Budget-Constrained Item Cold-Start Handling in Collaborative Filtering Recommenders via Optimal Design

Oren Anava*
Technion, Haifa, Israel
oanava@tx.technion.ac.il

Shahar Golan
Yahoo Labs, Haifa, Israel
shaharg@yahoo-inc.com

Nadav Golbandi
Yahoo Labs, Haifa, Israel
nadavg@yahoo-inc.com

Zohar Karnin
Yahoo Labs, Haifa, Israel
zkarnin@yahoo-inc.com

Ronny Lempel*
Outbrain Inc., Netanya, Israel
rlempel@outbrain.com

Oleg Rokhlenko
Yahoo Labs, Haifa, Israel
olegro@yahoo-inc.com

Oren Somekh
Yahoo Labs, Haifa, Israel
orens@yahoo-inc.com

ABSTRACT

It is well known that collaborative filtering (CF) based recommender systems provide better modeling of users and items associated with considerable rating history. The lack of historical ratings results in the user and the item cold-start problems. The latter is the main focus of this work. Most of the current literature addresses this problem by integrating content-based recommendation techniques to model the new item. However, in many cases such content is not available, and the question arises is whether this problem can be mitigated using CF techniques only. We formalize this problem as an optimization problem: given a new item, a pool of available users, and a budget constraint, select which users to assign with the task of rating the new item in order to minimize the prediction error of our model. We show that the objective function is monotone-supermodular, and propose efficient optimal design based algorithms that attain an approximation to its optimum. Our findings are verified by an empirical study using the Netflix dataset, where the proposed algorithms outperform several baselines for the problem at hand.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*

General Terms

Algorithms, Experimentation

Keywords

collaborative filtering, item cold-start, optimal design

*This work was done while the authors were at Yahoo Labs.

WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741109.

1. INTRODUCTION

Recommendation technologies are increasingly being used to route relevant and enjoyable information to users. Whether they try to navigate through overwhelming Web content, choose a restaurant, or simply find a book to read, users find themselves being guided by modern online services that use recommendation systems. Usually, such services are based on users' feedback and stated preferences, and use editors, content analysis or wisdom of the crowds to tag their database items.

One of the most common and effective recommendation techniques is Collaborative filtering (CF). This technique relies only on past user behavior (e.g., previous transactions or feedback), and does not require the creation of explicit profiles. Moreover, it requires no domain knowledge or content analysis, and excels at exploiting popularity trends, which drive much of the observed user interaction. One of the fundamental problems arising when employing CF techniques is the *cold-start* problem. Roughly speaking, the cold-start problem is caused by the system's incapability of dealing with new items or new users due to the lack of relevant transaction history.

This work focuses on the item cold-start problem, *without* assuming any availability of context nor content-based information. In particular, we consider the following setting: a publisher has a set of users, and is aware of those users' historical ratings of items (movies, books, etc.). Now the publisher is interested in evaluating a new item, on which she has no prior knowledge. At her disposal is a pool of available users, from which she can assign B users to the task of rating the new item. After receiving their ratings, the publisher estimates, to the best extent possible, how the remaining users will rate the new item.

In order to mitigate the reviewing period, the B users are all selected at once - the publisher does not have the luxury to receive some ratings, and then to adaptively select additional users based on that signal. An important aspect of this problem setting is that the satisfaction of the assigned users does not factor into the equation - there is no cost associated with selecting a user who will dislike the item. The sole purpose is to select users who will provide best characterization of the new item.

The rest of this paper is organized as follows: the remaining of this section presents informal statement of our results, and surveys the related work. Section 2 introduces notations and concretely defines the problem. Section 3 then presents our optimal design based algorithms and analysis. Experimental results are reported in Section 4. We conclude and discuss future work in Section 5.

1.1 Informal Statement of Results

We adopt a common assumption in *Matrix Factorization* (MF)-based CF [16]: ratings of items by users can be well approximated by the product of two low-dimensional matrices, one representing low-dimensional latent vectors of users, and the other holding latent representations of items. The problem at hand then translates to the following mathematical abstraction: given latent vectors representing each of the users, and a new item whose latent vector is unknown, choose which B ratings to reveal so that the item's latent vector could be estimated most accurately.

The main contributions of this work are as follows: (1) We cast this budget-constrained user selection problem as an optimization problem, in which the objective function stands for the expected prediction error of our model. (2) We devise two novel optimal design based algorithms for the problem, each attains approximation to the optimum under certain assumptions. (3) We validate the theoretical results with an empirical study, by simulating the setting of the problem using the Netflix dataset and comparing the effectiveness of our algorithms to that of several baselines.

1.2 Related Work

The two major fields of CF are *neighborhood methods* and *Latent Factor Models* (LFM). LFM characterizes both items and users as vectors in a space automatically inferred from observed data patterns. The latent space representation strives to capture the semantics of items and users, which drive most of their observed interactions. One of the most successful realizations of LFM, which combines good scalability with predictive accuracy, is based on low-rank MF (e.g., see [16]). In particular, low-rank MF provides a substantial expressive power that allows modeling specific data characteristics such as temporal effects [15], item taxonomy [6], and attributes [1].

One of the inherent limitations of MF is the need for historical user-item interactions. When such history is limited, MF-based recommenders cannot reliably model new entities, leading to the item and user cold-start problems. Despite the fact that users and items are usually represented similarly in the latent space, these two problems are essentially different due to the ability to interview new users when joining the service in order to bootstrap their modeling. Moreover, each of these problems has a different line of previous works, as surveyed below.

User cold-start problem. Modeling the preferences of new users can be done most effectively by asking them to rate several carefully selected items of a seed set during a short interview [13, 21, 22, 8]. Item seed sets were constructed according to various criteria such as *popularity* (items should be known to the users), *contention* (items should be indicative of users' tendencies), and *coverage* (items should possess predictive power on other items).

The importance of adaptive interviewing process using adaptive seed sets was already recognized in [21]. Adaptive interviewing is commonly implemented by decision trees algorithms [22, 18, 9]. A method for solving the problem of initial interview construction within the context of learning user and item profiles is presented in [26].

Item cold-start problem. To mitigate the item-cold problem of CF, a common practice involves utilizing external content on top of users' feedback. The basic approach is to leverage items' attributes and combine them with the CF model in a way that allows treating new items [1, 10, 11, 19]. In [1] the authors proposed a regression-based latent factor model for cold-start recommendation. In their model a dyadic response matrix Y is estimated by a latent factor model $Y \approx U^\top V$, where the latent factor matrices, U and V, are estimated by regression $U \approx FX$ and $V \approx MZ$. The matrices X and Z are the user attribute and item feature matrices, and F and M are weight matrices learnt by regression. In a later work [19], the authors improve the performance achieved in [1] by solving a convex optimization problem that estimates the weight matrices, instead of computing the low-rank matrix decomposition of [1]. Another approach to tackle the item cold-start issue by combining content information with CF, based on *Boltzmann machines*, is studied in [10, 11].

The works described above require content or context data for new items. However, such data may not be available. In such cases, ratings for new items are arriving and one wishes to predict the missing ratings, with increasing accuracy, as more feedback is provided. Due to the incremental nature of the problem, one cannot afford to retrain the CF model with each arriving rating. A common heuristic approach is to estimate a new item's latent vector by a linear combination of the latent vectors of its raters and their respective ratings. It is based on the fact that users and items coexist as vectors in the same latent space and that ratings are usually estimated by user-item vector similarities. This principle was successfully practiced in various MF techniques [20, 14, 2, 3]. We note that this approach will serve us as baseline, as it does not require any content-based information and is thus suitable for our setting.

