with the modified term $\hat{y}_{ui} \sum_{j:j\neq i} y_{uj}\sigma_{ij}$ would reduce the rating of items that have large covariances with 'current' user holdings. Therefore, no matter which and how many items the user finally accepts from the recommended list, they would all possess diversification potential.

We can see that both the first and the second term in (5) are quadratic functions of $\hat{y}_{ui}$, The idea of the reconstruction is to combine these two terms into one. If we denote $c_{ui} + \frac{\gamma}{2}\lambda_{MV}\sigma_i^2$ as $\tilde{c}_{ui}$ and define $\tilde{y}_{ui}$ so that $2\tilde{c}_{ui}\tilde{y}_{ui}$ can be $2c_{ui}y_{ui} - \frac{\gamma}{2}\lambda_{MV}\sum_{j:j\neq i}\frac{y_{uj}}{|y_u|}\sigma_{ij} + \lambda_{MV}\mu_i$ then Equation (5) becomes $\min \sum_{u,i} \tilde{c}_{ui}\hat{y}_{ui}^2 - 2\tilde{c}_{ui}\tilde{y}_{ui}\hat{y}_{ui} + c_{ui}y_{ui}^2 + \lambda(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2)$ and in perfect square form $\min \sum_{u,i} \tilde{c}_{ui}(\tilde{y}_{ui} - \hat{y}_{ui})^2 - \tilde{c}_{ui}\tilde{y}_{ui}^2 + c_{ui}y_{ui}^2 + \lambda(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2)$. Since $-\tilde{c}_{ui}\tilde{y}_{ui}^2 + c_{ui}y_{ui}^2$ is independent with P and Q, (5) is equivalent to (6).

$$\min \sum_{u,i} \tilde{c}_{ui}(\tilde{y}_{ui} - \hat{y}_{ui})^2 + \lambda(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2) \qquad (6)$$

Note that (13) has exactly the same form of ordinary WMF with modified ratings $\tilde{y}_{ui}$ and their weighting coefficients $\tilde{c}_{ui}$. To interpret $\tilde{y}_{ui}$ and $\tilde{c}_{ui}$, we define two MV related parameters $c_{ui}^{MV} = \frac{\gamma}{2}\lambda_{MV}\sigma_i^2$ and $y_{ui}^{MV} = \left(\frac{\mu_i}{\gamma} - \frac{1}{2}\sum_{j:j\neq i}\frac{y_{uj}}{|y_u|}\sigma_{ij}\right)/\sigma_i^2$. Then the definition $\tilde{y}_{ui}$ can be rewritten as $\tilde{y}_{ui} = c_{ui}y_{ui}/\tilde{c}_{ui} + c_{ui}^{MV}y_{ui}^{MV}/\tilde{c}_{ui}$ where $\tilde{c}_{ui} = c_{ui} + c_{ui}^{MV}$.

$$\tilde{y}_{ui} = \frac{2c_{ui}y_{ui} + \lambda_{MV}\left(\mu_i - \frac{\gamma}{2}\sum_{j:j\neq i}\frac{y_{uj}}{|y_u|}\sigma_{ij}\right)}{2\tilde{c}_{ui}} = \frac{c_{ui}y_{ui} + \frac{\gamma}{2}\lambda_{MV}\sigma_i^2\frac{\frac{\mu_i}{\gamma} - \frac{1}{2}\sum_{j:j\neq i}\frac{y_{uj}}{|y_u|}\sigma_{ij}}{\sigma_i^2}}{\tilde{c}_{ui}} = \frac{c_{ui}y_{ui} + c_{ui}^{MV}y_{ui}^{MV}}{\tilde{c}_{ui}} \qquad (7)$$

The MV rating y_{ui}^{MV} would have a large value when the mean return of item i (μ_i) is high and the risk of item i (σ_i^2) is low. And the modified target rating $\tilde{y}_{ui}$ is a weighted sum of the user's current holdings y_{ui} and the MV rating y_{ui}^{MV}. Therefore, the recommendation $\hat{y}_{ui}$ would directly reflect the MV rating to favor items with better risk-return profiles. Also, the weighting term $\tilde{c}_{ui}$ becomes large when σ_i^2 is large, and thus, the model focuses more on matching ratings of risky items compared to safe items. By simply changing the true rating y_{ui} into a weighted sum of y_{ui} and y_{ui}^{MV}, we can train a WMF model to make user preferred recommendations, while making the resulting portfolio more efficient in terms of risk-return trade-off. The model can be easily trained by ALS.

