Predicting the Category of Customers' Next Product to Buy in Web Shops

Laura Rekasiute
Erasmus University Rotterdam
Rotterdam, the Netherlands
478746lr@student.eur.nl

Alvaro Jose Jimenez Palenzuela
Erasmus University Rotterdam
Rotterdam, the Netherlands
371623aj@student.eur.nl

Nijole Salnaite
Erasmus University Rotterdam
Rotterdam, the Netherlands
480399ns@student.eur.nl

Ramon Carrera Cuenca
Erasmus University Rotterdam
Rotterdam, the Netherlands
4478116rc@student.eur.nl

Flavius Frasincar
Erasmus University Rotterdam
Rotterdam, the Netherlands
frasincar@ese.eur.nl

ABSTRACT

Recommender systems are widely used by online retailers to entice customers into making new purchases. Understanding and predicting customer behavior is thus of utmost importance to retailers. In this paper our main goal is to predict the next product category that a certain customer will buy given his/her purchase history. We propose a Sequential Event Prediction model that captures both general and customer-specific consumption behavior through confidence rules. We use anonymized purchasing data from a Web shop in the Netherlands to show empirically that our approach outperforms several models proposed in the literature.

CCS CONCEPTS

• **Information systems → Recommender systems**; *Personalization*; Retrieval tasks and goals.

KEYWORDS

sequential event prediction, recommender systems, supervised ranking, product category, Web shop

ACM Reference Format:
Laura Rekasiute, Alvaro Jose Jimenez Palenzuela, Nijole Salnaite, Ramon Carrera Cuenca, and Flavius Frasincar. 2022. Predicting the Category of Customers' Next Product to Buy in Web Shops. In *The 37th ACM/SIGAPP Symposium on Applied Computing (SAC '22), April 25–29, 2022, Virtual Event, Czech Republic.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3477314.3506984

1 INTRODUCTION

With the accelerated growth of e-commerce and the massive amount of purchase data being collected, it is crucial that online retailers make a sensible use of this data to remain competitive. In fact, recommender systems that exploit this data are extensively used nowadays. Online retailers can increase their revenues mainly in two ways: by attracting new customers – which is said to be much more expensive than retaining an existing one [11] – or by selling more products to existing customers, which is known as cross-selling. For instance, most retailers periodically send personalized e-mails where they recommend certain products or product categories to their customers. If these recommendations are relevant to the customers, it is likely that they will visit their website and make a purchase, as shown in [2]. Therefore, it is of retailers' interest to be able to determine the most relevant product categories for each customer at each point in time.

In this paper, our main objective is to predict the next product category that a certain customer will buy given his/her purchase history in a Web shop (for our experiments we use a Web shop from the Netherlands, name is not disclosed due to privacy concerns). Unlike other approaches, we are only interested in predicting the product category and not the specific product. Note that the problem at hand also differs from the classical 'market basket prediction' problem which, e.g., supermarkets face. As opposed to the products that supermarkets sell, which are usually basic products that are bought recurrently, the products that our considered Web shop offers are durable goods, which are usually purchased only once. Furthermore, models for basket prediction usually rely on marketing instruments such as pricing and promotions [9].

There are several aspects we must take into account when constructing a suitable model that will allow us to make predictions. A customer's purchase history consists of many purchases that are ordered chronologically. Each of these purchases contains one or several products that were bought together. In our application the sequential order of the purchases is relevant, however, the order of the products inside a purchase is not. Furthermore, note that purchases are not equally spaced in time. The mathematical translation of this is that we need to model sequences (customer histories) of different lengths, each one consisting of itemsets (purchases) of different sizes.

In our research approach we frame the problem of predicting the next product category that a customer will buy in the context of Sequential Event Prediction (SEP) [15]. To this end, we build on the theoretical framework proposed by [7], which uses association rules to model sequences of events. The novelty of their work lies in that they establish a theoretical foundation for using association rules in supervised sequence learning. The method they propose computes partial probabilities (the confidence of these association

SAC '22, April 25–29, 2022, Virtual Event, Czech Republic

ACM ISBN 978-1-4503-8713-2/22/04…$15.00
https://doi.org/10.1145/3477314.3506984

rules) that are weighted according to optimized parameters in order to rank the items that are most likely to appear next in a sequence. As opposed to some *black-box* approaches whose results are difficult to interpret, by using these association rules marketeers are able to gain a better insight into the customers' consumption behaviors. We extend the work of Letham et al. by using additional parameters that allow us to better capture the heterogeneity in the consumption behavior across customers, leading to better predictive power. We apply this model individually, as personalization can yield more accurate predictions [3].

The structure of our paper is as follows. Section 2 reviews previous literature related to the problem at hand. Section 3 briefly describes and explores the data that is used to evaluate the model. The core of this paper – the proposed method – is presented in Section 4. Section 5 introduces the performance measures and discusses the results of the evaluation of our model. Lastly, Section 6 presents the conclusions and suggests some directions for future research.