The optimal design approach. In the statistical field called *optimal design of experiments*, or just *optimal design* [4, 24], a statistician is faced with the task of choosing a limited amount of experiments to perform from a given pool, with the goal of gaining the most informative results. There are various measurements for the gain obtained from a subset of experiments. The measure of interest in our setting is referred to as the A-optimality criterion. According to this criterion, the experiments correspond to noisy outputs of a (usually linear) regression function and the goal is to minimize the Mean Squared Error (MSE) over all the possible inputs of the regressor.

An early attempt to tackle the user cold-start problem using optimal design techniques and an interviewing process appears in [25]. In this work, an adaptation of A-optimality, also referred to as *transductive optimal design*, aims to minimize the MSE over some arbitrary (yet known in advance) set of points (or users, in our context). Despite the original presentation in the setting of the user cold-start problem, the generality of the employed techniques allows handling

the item cold-start problem in a similar manner. However, their work differs from ours due to the absence of performance guarantee for the proposed algorithms, and the use of an inferior prediction model.

2. PRELIMINARIES AND MODEL

2.1 The Latent Factor Model

We denote by $\mathcal{I}$ the set of all items, and by $\mathcal{U}$ the set of all users. The cardinality of these sets is denoted by $|\mathcal{I}| = m$, and $|\mathcal{U}| = n$. We use $\mathcal{R}$ to denote the rating matrix, which is of size $n \times m$. A rating r_{ui} indicates the preference of item i by user u, where high values mean stronger preference. The matrix $\mathcal{R}$ is typically sparse, since most users rate just a small portion of the items.

Using the notation above, the LFM (with k denoting its dimension) seeks to estimate each rating r_{ui} as follows:

$$r_{ui} \approx \mu + b_i + b_u + Q_i^\top P_u \ , \tag{1}$$

where $b_i \in \mathbb{R}$ and $Q_i \in \mathbb{R}^k$ are the bias and the latent factor vector of item i, respectively; $b_u \in \mathbb{R}$ and $P_u \in \mathbb{R}^k$ are the bias and the latent factor vector of user u, respectively; and $\mu \in \mathbb{R}$ denotes the overall average rating in $\mathcal{R}$. We use ε_{ui} to denote the residual error of our model with respect to user u and item i.

Intuitively, the term $Q_i^\top P_u$ captures the interaction (or affinity) between user u and item i, where high values implies stronger preference and vice versa. We denote by Q the $k \times m$ matrix, whose columns correspond to the vectors Q_i for $i = 1, \ldots, m$, and by P the $k \times n$ matrix, whose columns correspond to the vectors P_u for $u = 1, \ldots, n$.

An important aspect of our work is the following: as training the LFM is a completely orthogonal task to ours, we assume that we are given an already trained model and refer to it as ground truth. Practically, we assume that each rating r_{ui} complies with the following noisy model:

$$r_{ui} = \mu + b_i + b_u + Q_i^\top P_u + \varepsilon_{ui} \ , \tag{2}$$

where ε_{ui} is assumed to be a zero-mean noise term. This assumption is very common and can be found in several previous works (see [25] for instance). Moreover, a simple sanity check can verify that the assumption complies with the employed LFM: test the significance of the residuals as generated from a Gaussian distribution.

2.2 Problem Definition

We proceed to formally define our problem. Let i be a new item ($i \notin \mathcal{I}$), and denote by $\mathcal{U}^i \subset \mathcal{U}$ the pool of available users to rate it. Also, we denote the budget constraint by B (i.e., B is the number of users we are allowed to assign with the task of rating item i), and use the notation $\mathcal{U}_B^i$ for subsets of $\mathcal{U}^i$, which are of size B.

Now, given a budget constraint B, our goal is to select B users to rate item i in order to minimize the expected MSE on the set of the remaining users $\mathcal{U} \setminus \mathcal{U}_B^i$. More formally, we wish to solve the following optimization problem:

$$\min_{\mathcal{U}_B^i \subset \mathcal{U}^i} \left\{ \mathbb{E}\left[\frac{1}{|\mathcal{U} \setminus \mathcal{U}_B^i|} \sum_{u \in \mathcal{U} \setminus \mathcal{U}_B^i} (\tilde{r}_{ui} - r_{ui})^2 \right] \right\},$$

where $\tilde{r}_{ui}$ denotes our prediction of r_{ui}, and the expectation is taken over the noise terms consisting the ratings $\{r_{ui}\}$ (as defined in Equation (2)).

To simplify the notations, in the sequel we refer to the set of the remaining users as $\mathcal{U}$ (instead of $\mathcal{U} \setminus \mathcal{U}_B^i$). Essentially, this means that the sets $\mathcal{U}$ and $\mathcal{U}_B^i$ are disjoint, and the problem of interest is thus translated to:

$$\min_{\mathcal{U}_B^i \subset \mathcal{U}^i} \left\{ \mathbb{E}\left[\frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} (\tilde{r}_{ui} - r_{ui})^2 \right] \right\}. \tag{3}$$

Clearly, the objective function inherently depends on how we generate the predictions $\{\tilde{r}_{ui}\}$ given the set of B users $\mathcal{U}_B^i$, and their ratings of item i. Thus we can divide the problem in Equation (3) into two distinct problems:

Rating Prediction (Problem A): given a subset of users $\mathcal{U}_B^i \subset \mathcal{U}^i$ and their ratings of item i, generate predictions $\{\tilde{r}_{ui}\}$ for all $u \in \mathcal{U}$.

User Selection (Problem B): given a pool of available users $\mathcal{U}^i$, select a subset of users $\mathcal{U}_B^i \subset \mathcal{U}^i$ to reveal their ranking of item i.

We shall point out that viewing the latent factor model as ground truth gives rise to an optimal solution for **Problem A**: least squares estimation (detailed in Section 3). However, in the CF literature there exist other approaches that can potentially function better in practice (especially if the data is not well represented by the LFM). These approaches are presented in Section 4.3, and will serve us as baselines for evaluating the merit of our approach.

3. ALGORITHMS AND ANALYSIS

In this section we tie together the two problems described in Section 2, and present a holistic approach that tackles the unified problem as a whole - the optimal design approach. *Optimal design* is a statistic paradigm [4, 24] that is aimed at selecting which experiments to conduct out of a given pool in order to maximize the result accuracy. In our setting, we wish to select a subset of users such that the prediction error of our model is minimized. Roughly speaking, the optimal design approach seeks to estimate the MSE (over the entire user set) for each selection of B users, *without* exposing their actual ratings. This way, we can evaluate the merit of any given subset of users (in terms of MSE), and make the selection accordingly.

Basically, the optimal design approach relies on the assumption that the ratings are generated via Equation (2), and consequently on the optimal solution for **Problem A**: least squares estimation. To make this point clearer we first introduce this estimator.

3.1 Least Squares Estimation

Assume we are given a new item $i \notin \mathcal{I}$, and a subset of users $\mathcal{U}_B^i \subset \mathcal{U}^i$ who provided their ratings for this item. Our task is to predict $\tilde{r}_{ui}$ for all $u \in \mathcal{U}$. Recall our modeling assumption regarding each of ratings r_{ui} from Equation (2). In particular, we assume that the ratings of the new item i by the users within $\mathcal{U}_B^i$ follow this model as well, where μ, b_u and P_u are given by our prediction model (Section 2), whereas b_i and Q_i are unknown to us (since the new item was not considered while training the model). Therefore, the least squares estimator seeks to estimate b_i and Q_i, and consequently to generate a prediction

$$\tilde{r}_{ui} = \mu + \tilde{b}_i + b_u + \tilde{Q}_i P_u.$$

From now on, we denote by (b_i, Q_i) the $(k+1)$-dimensional concatenation of b_i and the vector Q_i; the notation $(\tilde{b}_i, \tilde{Q}_i)$ shall stand for the corresponding estimator.