5 EXPERIMENTS

5.1 Data & Models

Two datasets are used for experiments. One is Survivorship-Bias-Free US Mutual Fund data from Center for Research in Security Prices (CRSP), and the other is Stock Ownership data from Thomson Reuters. Each dataset contains holdings snapshots of users (mutual funds or institutional investors) for every month from 2001 to 2020. Both datasets were retrieved from Wharton Research Data Services (WRDS) database.

We split each dataset into yearly sub-datasets. The user-item interaction data of year T is the holdings snapshot reported in December of year T. For MVECF model, stock mean return and covariance matrix are estimated using the returns data in the past 5 years (years T-4 to T). For ex post performance evaluation, we use the next 5 years returns data (years T+1 to T+5).

We consider two versions of MVECF. The first one ($\text{MVECF}_{\text{reg}}$) is the regularization model and the second one ($\text{MVECF}_{\text{WMF}}$) is the ordinary WMF form version. To demonstrate the performance of MVECF, we use the state-of-the-art conventional recommender systems WMF, BPR, LightGCN, UltraGCN, and HCCF as the baseline models.

In addition, we consider three existing models for stock recommendations. First, novelty enhancing BPR model ($\mathrm{BPR_{nov}}$) of [25]. For this model, we define the dissimilarity (distance) between two items i and j as $\sqrt{1-\rho_{ij}}$, where ρ_{ij} is the return correlation between i and j. Second, the 2-step method [20] introduced in Section 2.1. It filters top-k items using a base recommendation model, and then make the final recommendation by re-ranking the scores of top-k items using the MPT method. We consider two versions of this model with two different base recommendation methods: WMF ($\mathrm{2Step_{wmf}}$) and UltraGCN ($\mathrm{2Step_{ugcn}}$).

The data, code, and more detailed experimental settings are given in https://github.com/author-mvecf/MVECF.git

5.2 Performance Evaluation

The performance of stock recommender systems should be evaluated in two aspects: conventional recommendation performance (precision, recall) and MV efficiency. To be more specific, top 20 items of each user's test data are recommended and mean average precision at 20 (MAP@20) and recall at 20 (Recall@20) are used for conventional recommendation performance evaluation.

For MV efficiency evaluation, we recommend top 20 items of all non-holding items to users. This is because recommending items only in pre-chosen test set is inappropriate for investment setting. We use Sharpe ratios of the initial portfolio (before recommendation) and the recommended portfolio (after adding 20 recommended items). Both portfolios are constructed as equally weighted portfolios because MVECF models recommend just items (stocks), not their weights.

Next, we use two measures that are based on the calculated Sharpe ratios. First, $\Delta\mathrm{SR} = \mathrm{SR} - \mathrm{SR_{init}}$, the improvement in the Sharpe ratio, where SR and $\mathrm{SR_{init}}$ are Sharpe ratios of recommended and initial portfolios, respectively. Second, $\mathrm{P(SR > SR_{init})}$, the percentage of users whose Sharpe ratio has increased after recommendation. While $\Delta\mathrm{SR}$ would show the amount of improvement, $\mathrm{P(SR > SR_{init})}$ would show how many users get the improvement.

As noted in Introduction, ex-post evaluation is particularly important in investment management. Hence, the same metrics, $\Delta\mathrm{SR}$ and $\mathrm{P(SR > SR_{init})}$, are calculated with 5 years performance (realized return and variance) of the recommended portfolio. The reason for using Sharpe ratio as our MV performance measure is that it is undoubtedly the most widely used investment performance measure in both practice and academia.