2 LITERATURE REVIEW

A lot of research has been done in the field of recommendation algorithms over the last decade [5, 12, 14, 17]. These algorithms are best known for their use on e-commerce Web sites, where customers' interests are employed to generate personalized lists of recommended items [1]. In real-world scenarios, customers usually purchase series of baskets of items at different times. This recommendation task in e-commerce sites is formulated as the next basket recommendation [13]. In 2003 Linden et al. introduced the item-to-item collaborative filtering. This model was created and used by Amazon.com to personalize the online store for each customer. The algorithm attempts to find similar items to the ones purchased by the user and recommends the most popular or correlated items. To determine the most similar match of a given item, the algorithm uses cosine similarity to build a table of similar items by finding those items that customers tend to purchase together. This technique measures the cosine of the angle between two vectors, where each vector corresponds to an item rather than to a customer. Item-to-item collaborative filtering is an effective form in terms of creating a personalized shopping experience for each customer. According to the authors, the method is scalable over very large numbers of customers and product catalogs. Nonetheless, we argue that this method may have difficulties in considering sequential features of transaction histories, i.e., the algorithm does not take into account the sequential nature of the data. Moreover, the authors present this method as a solution for large numbers of items disconsidering product categories, while in our work we consider product categories.

In 2010, Rendle et al. proposed the Factorizing Personalized Markov Chains (FPMC) method, which is based on underlying Markov chains. For each customer an individual transition matrix is generated, which in total results in a transition cube which contains all the individual transition matrices. In this model sequential data and user-taste are captured by user-specific transition matrices. To deal with the sparsity of the data, a factorization model based on pairwise interactions is proposed. The factorization model allows each transition to be influenced by similar users, similar items, and similar transitions. The FPMC can capture sequential effects as well as the general interests of customers. However, this method can only model sequential behaviors between adjacent baskets. In addition, it utilizes a linear operation on multiple factors influencing customers' next purchase and it cannot depict the interactions among multiple factors, which we consider here.

In 2015, Wang et al. introduced the Hierarchical Representation Model (HRM) which explores sequential behavior (buying one item leads to buying another item) and the general taste of individuals by involving transactions and user representation in prediction. HRM represents each customer and item as a vector and is based on a two-layer structure, where the first layer aggregates item vectors from the last transaction to build the transaction representation, and the second layer aggregates the individual vector and the transaction representation to form the hybrid representation from the last transaction. This final hybrid representation is used to predict the items in the next basket. Furthermore, HRM allows different interactions among multiple factors of the input representation, such as average pooling (a linear operation) and max pooling (a nonlinear operation). Nevertheless, the main deficiency of this model is that it extracts local sequential features only between adjacent baskets, which we address in this work by taking in account baskets that are further apart from each other.

Also in 2015, Yu et al. introduced the Dynamic Recurrent Basket Model (DREAM), which is based on a Recurrent Neural Network (RNN). DREAM learns a dynamic representation of a user and captures global sequential features among overall historical transaction data. Yu et al. formalize the problem of predicting a ranked list of items for each buyer at a specific time moment. A dynamic representation with different baskets over time is made for each customer by pooling and matrix operations, and global sequential features are obtained by the recurrent structure. The paper shows that the employed nonlinear operations are effective in learning the representation of a basket and capturing complex interactions among multiple factors of items. Moreover, the authors show that DREAM outperforms the aforementioned FPMC and HRM models. However, since the method could be classified as a *black-box* approach, it is not well-suited for our case as we would like to have deeper insights into the model.

In 2011, Rudin et al. presented a theoretical framework for the problem of sequential event prediction, which aims to determine which event will be revealed next. This approach is common in the medicine field – e.g., [10] uses the Hierarchical Association Rule Model (HARM) based on ranking association rules to predict patients' medical conditions. The idea behind this method is to use the information from similar patients to add to the missing data on a particular patient's history. The algorithm proposed by Rudin et al. uses association rules, defined as an implication "$a \rightarrow b$" (meaning that itemset a co-occurs with item b), to find correlations and make predictions based on subsets of past events that occur at the same time. Two algorithms, employed as ranking models which are based on association rules, are presented: a *max confidence, min support* algorithm, and an *adjusted confidence* algorithm. The latter has an advantage over the first one: it allows rare rules to be used, and among rules with similar confidence, it prefers those with a larger support. However, this paper does not look over marketing

cases as proposed here, and does not discuss the possibility of including additional information such as users' heterogeneity, which we hypothesize could help improve the prediction accuracy.

Lately, in 2013, Letham et al. proposed in [7] association rules for sequential event prediction in a supervised ranking framework, which is based on the predictive power of sets of past events. In their approach they employ optimization-based algorithms based on the principle of empirical risk minimization (ERM). Each of the sequential event predictions is treated as a supervised ranking problem. To be able to find the partial probabilities, Letham et al. propose the so-called *max confidence* algorithm. The method is based on association rules "$a \rightarrow b$". Items b are scored and ranked in descending order to make the predictions. We believe that this approach suits well our problem of ranking the categories that a customer might purchase next. Therefore, we use this theoretical framework as the base for our model and extend it by personalizing the parameters for each customer and considering both individual and aggregate-level data.