Using the notation above, the least squares estimator aims to estimate (b_i, Q_i) by minimizing the MSE over the set $\mathcal{U}_B^i$, that is, to solve the following problem:

$$\min_{\substack{b\in\mathbb{R}\\ q\in\mathbb{R}^k}} \left\{ \frac{1}{|\mathcal{U}_B^i|} \sum_{v\in\mathcal{U}_B^i} \left(r_{vi} - q^\top P_v - b - b_v - \mu \right)^2 \right\}.$$

This minimization problem is in fact analytically solvable, and yields the following solution:

$$\left(\tilde{b}_i, \tilde{Q}_i\right) = \left(\sum_{v\in\mathcal{U}_B^i} P'_v P'^\top_v \right)^{-1} \left(\sum_{v\in\mathcal{U}_B^i} (r_{vi} - b_v - \mu) P'_v \right),$$

where P'_v is the concatenated column vector $(1, P_v)$.

The above might not be well-defined since $\sum_{v\in\mathcal{U}_B^i} P'_v P'^\top_v$ need not be invertible in general. In practice, a regularization term of the form $\lambda\left(\|q\|_2^2 + b^2\right)$ is usually added to the objective function to avoid this problem. The estimator $(\tilde{b}_i, \tilde{Q}_i)$ then takes the following form:

$$\left(\lambda I + \sum_{v\in\mathcal{U}_B^i} P'_v P'^\top_v \right)^{-1} \left(\sum_{v\in\mathcal{U}_B^i} (r_{vi} - b_v - \mu) P'_v \right).$$

In the sequel, we will assume that $\sum_{v\in\mathcal{U}_B^i} P'_v P'^\top_v$ is invertible to ease the readability of the paper. We emphasize that all the results presented in this paper would still hold for the regularized estimator.

3.2 The Optimal Design Approach

We proceed to formally define necessary notations that will help us adapting the optimal design approach to our setting. We somewhat abuse notations and denote by P the matrix whose columns correspond to the latent factor vectors P'_u for $u \in \mathcal{U}$, and by P_B the matrix whose columns correspond to the latent factor vectors P'_v for $v \in \mathcal{U}_B^i$. Similarly, we use the notations $\varepsilon_{\mathcal{U}}$ and ε_B for vectors whose elements correspond to ε_{ui} for $u \in \mathcal{U}$ and ε_{vi} for $v \in \mathcal{U}_B^i$, respectively. Finally, we use r_B to denote the vector whose elements correspond to $(r_{vi} - b_v - \mu)$ for $v \in \mathcal{U}_B^i$.

We divide the analysis into two cases: (1) the noise terms $\{\varepsilon_{ui}\}$ are assumed to be zero-mean and i.i.d.; and (2) the noise terms $\{\varepsilon_{ui}\}$ are only assumed to be zero-mean and independent (but not necessarily identically distributed).

3.2.1 Identically Distributed Noise Terms

The following is our key observation: the optimization problem considered in Equation (3) can be reduced to a simpler problem, if the noise terms are assumed to be i.i.d. and the least squares estimator is used for estimating (b_i, Q_i). This observation is stated and proven below in Lemma 1.

Before we continue, we will assume without loss of generality that $PP^\top = |\mathcal{U}|\, I_{(k+1)\times(k+1)}$, meaning that the user vectors are in isotropic position. Otherwise , we can simply apply an invertible linear transformation to the LFM space of the users and its inverse to the space of the items; this does not affect our results as $(F^\top x)^\top (F^{-1} y) = x^\top y$ for all vectors x, y and invertible linear transformation F. This assumption is here merely to simplify the statements and proofs; the statements hold without it as well, and in particular, one is not required to compute the inverse of $PP^\top$ when implementing the algorithm.

Algorithm 1 Backward Greedy Selection (BGS1)

1: Input: users set $\mathcal{U}^i$, and corresponding matrix $P_{\mathcal{U}^i}$.
2: Output: users subset $\mathcal{U}_B^{ALG}$.
3: Initialize $P_B = P_{\mathcal{U}^i}$ and $\mathcal{U}_B^{ALG} = \mathcal{U}^i$.
4: **for** $j = 1$ to $|\mathcal{U}^i| - B$ **do**
5: $\quad v_j \leftarrow \arg\min_{v\in\mathcal{U}_B^{ALG}} \left\{ \mathrm{tr}\left(\left(P_{B\setminus v} P_{B\setminus v}^\top \right)^{-1} \right) \right\}$
6: $\quad$ Update $P_B \leftarrow P_{B\setminus v_j}$.
7: $\quad$ Update $\mathcal{U}_B^{ALG} \leftarrow \mathcal{U}_B^{ALG} \setminus v_j$.
8: **end for**

Lemma 1 *Assume there exist b_i and Q_i, such that for every $u \in \mathcal{U}$ it holds that $r_{ui} = \mu + b_u + b_i + Q_i^\top P_u + \varepsilon_{ui}$, where $\mathbb{E}[\varepsilon_{ui}] = 0$ and $\mathbb{E}[\varepsilon_{ui}^2] = \sigma^2$. Then, the following is equivalent[1] to the problem presented in Equation (3):*

$$\min_{\mathcal{U}_B^i\subset\mathcal{U}^i} \left\{ \sigma^2 tr\Big(\left(P_B P_B^\top \right)^{-1} \Big) + \sigma^2 \right\},$$

if the least squares estimator is considered.

The expression in the lemma above has the following interpretation: the additive σ^2 term is the inherent model error. I.e., this is the term that cannot be avoided as long as we assume a latent factor model of dimension k. The second term of $\sigma^2 \mathrm{tr}\left(\left(P_B P_B^\top \right)^{-1} \right)$ represents the error originating from a sub-optimal choice of the item's parameters, i.e., from the distance between $(\tilde{b}_i, \tilde{Q}_i)$ and (b_i, Q_i). The proof of the lemma appears in Appendix A.1.

After establishing a new (and equivalent) optimization problem, we turn now to present a simple greedy algorithm for finding an approximation to its optimum. Here, $P_{B\setminus v_j}$ refers to the matrix P_B, where the column that corresponds to the user v_j is removed. Before stating our main theorem, we introduce some necessary definitions:

Definition 1 *let $f : \mathbb{R}_+ \to \mathbb{R}$. For a positive definite matrix $A \in R^{k\times k}$ with the decomposition $A = U^\top diag(\lambda_1, \ldots, \lambda_k) U$, we extend the definition of f as follows:*

$$f(A) = U^\top diag(f(\lambda_1), \ldots, f(\lambda_k)) U.$$

Definition 2 *We say that f is* operator monotone *if for any positive definite matrices A and B:*

$$A \preceq B \Rightarrow f(A) \preceq f(B).$$

Definition 3 *We say that f is* operator antitone *if $-f$ is operator monotone.*

Definition 4 *We say that $f : 2^E \to \mathbb{R}$ is* supermodular *if*

$$f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B),$$

for any $A \subset B \subset E$ and $x \in E \setminus B$.

[1]The equivalence here between the two problems is in the following sense: the value of both objective functions is the same for any subset $\mathcal{U}_B^i \subset \mathcal{U}^i$.