We also analyze the improvement in mean return $\Delta\mu$ and risk $\Delta\sigma$ of recommended portfolio to see the risk-return tradeoff of MVECF. The hyperparameters are tuned within the validation set (10% of total data). Experiments are repeated with different values of balancing hyperparameter λ_{MV} in 0.1, 1, 10 and risk-aversion level γ in 1, 3, 5.

Table 1: Average of evaluation metrics across all datasets for recommended portfolio by MVECF models

	Parameters	Performance metrics					
	λ_{MV}	$\Delta\mu$	$\Delta\sigma$	ΔSR	$\mathrm{P(SR > SR_{init})}$	MAP@20	Recall@20
$\mathrm{MVECF_{reg}}$	0.1	-0.0008	-0.0092	0.0163	0.7463	0.2365	0.6375
	1	-0.0016	-0.0123	0.0186	0.7794	0.2315	0.6254
	10	-0.0029	-0.0178	0.0258	0.8745	0.2250	0.6091
$\mathrm{MVECF_{WMF}}$	0.1	-0.0003	-0.0057	0.0115	0.7086	0.2526	0.7008
	1	-0.0003	-0.0092	0.0183	0.8267	0.2530	0.6956
	10	0.0021	-0.0206	0.0590	0.9893	0.2345	0.6063
	γ	$\Delta\mu$	$\Delta\sigma$	ΔSR	$\mathrm{P(SR > SR_{init})}$	MAP@20	Recall@20
$\mathrm{MVECF_{reg}}$	1	-0.0022	-0.0159	0.0247	0.8541	0.2289	0.6193
	3	-0.0029	-0.0178	0.0258	0.8745	0.2250	0.6091
	5	-0.0032	-0.0184	0.0261	0.8784	0.2231	0.6064
$\mathrm{MVECF_{WMF}}$	1	0.0144	-0.0077	0.0762	0.9875	0.2338	0.6028
	3	0.0021	-0.0206	0.0590	0.9893	0.2345	0.6063
	5	0.0000	-0.0244	0.0599	0.9903	0.2321	0.6027

5.3 Experiment Results

Before we demonstrate the relative performances of MVECF models to baseline models, let us check whether MVECF performs as we expected. Table 1 shows the averages of various evaluation metrics of MVECF models with different values of λ_{MV} and γ. The upper half shows the results of various values of λ_{MV} when γ is fixed to 3. Both models show better ΔSR and $P(SR > SR_{init})$ when λ_{MV} is large, and better MAP@20 and Recall@20 when λ_{MV} is small. We can easily see that the tradeoff between MV efficiency and conventional recommendation performance is well controlled by λ_{MV}.

The bottom half is the results of varying γ when λ_{MV} is fixed to 10. As we assume a more risk-averse user by increasing γ, it clearly shows MVECF models focus more on reducing risk than increasing return. Hence, γ controls the risk-return tradeoff as desired.

Now, we compare the MV efficiency and conventional recommendation performance of MVECF with baseline models. Figure 1 shows the results of conventional recommender systems, existing stock recommendation models, and two versions of MVECF with $\lambda_{MV} = 10$ and $\gamma = 3$ for a total of 20 sub-datasets. Each marker in the graph represents the average performance of all users in one yearly sub-dataset. Figure 2 represents the ex-post Sharpe ratio performance.

From the two figures, it is evident that both $\mathrm{MVECF_{reg}}$ and $\mathrm{MVECF_{WMF}}$ outperform conventional recommender systems in terms of MV efficiency (i.e., ΔSR and $P(SR > SR_{init})$). $\mathrm{MVECF_{WMF}}$, in particular, exhibits dominating performance with $P(SR > SR_{init})$ values near 100% for all datasets, implying that MVECF is beneficial for almost all investors in improving MV efficiency. Conventional recommender systems, however, exhibit ΔSR and $P(SR > SR_{init})$ values not much different from each other, as they do not address portfolio diversifications.

Regarding recommendation performance, MVECF models naturally show inferior performance compared to conventional recommendation systems. However, the decline is not substantial. The average decrease in recommendation performance across 20 sub-datasets compared to the best performing UltraGCN is less than 5% in MAP@20 and less than 10% in Recall@20 for both MVECF models. This is relatively small compared to the improvement in MV efficiency.

As for existing stock recommendation models, Figure 1 indicates that two-step methods ($\mathrm{2Step_{wmf}}$ and $\mathrm{2Step_{ugcn}}$) achieve similar MV efficiency levels to MVECF models, whereas $\mathrm{BPR_{nov}}$ does not. However, a clear difference can be seen from the ex-post performance in Figure 2. While MVECF models outperform all other models in most cases, two-step models even show negative ΔSR and $P(SR > SR_{init})$ below 50%. This indicates that the two-step methods would lead to really poor investment performances.

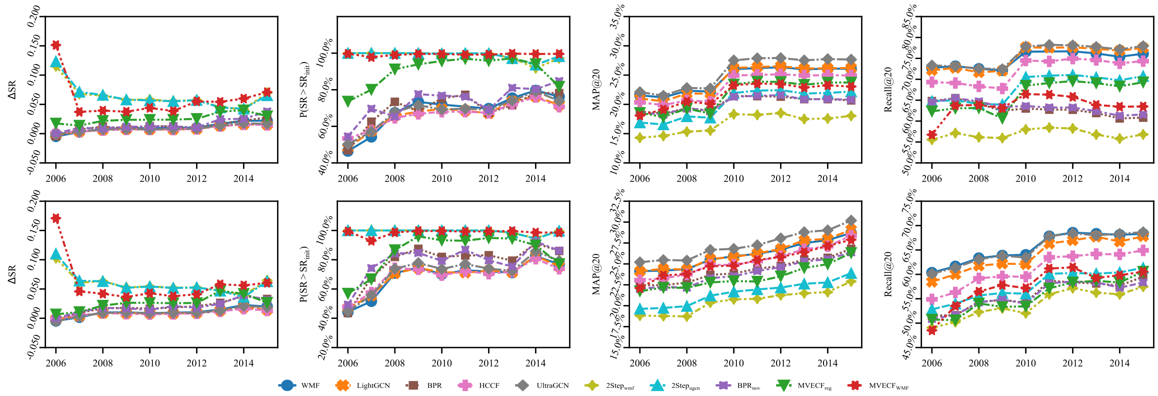

Figure 1: Performance comparison between MVECF and baseline models. The figures in upper row are ΔSR, $P(SR > SR_{init})$, MAP@20 and Recall@20 with CRSP dataset, and the figures in lower row are those with Thomson Reuters dataset.

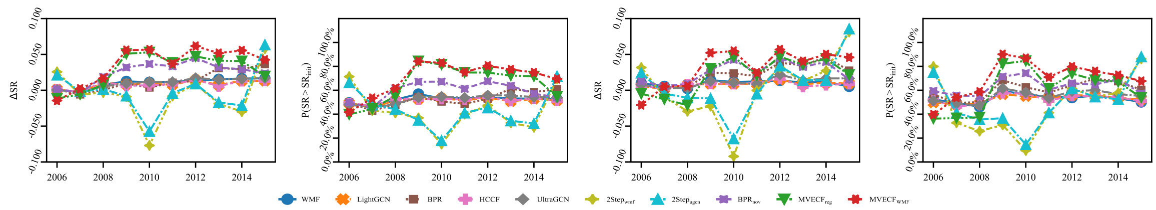

Figure 2: Ex-post SR performance comparison between MVECF and baseline models. The two left figures are ΔSR and $P(SR > SR_{init})$ with CRSP dataset, and the two right figures are those with Thomson Reuters dataset.

5.4 Incorporating MVECF into Ranking Models

Most studies on recommender systems are based on ranking models with implicit feedbacks. Note that MVECF is developed based on WMF, which is a rating-prediction model. However, the key point of MVECF is to modify the true rating y_{ui} to $\tilde{y}_{ui}$, which incorporates the MV efficiency of items. Using this, we show that MVECF can be easily incorporated into recent state-of-the-art ranking models.