Definition 5 *The steepness of a decreasing supermodular function f is denoted by s and defined as follows:*

$$s = \max_{x \in E} \frac{[f(\phi) - f(x)] - [f(E \setminus \{x\}) - f(E)]}{[f(\phi) - f(x)]}.$$

In the last definition, if $f(\phi)$ is not defined we can simply extend it as follows:

$$f(\phi) = \max_{\substack{A,B \subset E \\ A \cap B = \phi}} \{f(A) + f(B) - f(A \cup B)\} \quad (4)$$

The following is our main theorem:

Theorem 2 *Let i be a new item, and $\mathcal{U}^i$ its corresponding set of available users, and assume that $\mathbb{E}[\varepsilon_{ui}] = 0$ and $\mathbb{E}\left[\varepsilon_{ui}^2\right] = \sigma^2$ for all $u \in \mathcal{U}$. Then, Algorithm 1 generates a subset $\mathcal{U}_B^{ALG} \subset \mathcal{U}^i$ for which:*

$$\mathbb{E}\left[MSE(\mathcal{U}_B^{ALG})\right] \leq \sigma^2 + \frac{e^t - 1}{t}\left(\mathbb{E}\left[MSE(\mathcal{U}_B^*)\right] - \sigma^2\right).$$

where $t = \frac{s}{1-s}$ and $\mathcal{U}_B^*$ is the optimal subset of B users, minimizing the expected MSE.

The above statement can be interpreted as follows. Both Algorithm 1 and the optimal solution have an additive term of σ^2, corresponding to the inherent LFM error. The additional term, corresponding to the sub-optimality of the choice of $(\tilde{b}_i, \tilde{Q}_i)$, is multiplicatively approximated by a factor of $(e^t - 1)/t$ compared to the optimal selection. Notice that the stated guarantee strictly dominates the following one:

$$\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_B^{ALG})\right] \leq \frac{e^t - 1}{t}\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_B^*)\right],$$

since $\sigma^2 > 0$ (trivially holds). We shall proceed to prove the main theorem.

Proof. We first define an auxiliary function $\Phi : 2^{\mathcal{U}^i} \to \mathbb{R}$ as follows:

$$\Phi(\mathcal{U}_D^i) = \mathrm{tr}\left(\left(P_D P_D^\top\right)^{-1}\right) - \varphi_i,$$

where $\varphi_i = \mathrm{tr}\left(\left(P_{\mathcal{U}^i} P_{\mathcal{U}^i}^\top\right)^{-1}\right)$, and $\Phi(\varnothing)$ is defined as in Equation (4). Notice that Φ is well-defined for any subset $\mathcal{U}_D^i \subset \mathcal{U}^i$ (of size $0 \leq D \leq \left|\mathcal{U}^i\right|$), and not only for subsets of size B. The matrix P_D is then defined as the matrix whose columns correspond to the latent factor vectors P_v' for $v \in \mathcal{U}_D^i$.

Next, we show that Φ has the following three properties: it is supermodular, monotonically decreasing, and equal to 0 at $\mathcal{U}^i$. For such functions, [12] proved that the backward greedy algorithm, eliminating elements one by one, attains an $\left(\frac{e^t-1}{t}\right)$-approximation to the optimum. The third property follows immediately from the definitions of Φ and φ_i, and thus we are left to prove the other two properties.

Before proving these properties, we provide a simple intuition. Notice first that $\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_D^i)\right]$ can be written as a linear function of $\Phi(\mathcal{U}_D^i)$:

$$\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_D^i)\right] = \sigma^2\left(\Phi(\mathcal{U}_D^i) + \varphi_i + 1\right).$$

Thus, minimizing Φ is equivalent to minimizing the MSE. Now, the monotonicity of Φ gives rise to the following insight: the MSE decreases as the number of users increases. The supermodularity then translates into the following rational: the marginal return of a user diminishes as the set of selected users expands.

We shall proceed to formally prove these properties. For the monotonicity, we take two subsets $\mathcal{U}_D^i, \mathcal{U}_E^i \subset \mathcal{U}^i$, such that $\mathcal{U}_D^i \subset \mathcal{U}_E^i$, and prove that $\Phi(\mathcal{U}_D^i) \geq \Phi(\mathcal{U}_E^i)$. Thus, denote by P_D and P_E the matrices that correspond to $\mathcal{U}_D^i$ and $\mathcal{U}_E^i$, respectively. Then, it holds that $P_E P_E^\top \succeq P_D P_D^\top$, which also implies that

$$\left(P_E P_E^\top\right)^{-1} \preceq \left(P_D P_D^\top\right)^{-1}.$$

Now, since the trace of a positive semidefinite (PSD) matrix is the sum of its eigenvalues, the above also implies that

$$\mathrm{tr}\left(\left(P_E P_E^\top\right)^{-1}\right) \leq \mathrm{tr}\left(\left(P_D P_D^\top\right)^{-1}\right),$$

which proves that Φ is monotonically decreasing.

For the supermodularity, we use recent techniques presented in [23], and specifically the following proposition:

Proposition 3 *Let $\mathcal{U}_D^i \subset \mathcal{U}^i$, and P_D be the matrix whose columns correspond to P_v' for $v \in \mathcal{U}_D^i$. Then, for $f : \mathbb{R}_+ \to \mathbb{R}$ such that f' is operator monotone, it holds that*

$$\Phi(\mathcal{U}_D^i) = tr\left(f\left(P_D P_D^\top\right)\right)$$

is supermodular.

In our case $f(x) = \frac{1}{x}$, and its derivative $f'(x) = -\frac{1}{x^2}$ is operator monotone, which implies that Φ is supermodular. The proof resembles the proof of Corollary 2.4 of [23], albeit plugging operator monotone derivative instead of operator antitone one.

Finally, as mentioned before, these three properties yield an $\left(\frac{e^t-1}{t}\right)$-approximation to the problem of minimizing Φ. That is, Algorithm 1 generates a subset $\mathcal{U}_B^{ALG}$ for which:

$$\begin{aligned}
\Phi\left(\mathcal{U}_B^{ALG}\right) &\leq \left(\frac{e^t - 1}{t}\right) \min_{\mathcal{U}_B^i \subset \mathcal{U}^i} \Phi\left(\mathcal{U}_B^i\right) \\
&= \left(\frac{e^t - 1}{t}\right) \min_{\mathcal{U}_B^i \subset \mathcal{U}^i} \left\{\frac{\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_B^i)\right]}{\sigma^2} - 1 - \varphi_i\right\} \\
&\leq \left(\frac{e^t - 1}{t}\right) \min_{\mathcal{U}_B^i \subset \mathcal{U}^i} \left\{\frac{\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_B^i)\right]}{\sigma^2} - 1\right\} - \varphi_i \\
&= \left(\frac{e^t - 1}{t}\right)\left(\frac{\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_B^*)\right]}{\sigma^2} - 1\right) - \varphi_i.
\end{aligned}$$

By substituting $\Phi(\mathcal{U}_B^{ALG}) = \frac{\mathbb{E}\left[\mathrm{MSE}(\mathcal{U}_B^{ALG})\right]}{\sigma^2} - \varphi_i - 1$ in the inequality above, we get the stated result. □

3.2.2 Non-Identically Distributed Noise Terms

The analysis in Section 3.2.1 relies heavily on the assumption that $\{\varepsilon_{ui}\}$ are identically distributed. However, it is sometimes reasonable to assume that each user has a different noise distribution with respect to the employed model. In such cases, the identical distribution assumption only weakens our model. In this section we extend our analysis to the case where each of the users has a different (yet known in advance) noise distribution.

Thus, let i be a new item, and $\mathcal{U}_B^i \subset \mathcal{U}^i$ a subset of users assigned for the task of rating it. We start by introducing a new estimator for (b_i, Q_i), which generalizes the least

Algorithm 2 Backward Greedy Selection (BGS2)

1: Input: users set $\mathcal{U}^i$, and corresponding matrix $P_{\mathcal{U}^i}$.
2: Output: users subset $\mathcal{U}_B^{ALG}$.
3: Initialize $P_B = P_{\mathcal{U}^i}$, $C_B = C_{\mathcal{U}^i}$, and $\mathcal{U}_B^{ALG} = \mathcal{U}^i$.
4: **for** $j = 1$ to $|\mathcal{U}^i| - B$ **do**
5: $\quad v_j \leftarrow \arg\min_{v \in \mathcal{U}_B^{ALG}} \left\{ \mathrm{tr}\left(\left(P_{B\setminus v} C_{B\setminus v}^{-2} P_{B\setminus v}^\top \right)^{-1} \right) \right\}$
6: $\quad$ Update $P_B \leftarrow P_{B\setminus v_j}$ and $C_B \leftarrow C_{B\setminus v_j}$.
7: $\quad$ Update $\mathcal{U}_B^{ALG} \leftarrow \mathcal{U}_B^{ALG} \setminus v_j$.
8: **end for**

squares estimator:

$$(\tilde{b}_i, \tilde{Q}_i) = \left(P_B C_B^{-2} P_B^\top \right)^{-1} P_B C_B^{-2} r_B,$$

where again, C_B is the square root of the covariance matrix that corresponds to ε_{vi} for $v \in \mathcal{U}_B^i$. Intuitively, this estimator is a least squares estimator that considers variance-scaled users. In the sequel, we refer to this estimator as *generalized least squares estimator*. The following lemma states that this is indeed an unbiased estimator for (b_i, Q_i). The proof is technical and appears in Appendix A.2.

Lemma 4 *Assume there exist b_i and Q_i, such that for every $u \in \mathcal{U}$ it holds that $r_{ui} = \mu + b_u + b_i + Q_i^\top P_u + \varepsilon_{ui}$, where $\mathbb{E}[\varepsilon_{ui}] = 0$ and $\mathbb{E}[\varepsilon_{ui}^2] = \sigma_u^2$. Then,*

$$(\tilde{b}_i, \tilde{Q}_i) = \left(P_B C_B^{-2} P_B^\top \right)^{-1} P_B C_B^{-2} r_B \tag{5}$$

is an unbiased estimator for $(b_i, Q_i^\top)$.

We proceed to state the lemma that encapsulates our parallel key observation for the case of non-identically distributed noise terms; its proof appears in Appendix A.3.

Lemma 5 *Assume there exist b_i and Q_i, such that for every $u \in \mathcal{U}$ it holds that $r_{ui} = \mu + b_u + b_i + Q_i^\top P_u + \varepsilon_{ui}$, where $\mathbb{E}[\varepsilon_{ui}] = 0$ and $\mathbb{E}[\varepsilon_{ui}^2] = \sigma_u^2$. Then, the following is equivalent to the problem presented in Equation* (3)*:*

$$\min_{\mathcal{U}_B^i \subset \mathcal{U}^i} \left\{ tr\left(\left(P_B C_B^{-2} P_B^\top \right)^{-1} \right) + \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sigma_u^2 \right\},$$

if the generalized least squares estimator is considered.

After establishing here also an equivalent optimization problem, we turn to present a simple adaptation of Algorithm 1 to this problem. Here, the notation $C_{\mathcal{U}^i}$ refers to the square root of the covariance matrix of all available users, and the notations $P_{B\setminus v_j}$ and $C_{B\setminus v_j}$ are as defined in Section 3.2.1. For Algorithm 2 we can prove the following:

Theorem 6 *Let i be a new item, and $\mathcal{U}^i$ its corresponding set of available users, and assume that $\mathbb{E}[\varepsilon_{ui}] = 0$ and $\mathbb{E}[\varepsilon_{ui}^2] = \sigma_u^2$ for all $u \in \mathcal{U}$. Then, Algorithm 2 generates a subset $\mathcal{U}_B^{ALG} \subset \mathcal{U}^i$, for which the following holds:*

$$\mathbb{E}\left[MSE(\mathcal{U}_B^{ALG}) \right] \leq \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sigma_u^2 + \frac{e^t - 1}{t} \left(\mathbb{E}\left[MSE(\mathcal{U}_B^*) \right] - \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sigma_u^2 \right).$$

The proof relies on the same techniques of Theorem 2, albeit plugging the variance-scaled matrix $P_B C_B^{-1}$ instead of the matrix P_B, and is thus omitted here.

Reduction to unknown variances. Until now, we assumed that each user has its own noise variance with respect to the model. Our analysis relied on the fact that these variances are known to us in advance. Clearly, this is never the case in real life. However, in the setting of CF-based recommender systems, we can utilize the past ratings of a given user to estimate this variance. Formally, let $\mathcal{R}(u)$ be the set of items that user u has rated. Then, σ_u^2 can be estimated as follows:

$$\tilde{\sigma}_u^2 = \frac{1}{|\mathcal{R}(u)|} \sum_{i \in \mathcal{R}(u)} (\tilde{r}_{ui} - r_{ui})^2,$$

where $\tilde{r}_{ui}$ is as predicted by the LFM (Equation (1)). In the experimental section, we show that despite the fact that the variances are estimated and not known, Algorithm 2 outperforms all the baselines (including Algorithm 1).

4. EXPERIMENTAL RESULTS

4.1 Dataset and Model

We consider a large movie ratings dataset released by Netflix as the basis of a well publicized competition [5]. The original dataset contains more than 100 million date-stamped ratings (integers between 1 and 5) collected between Nov. 11, 1999 and Dec. 31, 2005, from about 480,000 anonymous Netflix customers on 17,770 movies.

The dataset is processed as follows: a new ratings matrix (denoted henceforth by $\mathcal{R}$) is constructed from 17,000 arbitrarily chosen movies. The matrix $\mathcal{R}$ contains about 96 million ratings from the same number of users as in the original dataset. The other 770 movies are used as "new items" in the various experiments we conduct[2]. Note that since most users have not rated most of the movies, the ratings matrix $\mathcal{R}$ is very sparse (only 1% of all ratings exist).

We factorize $\mathcal{R}$ using the LFM (Equation (1)), with its dimension set to be $k = 20$. We use the common Root Mean Squared Error (RMSE) metric to measure the prediction accuracy of the resulted model. For the task of minimizing this metric, we apply the Stochastic Gradient Descent (SGD) algorithm, where the learning rate is inspired by the AdaGrad algorithm of [7].

4.2 Experimental Setup

We now describe how we test algorithms for user selection in an offline manner on the Netflix dataset. As mentioned before, a set of 770 movies is withheld from our model. We henceforth denote this set by $\mathcal{N}$, and think of it as a set of movies that are new to our recommender system. For each movie $i \in \mathcal{N}$, we denote by $\mathcal{R}(i)$ the set of Netflix users who have actually rated this movie. We then adapt the dataset to our setting as follows:

The pool of available users: we consider the set $\mathcal{R}(i)$ as the pool of available users to rate movie i, i.e., we set $\mathcal{U}_i = \mathcal{R}(i)$. Note that indeed, $\mathcal{U}^i$ changes for every movie i, as the actual available ratings differ from one

[2]Note we cannot use one of the official Netflix test sets, as none of them captures the notion of new items.

movie to another in the Netflix dataset. This coincides with the pool of available users changing in time.