In traditional ranking models, items associated with real user-item interactions are considered as positive samples, while negative samples are chosen from the items without such interactions. We propose an MV efficient sampling scheme that identifies items with $\tilde{y}_{ui} > \tau$ as positive samples and items with $\tilde{y}_{ui} < \tau$ as negative samples, where τ is a predefined threshold level.

As discussed in Section 4.2, $\tilde{y}_{ui}$ is a weighted sum of the true rating y_{ui} and the MV rating y_{ui}^{MV}. The MV rating y_{ui}^{MV} is bad (good) when the item is highly (less) correlated with the user's portfolio and/or has low (high) expected return. Hence, even though a user actually holds item i, if y_{ui}^{MV} is really bad, it will be classified as a negative sample in MV efficient sampling. Similarly, items with really good y_{ui}^{MV} will be regarded as positive samples, regardless of actual interactions y_{ui}. Given the sparsity of true positive items, we set the threshold τ so that 1% of the original negative samples can be converted to positive samples in MV efficient sampling.

Table 2 presents the MV efficiency and recommendation performance of LightGCN and UltraGCN, which exhibited the best recommendation performance in Section 5.4, as well as LightGCN and UltraGCN with the MV efficient sampling.

The results clearly show that the state-of-the-art graph based ranking models can be easily extended to improve MV efficiency ($\Delta\mathrm{SR}$ and $\mathrm{P}(\mathrm{SR} > \mathrm{SR}_{\mathrm{init}})$) with simple modification.

Table 2: Average of evaluation metrics across all datasets for LightGCN and UltraGCN, and the models with MV efficient sampling $\mathrm{LightGCN}_{\mathrm{MVECF}}$ and $\mathrm{UltraGCN}_{\mathrm{MVECF}}$.

	$\Delta\mathrm{SR}$	$\mathrm{P}(\mathrm{SR} > \mathrm{SR}_{\mathrm{init}})$	MAP@20	Recall@20
LightGCN	0.0086	0.6791	0.2526	0.6966
$\mathrm{LightGCN}_{\mathrm{MVECF}}$	0.1139	0.9928	0.1868	0.6282
UltraGCN	0.0094	0.6979	0.2641	0.7058
$\mathrm{UltraGCN}_{\mathrm{MVECF}}$	0.1208	0.9972	0.1743	0.6219

6 CONCLUSION

In this paper, we proposed the mean-variance efficient collaborative filtering (MVECF) for stock recommendation that can systemically handle the risk-return profile of recommended portfolios while recommending stocks with the consideration of user preferences. Starting from a simple regularization, we were able to derive MVECF as an ordinary WMF form. The performances of portfolios recommended by MVECF outperformed other recommender systems in both in-sample and out-of-sample settings with only minimal reductions in the recommendation performance. Furthermore, we demonstrated that the modified user-item rating of MVECF can be integrated into the positive and negative sampling of ranking models, allowing state-of-the-art graph-based models to offer MV efficient recommendations for users.

The importance of this research lies in addressing the unique challenges of stock recommendation within the rapidly evolving fintech industry. As personalized stock recommendations become increasingly relevant for attracting and retaining customers, our approach can significantly enhance customer engagement and satisfaction, providing investment companies and online brokers with a competitive edge in the fintech landscape. This is the first study to identify the key requirements for stock recommender systems and develop a proper CF model for stock recommendations to fully utilize the user-item interactions. We believe that our study can encourage many researchers to develop more advanced stock recommender systems that can properly handle the risk-return characteristics of stocks as well as the preference of users.

REFERENCES

[1] Chang, J.; and Tu, W. 2018. A stock-movement aware approach for discovering investors' personalized preferences in stock markets. In 2018 IEEE 30th International Conference on Tools with Artificial Intelligence, 275-280.

[2] De Rossi, G.; Kolodziej, J.; and Brar, G. 2020. A recommender system for active stock selection. Computational Management Science 17(4): 517-547.

[3] Dimmock, S. G.; Kouwenberg, R.; Mitchell, O. S.; and Peijnenburg, K. 2021. Household portfolio underdiversification and probability weighting: Evidence from the field. The Review of Financial Studies 34(9): 4524-4563.

[4] Gao, J.; Ying, X.; Xu, C.; Wang, J.; Zhang, S.; and Li, Z. 2021. Graph-based stock recommendation by time-aware relational attention network. ACM Transactions on Knowledge Discovery from Data, 16(1): 1-21.

[5] García-Crespo, Á.; López-Cuadrado, J. L.; González-Carrasco, I.; Colomo-Palacios, R.; and Ruiz-Mezcua, B. 2012. SINVLIO: Using semantics and fuzzy logic to provide individual investment portfolio recommendations. Knowledge-Based Systems 27: 103-118.

[6] Gonzalez-Carrasco, I.; Colomo-Palacios, R.; Lopez-Cuadrado, J. L.; Garcı, Á.; and Ruiz-Mezcua, B. 2012. PB-ADVISOR: A private banking multi-investment portfolio advisor. Information Sciences 206: 63-82.

[7] He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, 639-648.

[8] He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T. S. 2017. Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web, 173-182.

[9] Hendershott, T.; Zhang, X.; Zhao, J. L.; and Zheng, Z. 2021. FinTech as a game changer: Overview of research frontiers. Information Systems Research 32(1): 1-17.

[10] Hu, Y.; Koren, Y.; and Volinsky, C. 2008. Collaborative filtering for implicit feedback datasets. In 2008 Eighth IEEE international conference on data mining, 263-272.

[11] Hurley, N. J. 2013. Personalised ranking with diversity. In Proceedings of the 7th ACM Conference on Recommender Systems, 379-382.

[12] Kim, J.H., Lee, Y., Kim, W.C., & Fabozzi, F.J. 2021. Mean-variance optimization for asset allocation, Journal of Portfolio Management, 47(5), 24-40

[13] Mao, K., Zhu, J., Xiao, X., Lu, B., Wang, Z., and He, X. 2021. UltraGCN: ultra simplification of graph convolutional networks for recommendation. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, 1253-1262.

[14] Markowitz, H. 1952. Portfolio Selection. Journal of Finance 7(1): 77–91.

[15] Matsatsinis, N. F.; and Manarolis, E. A. 2009. New hybrid recommender approaches: an application to equity funds selection. In International Conference on Algorithmic Decision Theory, 156-167.

[16] Musto, C.; Semeraro, G.; Lops, P.; De Gemmis, M.; and Lekkas, G. 2015. Personalized finance advisory through case-based recommender systems and diversification strategies. Decision Support Systems 77: 100-111.

[17] Sharpe, W. F. 1966. Mutual Fund Performance. Journal of Business 39: 119–138.

[18] Shi, Y.; Zhao, X.; Wang, J.; Larson, M.; and 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, 175-184.

[19] Su, R.; Yin, L. A.; Chen, K.; and Yu, Y. 2013. Set-oriented personalized ranking for diversified top-n recommendation. In Proceedings of the 7th ACM Conference on Recommender Systems, 415-418.

[20] Swezey, R. M.; and Charron, B. 2018. Large-scale recommendation for portfolio optimization. In Proceedings of the 12th ACM Conference on Recommender Systems, 382-386.

[21] Tu, W.; Cheung, D. W.; Mamoulis, N.; Yang, M.; and Lu, Z. 2016. Investment recommendation using investor opinions in social media. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 881-884.

[22] Vargas, S.; and Castells, P. 2014. Improving sales diversity by recommending users to items. In Proceedings of the 8th ACM Conference on Recommender systems, 145-152.

[23] Wang, X.; He, X.; Wang, M.; Feng, F.; and Chua, T. S. 2019. Neural graph collaborative filtering. In Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval, 165-174.

[24] Wasilewski, J.; and Hurley, N. 2016. Incorporating diversity in a learning to rank recommender system. In The twenty-ninth international flairs conference, 572-576

[25] Wasilewski, J.; and Hurley, N. 2019. Bayesian personalized ranking for novelty enhancement. In Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, 144-148.