Selecting a subset of users: the conceptual step of assigning a subset of users $\mathcal{U}_B^i \subset \mathcal{U}^i$ to rate movie i corresponds to the action of revealing their actual ratings in the dataset.

The set of all users: due to the small portion of actual ratings in the dataset, we cannot measure the prediction error on all users. Thus, after revealing the actual ratings of the subset $\mathcal{U}_B^i$, we use the remaining unrevealed ratings for this task, i.e., we set $\mathcal{U} = \mathcal{U}_i \setminus \mathcal{U}_B^i$. Note that we *essentially use the non-selected users for the evaluation task* in this step.

Despite the fact that our analysis was carried out using the MSE metric, we measure the prediction error of the different approaches using the RMSE metric:

$$\mathrm{RMSE}(\mathcal{U}_B^i) = \sqrt{\frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}(r_{ui}-\tilde{r}_{ui})^2}.$$

Since the root square is a monotone function, minimizing the MSE is equivalent to minimizing the RMSE (in the sense that both problems has the same minimizer). Thus, to be consistent with previous works, our experimental results are presented using the RMSE metric.

An important aspect of our experimental setup is the preselection of users who watched the "new" movie. This might seem unrealistic, since generally there is no guarantee that users will rate the new item (as we implicitly assume). We justify our setup as follows: each user we select to rate the new item is given some reward, which is (in most cases) sufficient to convince the user to provide rating. This mitigates the need to select "frequent raters", and highlights the rather interesting geometric aspect of the problem: selected users should be scattered (in a sense) in the LFM space.

4.3 Baselines

We turn to present several common and less common baselines from the recent literature. Recall that our problem (as defined in Equation (3)) was divided into two separate problems, and thus we consider two types of baselines. We begin with baselines for **Problem A**: rating prediction.

Similarity based estimator. Just like the least squares estimator, this baseline seeks to estimate (b_i, Q_i) and consequently generate a prediction $\tilde{r}_{ui} = \mu + \tilde{b}_i + b_u + \tilde{Q}_i P_u$. The estimation uses the assigned users who liked the item, and relies on the following underlying assumption: "good" user-item interaction is expressed by the closeness of the corresponding representative vectors in the LFM space. Thus, yielding the estimator:

$$\tilde{b}_i = \frac{1}{|\mathcal{U}_B^i|}\sum_{v\in\mathcal{U}_B^i}(r_{vi}-b_v)-\mu\ ,$$

$$\tilde{Q}_i = \frac{1}{|\{v\in\mathcal{U}_B^i | r_{vi}\geq\gamma\}|}\sum_{v\in\mathcal{U}_B^i | r_{vi}\geq\gamma} P_v\ ,$$

where γ is a predefined threshold (e.g., 4 or 5 for the Netflix dataset). We note that integrating negative feedback (low ratings) yields an empirically inferior similarity based estimator that is thus omitted here. This approach was successfully practiced in [20, 14, 2, 3].

We continue with presenting baselines for **Problem B**: user selection.

Forward greedy selection (B1). This baseline relies as well on the optimal design approach, and specifically on the A-optimality criterion. Basically, it aims to minimize the trace of the inverse of the information matrix $P_B P_B^\top$, similarly to our first algorithm. However, the selection is done in a forward manner: starting with an empty set, and greedily adding the user whose contribution is the highest. This approach was successfully practiced in many fields of active learning, rather than only for the item cold-start problem of CF-based recommenders (see [17] for a more comprehensive survey). However, we are not aware of any performance guarantee for this algorithm.

Clustering (B2). Intuitively, having a heterogeneous set of users will result in a more objective ratings of the item. To this end, we suggest to cluster the users with respect to their representation in the Euclidean LFM space and subsequently sample from the resulting clusters. For the clustering task, we use the standard k-means algorithm with cosine similarity as its similarity measure.

We propose two clustering-based baselines for the selection problem, each differs in the number of clusters and the sampling procedure: (1) split the pool of available users into B clusters, and select the users that are closest to the cluster center of mass (one user per cluster); and (2) split the pool of available users into c clusters (where $c < B$), and from each cluster select number of users at random and proportionally to the cluster size. For this baseline, the value of c is determined empirically.

Random selection (B3). In this baseline, the B users are selected randomly from the pool of available users. At first glance, this baseline seems rather weak. However, if B is sufficiently large, then $\mathcal{U}_B^i$ is a statistically good representative of $\mathcal{U}^i$, and this baseline cannot be easily beaten. Moreover, this baseline has many practical appeals and is thus very popular with state-of-the-art recommenders.

Frequent raters (B4). Here we select users who tend to provide many ratings. This baseline utilizes the following conjecture: users with extensive rating history have better modeling in CF-based recommender systems. Therefore, preferring these users over users that have limited rating history seems reasonable in our setting.

Edgy raters (B5). This baseline selects users that provide diverse ratings, or more accurately, ratings with large variance. Intuitively, these users provide more information regarding an item they like or dislike, than users that tend to rate all items in a similar manner.

Early birds raters (B6). This baseline slightly differs from the previous baselines: instead of actively selecting users to rate the new item, we conceptually invite all users to opine on the new item at their convenience, and consider the (chronologically) first B returned ratings as our selection. The intention here is not to offer another active selection baseline, but rather to study the utility of early reviews, coming from people who are enthusiastic about trying out the new item, for modeling the rest of the population.

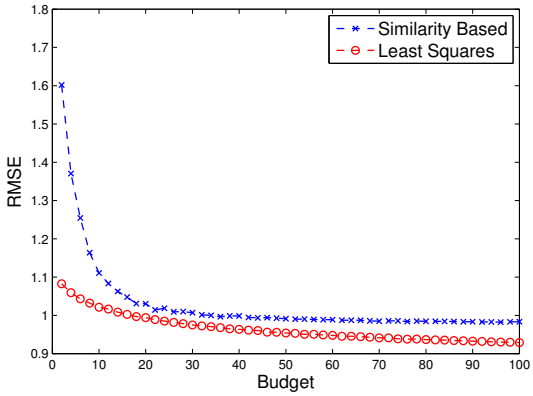

Figure 1: Rating prediction comparison.

4.4 Results

We separately report the results for each of the considered problems, starting with the rating prediction problem and moving on to the user selection problem.

4.4.1 *Rating Prediction Comparison*

We compare the performance of the least squares estimator and the similarity based estimator. To this end, we select 300 movies from the set $\mathcal{N}$, arbitrarily. For each of these movies, we randomly select B users ($B = 2, 4, \ldots, 100$), and apply both estimators to generate predictions for the remaining users. The prediction accuracy is measured using the RMSE metric over all predictions (and not separately for each movie). This routine is averaged over 50 runs to ensure stability.

As evident in Figure 1, the least squares estimator outperforms the considered baseline. Qualitatively, similar results were obtained when applying the other (non-random) selection schemes of B users, and are thus omitted here.

4.4.2 *User Selection Comparison*

Here the performance of our approach is compared with the baselines (presented in Section 4.3) for **Problem B**. To carry out the experiment, we select 300 movies from the set $\mathcal{N}$, arbitrarily. For each of these movies, we apply all six baselines (B1-B6) and our two algorithms (denoted by BGS1 and BGS2) to select a subset of B users. We then generate $(\tilde{b}_i, \tilde{Q}_i)$ using the least squares estimator (or the generalized least squares estimator for BGS2), regardless of how the selection was made. Since this estimator outperforms the similarity based estimator for any selection of B users (as demonstrated in Figure 1), we do not lose information by omitting them. Here again, we measure the prediction accuracy using the RMSE metric over all predictions. For the non-deterministic baselines: Clustering (B2) and Random (B3), we average the results over 50 runs to ensure stability.