[26] Xia, L., Huang, C., Xu, Y., Zhao, J., Yin, D., and Huang, J. 2022. Hypergraph contrastive collaborative filtering. In Proceedings of the 45th International ACM SIGIR conference on research and development in information retrieval, 70-79.

[27] Yujun, Y.; Jianping, L.; and Yimei, Y. 2016. An efficient stock recommendation model based on big order net inflow. Mathematical Problems in Engineering, 2016.

[28] Zhang, M.; and Hurley, N. 2008. Avoiding monotony: improving the diversity of recommendation lists. In Proceedings of the 2008 ACM conference on Recommender systems, 123-130.

[29] Zheng, Z.; Gao, Y.; Yin, L.; and Rabarison, M. K. 2020. Modeling and analysis of a stock-based collaborative filtering algorithm for the Chinese stock market. Expert Systems with Applications, 162.

[30] Ziegler, C. N.; McNee, S. M.; Konstan, J. A.; and Lausen, G. 2005. Improving recommendation lists through topic diversification. In Proceedings of the 14th international conference on World Wide Web, 22-32.

A APPENDICES

A.1 Data details

We split each dataset into yearly sub-datasets as shown in Figure 3. In our experiment, we recommend items to users at the end of each year, so we use holdings snapshot reported in December of year T as the holdings of year T. We define preferred items as the current holdings to avoid recommending the items that were recently sold by the user. Although we have monthly holdings data, portfolios do not change much every month, so we create yearly sub-datasets for more robust experiments.

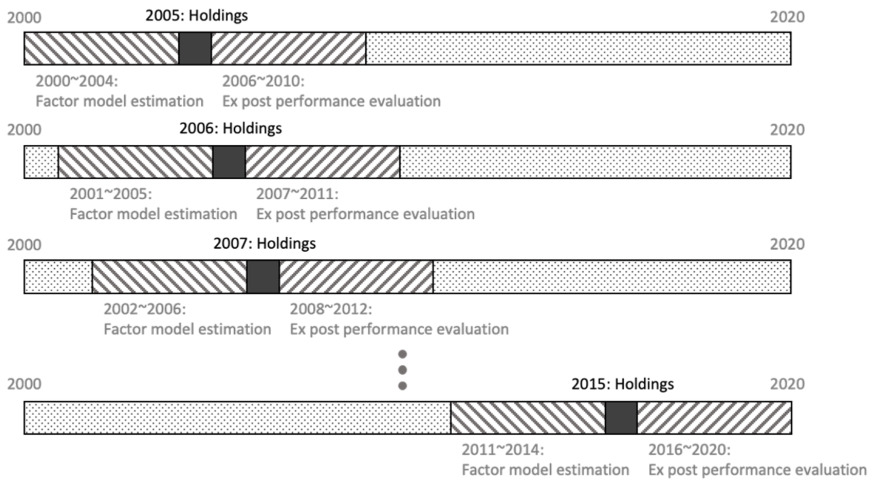

Figure 3. Yearly sub-dataset construction

For MVECF model, we need stock mean returns and covariance matrix should be estimated before the recommendation. Hence, we estimate them using returns data in the past 5 years (years T-4 to T). For ex post performance evaluation, we use the next 5 years returns data (years T+1 to T+5). For example, when we are running an experiment for the year 2015 dataset, the user-item interaction data is constructed from the holdings in Dec 2015, the stock mean return and covariance matrix are estimated with the stock returns data from 2011 to 2015, and the stock returns data from 2016 to 2020 is used for ex-post performance evaluation.

As a result, we get 10 yearly sub-datasets from 2006 to 2015. The number of users and items, and the average number of holdings in resulting 20 yearly sub-datasets (10 for CRSP, 10 for Thomson Reuters) are summarized in Table 1. The user-item interaction data in each dataset is divided into train, test, and validation data at a ratio of 8:1:1. The sample data is CRSP data for year 2012.