As can be seen in Figure 2, the optimal design based algorithms outperform the other baselines for almost any budget. In particular, BGS2 (which accounts for independent yet not identically distributed noises) significantly [3] surpasses BGS1 and B1 (which are practically indistinguishable) for any budget. The second best performing baseline is B5, which utilizes the ratings of the "edgy" users. We conjecture that such users provide more information regarding a new item, in addition to being well-scattered in the LFM space. Baselines B2 and B3 (Random and Clustering[4]) seem to perform similarly in our setting, yet a closer look discovers that by clustering the users in the LFM space we get more stable results (in the sense of lower variance of the resulted RMSE). We remind the reader that our basic CF model was trained using state-of-the-art algorithms, where every percent of improvement is not easily achieved.

[3]The results are significant at any significance level, since we average them over $|\mathcal{N}| = 300$ movies.

5. CONCLUSIONS AND FUTURE WORK

In this work we started with a mature LFM model and tried to tackle the item cold-start problem by answering two questions: (a) given ratings of B users, how can we estimate other users' ratings of an arbitrary new item without retraining the model? and (b) how to choose the set of B users? We showed that in order to get approximately optimal results in terms of mean squared error, the above questions should be tackled jointly. In particular, we applied optimal design techniques and devised two greedy algorithms that achieve that goal under certain assumptions. We used the Netflix dataset to demonstrate the superiority of the proposed algorithms over a set of previously considered and non-considered baselines.

Our work can be seen as a "one-shot", non-adaptive active learning scheme that selects the B users at once. This leaves three interesting variants for future work. The first variant we consider is an *adaptive active learning* scheme, in which users are chosen sequentially, with each selection done only after receiving the ratings of the previously chosen user. The second variant tackles an *online* version of the problem, where potential users arrive one by one and the publisher must decide whether to assign them the item or not. In several recommendation settings, e.g. news recommendations, new items constantly arrive and have very short lifetimes, thereby requiring that the publisher identifies quickly which users to recommend them to. The tradeoff between (a) selecting a good set of users, and (b) quickly completing the reviewing process of the item, presents a challenging optimization problem. The third variant discusses a *multi-item* setting, where the publisher must concurrently explore several new items while users can review only one or a small fixed number of items. Note that the multi-item setting can be investigated in both the offline and the online settings.

In addition to the above variants, an interesting direction would be to study the user cold-start problem using the techniques we presented in this work. As discussed in the related work section, the user and the item cold-start problem are essentially different, mainly due to the ability to adaptively interview new users in order to bootstrap their modeling. Whereas this enables decision tree type solutions for user cold-start handling (which are currently the state-of-the-art), such solutions cannot be implemented in the setting considered in this paper. However, applying the optimal design approach to the user cold-start problem is possible and may potentially lead to better theoretical and empirical guarantees.

[4]We present here only the better performing clustering technique, which is the second one discussed in Section 4.3.

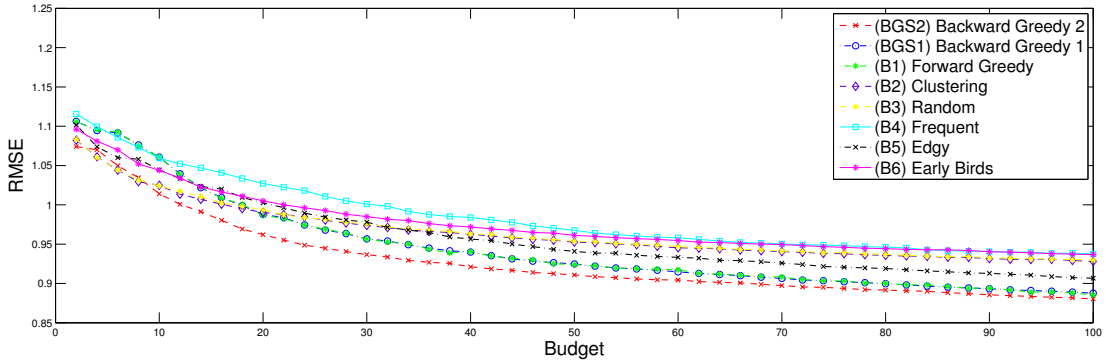

Figure 2: User selection comparison.

6. REFERENCES

[1] Deepak Agarwal and Bee-Chung Chen. Regression-based latent factor models. In *Proc. KDD*, 2009.

[2] Michal Aharon, Amit Kagian, Yehuda Koren, and Ronny Lempel. Dynamic personalized recommendation of comment-eliciting stories. In *Proc. RecSys*, 2012.

[3] Natalie Aizenberg, Yehuda Koren, and Oren Somekh. Build your own music recommender by modeling internet radio streams. In *Proc. WWW*, 2012.

[4] A. C. Atkinson and A. N. Donev. *Optimum Experimental Designs*. Oxford Univ. Press, 1992.

[5] J. Bennett and S. Lanning. The netflix prize. In *Proc. KDD Cup and Workshop*, 2007.

[6] Gideon Dror, Noam Koenigstein, and Yehuda Koren. Yahoo! music recommendations: Modeling music ratings with temporal dynamics and item. In *Proc. RecSys*, 2011.

[7] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research (JMLR)*, 12:2121–2159, 2011.

[8] Nadav Golbandi, Yehuda Koren, and Ronny Lempel. On bootstrapping recommender systems. In *Proc. CIKM*, 2010.

[9] Nadav Golbandi, Yehuda Koren, and Ronny Lempel. Adaptive bootstrapping of recommender systems using decision trees. In *Proc. WSDM*, 2011.

[10] Asela Gunawardana and Christopher Meek. Tied boltzmann machines for cold start recommendations. In *Proc. RecSys*, 2008.

[11] Asela Gunawardana and Christopher Meek. A unified approach to building hybrid recommender systems. In *Proc. RecSys*, 2009.

[12] Victor P. Il'ev. An approximation guarantee of the greedy descent algorithm for minimizing a supermodular set function. *Discrete Applied Mathematics*, 114(1):131–146, 2001.

[13] Arnd Kohrs and Bernard Merialdo. Improving collaborative filtering for new users by smart object selection. In *Proc. International Conference on Media Features (ICMF)*, 2001.

[14] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proc. KDD*, 2008.

[15] Yehuda Koren. Collaborative filtering with temporal dynamics. *Commun. of the ACM*, 53(4):89–97, 2010.

[16] Yehuda Koren, Robert M. Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer, IEEE*, 42(8):30–37, 2009.

[17] Andreas Krause and Carlos Guestrin. Near-optimal observation selection using submodular functions. In *AAAI*, volume 7, pages 1650–1654, 2007.

[18] Shao-Lun Lee. Commodity recommendations of retail business based on decision tree induction. *Expert Systems with Applications*, 37(5):3685–3694, 2010.

[19] Seung-Taek Park and Wei Chu. Pairwise preference regression for cold-start recommendation. In *Proc. RecSys*, 2009.

[20] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proc. KDD cup and workshop*, 2007.

[21] Al Mamunur Rashid, Istvan Albert, Dan Cosley, Shyong K Lam, Sean M McNee, Joseph A Konstan, and John Riedl. Getting to know you: learning new user preferences in recommender systems. In *Proc. International Conference on Intelligent User Interfaces*, 2002.