A.2 Hyperparameter tuning

Since MVECF is based on WMF, we need to consider the same set of hyperparameters of ordinary WMF. The hyperparameters of WMF are the latent dimension l, the regularization parameter λ, and the confidence level c_{ui} of each observation $y_{ui} = 1$.

We performed a grid search within the following ranges $l \in \{10, 30, 50\}$, $c_{ui} \in \{5, 10, 20, 40\}$, and λ ∈{0.0001, 0.001, 0.01}. The evaluation criteria for choosing the hyperparameter was MAP@20. The final hyperparameter values chosen for our experiments are $l = 30$, $c_{ui} = 10$ and λ =0.001. For SGD update of MVECF, we chose the learning rate α to 0.001, because the convergence was too slow for smaller learning rates and MVECF with large λ_{MV} did not converge in some of our datasets for larger learning rates.

For BPR_{nov} of [47], it is a modification of BPR model [28]. BPR uses matrix factorization to predict ratings by minimizing $\sum_{u,i,j\in\mathcal{D}_s} -\log\sigma(\hat{y}_{ui} - \hat{y}_{uj})$, where $\sigma(\cdot)$ is the logistic sigmoid function, and $\mathcal{D}_s$ is a set of triples (u, i, j) with $y_{ui} = 1$ and $y_{uj} = 0$. [47] defined a new set of triples $\mathcal{D}_s^{dist} = \{(u,i,j): (u,i,j) \in \mathcal{D}_s \text{ and } \text{dist}(i,j) < \tau\}$, where $\text{dist}(i,j)$ is the distance between items i and j, and τ is a distance threshold. They trained BPR_{nov} model with (u, i, j) sampled from $\mathcal{D}_s^{dist}$ with probability β and sampled from $\mathcal{D}_s$ with probability $1 - \beta$ to recommend items that are distinct from the items in the user's preference history. Therefore, the hyperparameters of BPR_{nov} are τ, β, learning rate α, regularization parameter λ, and latent dimension l. We set latent dimension $l = 30$ and $\beta = 0.8$. We performed a grid search within range $\lambda \in \{0,\ 0.00001,\ 0.0001,\ 0.001\}$ and $\alpha \in \{0.0001, 0.001, 0.01\}$ for BPR_{nov}. The chosen values are $\alpha = 0.001$ and $\lambda = 0.00001$. Finally, τ was set to 0.9 because it showed the best balance between novelty and recommendation performance among 0.8, 0.9, and 1.0.

For graph based ranking models (LightGCN, UltraGCN, and HCCF) and those models with new sampling methods, we used the same hyperparameters reported in each paper and set the learning rate to 0.001, and the embedding size to 32.

A.3 Train and validation loss

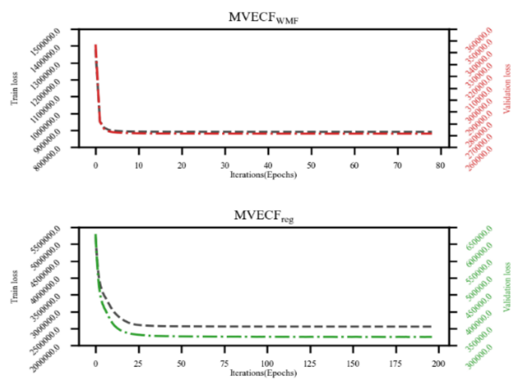

Figure 4. Train and validation losses of $\mathbf{MVECF_{WMF}}$ (top) and $\mathbf{MVECF_{reg}}$ (bottom).

Figure 4 shows the train loss and the validation loss of $\text{MVECF}_{\text{WMF}}$ and $\text{MVECF}_{\text{reg}}$ in one of the yearly sub-datasets (year 2015) of CRSP data. We can easily see that both models converge quite smoothly. In particular, we can see that $\text{MVECF}_{\text{WMF}}$ shows faster convergence compared to $\text{MVECF}_{\text{reg}}$. It implies that restructuring MVECF into an ordinary WMF form makes enhancement in the computational efficiency. Similar results are found in all other datasets.