[22] Al Mamunur Rashid, George Karypis, and John Riedl. Learning preferences of new users in recommender systems: an information theoretic approach. *ACM SIGKDD Explorations Newsletter*, 10(2):90–100, 2008.

[23] Guillaume Sagnol. Approximation of a maximum-submodular-coverage problem involving spectral functions, with application to experimental designs. *Discrete Applied Mathematics*, 161(1):258–276, 2013.

[24] Chien-Fu Wu. Some algorithmic aspects of the theory of optimal designs. *Annals of Statistics*, 6(6):1286–1301, 1978.

[25] Kai Yu, Jinbo Bi, and Volker Tresp. Active learning via transductive experimental design. In *Proc. ICML*, 2006.

[26] Ke Zhou, Shuang-Hong Yang, and Hongyuan Zha. Functional matrix factorizations for cold-start recommendation. In *Proc. SIGIR*, 2011.

APPENDIX

A. PROOFS

A.1 Proof of Lemma 1

Let i be a new item, and $\mathcal{U}^i$ its corresponding set of available users. Now, given a set of B users $\mathcal{U}^i_B \subset \mathcal{U}^i$, we can estimate (b_i, Q_i) using the least squares estimator:

$$\begin{aligned}\left(\tilde{b}_i, \tilde{Q}_i\right) &= \left(\sum_{v\in\mathcal{U}^i_B} P'_v P'^{\top}_v\right)^{-1}\left(\sum_{v\in\mathcal{U}^i_B}(r_{vi} - b_v - \mu)P'_v\right)\\ &= \left(P_B P_B^{\top}\right)^{-1} P_B r_B. \end{aligned} \tag{6}$$

Notice that according to Equation (2) we can also express the vector r_B as follows:

$$r_B = P_B^{\top}\left(b_i, Q_i\right) + \varepsilon_B,$$

and by substituting the above in Equation (6) and shifting sides we get that

$$\left(\tilde{b}_i, \tilde{Q}_i\right) - \left(b_i, Q_i\right) = \left(P_B P_B^{\top}\right)^{-1} P_B \varepsilon_B. \tag{7}$$

We denote by $\mathrm{MSE}(\mathcal{U}^i_B)$ the resulted MSE value by using the ratings of the users within $\mathcal{U}^i_B$ in our prediction model, that is, $\mathrm{MSE}\big(\mathcal{U}^i_B\big) = \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\left(\tilde{r}_{ui} - r_{ui}\right)^2$.

Then, we can derive:

$$\begin{aligned}\mathbb{E}\Big[\mathrm{MSE}\big(\mathcal{U}^i_B\big)\Big] &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\sum_{u\in\mathcal{U}}\left(\tilde{r}_{ui} - r_{ui}\right)^2\right]\\ &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(\left(\tilde{b}_i, \tilde{Q}_i\right) - \left(b_i, Q_i\right)\right) + \varepsilon_{\mathcal{U}}\right\|_2^2\right]\\ &\overset{(1)}{=} \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(P_B P_B^{\top}\right)^{-1} P_B\varepsilon_B + \varepsilon_{\mathcal{U}}\right\|_2^2\right]\\ &\overset{(2)}{=} \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(P_B P_B^{\top}\right)^{-1} P_B\varepsilon_B\right\|_2^2 + \left\|\varepsilon_{\mathcal{U}}\right\|_2^2\right],\end{aligned}$$

where the expectation is taken over the noise ε_{vi} and ε_{ui}. Equality (1) follows by substituting Equation (7); equality (2) holds since $\mathbb{E}\left[\varepsilon_{ui}\right] = 0$ for all $u \in \mathcal{U}$, and also since ε_B and $\varepsilon_{\mathcal{U}}$ are independent.

Next, we use the assumption $\mathbb{E}\left[\varepsilon_{ui}^2\right] = \sigma^2$ to get:

$$\begin{aligned}\mathbb{E}\left[\mathrm{MSE}\big(\mathcal{U}^i_B\big)\right] &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(P_B P_B^{\top}\right)^{-1} P_B\varepsilon_B\right\|_2^2\right] + \sigma^2\\ &\overset{(3)}{=} \frac{1}{|\mathcal{U}|}\left\|P^{\top}\left(P_B P_B^{\top}\right)^{-1} P_B C_B\right\|_F^2 + \sigma^2\\ &\overset{(4)}{=} \sigma^2\mathrm{tr}\left(\left(P_B P_B^{\top}\right)^{-1}\right) + \sigma^2,\end{aligned}$$

where C_B denotes the square root of the covariance matrix of ε_B; equality (3) follows from the definition of the Frobenius norm $\|\|_F$, and equality (4) holds since $C_B = \sigma^2 I_{B\times B}$, and since we assume that $PP^{\top} = |\mathcal{U}|\, I_{(k+1)\times(k+1)}$.

Thus, the lemma is obtained.

A.2 Proof of Lemma 4

By taking expectation we have that:

$$\begin{aligned}\mathbb{E}\left[\left(\tilde{b}_i, \tilde{Q}_i\right)\right] &= \mathbb{E}\left[\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2} r_B\right]\\ &= \mathbb{E}\left[\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2} P_B^{\top}\left(b_i, Q_i\right)\right]\\ &\quad + \mathbb{E}\left[\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2}\varepsilon_B\right]\\ &= \left(b_i, Q_i\right) + \left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2}\mathbb{E}\left[\varepsilon_B\right]\\ &= \left(b_i, Q_i\right),\end{aligned}$$

where in the second equality we used the assumption that $r_B = P_B^{\top}\left(b_i, Q_i\right) + \varepsilon_B$.

A.3 Proof of Lemma 5

Recall that as before we assume that

$$PP^{\top} = |\mathcal{U}|\, I_{(k+1)\times(k+1)}\ .$$

Now, let i be a new item, and $\mathcal{U}^i_B \subset \mathcal{U}^i$ a subset of users assigned for the task of rating it. Then, we can substitute r_B in the generalized least squares estimator (Equation (5)) to get the following result:

$$\left(\tilde{b}_i, \tilde{Q}_i\right) - \left(b_i, Q_i\right) = \left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2}\varepsilon_B.$$

We substitute the above in the MSE equation, and rewrite our motivation problem:

$$\begin{aligned}&\mathbb{E}\left[\mathrm{MSE}\big(\mathcal{U}^i_B\big)\right]\\ &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\sum_{u\in\mathcal{U}}\left(\tilde{r}_{ui} - r_{ui}\right)^2\right]\\ &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(\left(\tilde{b}_i, \tilde{Q}_i\right) - \left(b_i, Q_i\right)\right) + \varepsilon_{\mathcal{U}}\right\|_2^2\right]\\ &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2}\varepsilon_B + \varepsilon_{\mathcal{U}}\right\|_2^2\right]\\ &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2}\varepsilon_B\right\|_2^2\right] + \frac{\mathbb{E}\left[\left\|\varepsilon_{\mathcal{U}}\right\|_2^2\right]}{|\mathcal{U}|}\\ &= \frac{1}{|\mathcal{U}|}\mathbb{E}\left[\left\|P^{\top}\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-2}\varepsilon_B\right\|_2^2\right] + \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\sigma_u^2\\ &= \frac{1}{|\mathcal{U}|}\left\|P^{\top}\left(P_B C_B^{-2} P_B^{\top}\right)^{-1} P_B C_B^{-1}\right\|_F^2 + \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\sigma_u^2\\ &= \mathrm{tr}\left(\left(P_B C_B^{-2} P_B^{\top}\right)^{-1}\right) + \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\sigma_u^2,\end{aligned}$$

where the equalities follow similarly to Lemma 1.