3 DATA

The Web shop data consist of transaction information on a selection of 246,932 customers from a Web shop in the Netherlands. This dataset contains a random and anonymized set of purchases of customers made in a three year period from 1 January 2015 to 31 December 2017, with a total number of approximately 3.4 million orders. Although there are two levels of categorization (Products and Categories), we focus only on the latter (18 Categories). Due to privacy concerns the other summary statistics of the data are not made available.

We only take into consideration those customers who made more than one purchase, since a minimum of two purchases is required to train the model and to evaluate how well it performs. The subset of these contains 167,194 customers. The average number of purchases per customer in this subset is 11.2 and the average number of items per basket is 1.6. Last, note that in order to test the models we split each customer's purchase history into a training set, which contains all purchases except the last one, and a test set, which consists of the last purchase.

4 METHOD

In this section we first present the notation that we will employ in the rest of the section. Thereafter we present the base model, which corresponds to the model proposed for SEP by [7]. This base model is applied sequentially to all customers in order to obtain parameters that are common to all of them. As opposed to the SEP approach, we suggest to apply the method per customer and obtain individual-specific parameters. First, we present the *individual model*, which uses an individual confidence matrix; second, we introduce the *general model*, which instead uses a general confidence matrix; and last, we combine the intuition behind both models in the *mixed model*. We hypothesize that a model personalized to each customers' behavior would lead to more accurate predictions than a general model for all customers.

4.1 Notation

We first introduce the notation that we will use in presenting the models. For the sake of readability, in the following we refer to *customer purchases* instead of *events*, *customer histories* instead of *sequences*, and *categories* instead of *items*. However, note that this approach extends easily to other problems where one has multiple sequences of events. Following the notation in Letham et al., we define:

- m, the number of customer histories;
- $\mathbb{Z}$, the set of categories, of size N;
- T_i, the number of purchases of customer i;
- $z_{i,t}$, t-th purchase of customer i (category or set of categories bought at time t by customer i);
- $x_{i,t}$, all purchases of customer i up to and including time t ($x_{i,t} = \{z_{i,j}\}_{j=1,\ldots,t}$);
- $X_i = x_{i,T_i}$, all purchases of customer i – his/her full history; and,
- X^m, all purchases of all m customers.

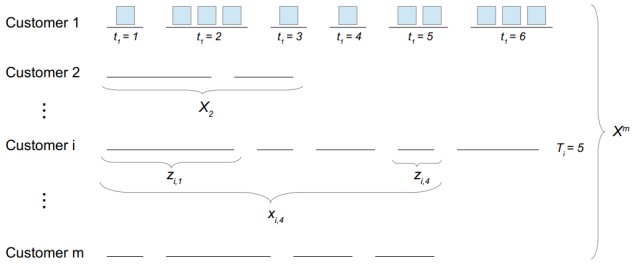


Figure 1: Notation scheme

Next, we introduce *association rules*, which are the building blocks of our model. Association rules establish dependency relationships between events. In our application, association rule "$a \rightarrow b$" denotes that buying category (categories) a implies buying category b thereafter. Association rules that establish dependencies between several categories and one category could allow us to capture more complex relationships in the data. For instance, "$\{m, n\} \rightarrow b$", where $\{m, n\} = a$ is a set of two categories and b is one category.

As opposed to Letham et al., who define the confidence of rule "$a \rightarrow b$" as the proportion of the sequences that have a and b (at any point in the sequence) given that a is present, we define the confidence of rule "$a \rightarrow b$" as the proportion of the customers who bought category a and also category b in the remaining part of the sequence after category a:

$$Conf(a \rightarrow b) = \hat{P}(b|a) = \frac{\#(b \text{ bought after } a)}{\#a}. \quad (1)$$

This allows us to take into account the order of the purchases as well as the dependency relationships between categories.

4.2 Base Model

Having defined the notation and the confidence rules, we now introduce the base model. This model employs a scoring function f

that, given a customer's history at t and parameters $\boldsymbol{\lambda}_\varnothing$, $\boldsymbol{\mu}$, scores a category b. The scoring function can be written as:

$$f(x_{i,t}, b; \boldsymbol{\lambda}_\varnothing, \boldsymbol{\mu}) = \lambda_{\varnothing,b} + \sum_{j=1}^{t} \sum_{a \subseteq z_{i,j}} \mu_a \hat{P}(b|a), \tag{2}$$

where parameter $\lambda_{\varnothing,b}$ gives a score for category b when no purchase history is known and generally represents the "baseline" score for b, and the term $\mu_a \hat{P}(b|a)$ gives a score that b will be bought later in the sequence a. Parameter μ_a can thus be regarded as a "correction" for the general confidence of rule "$a \rightarrow b$". The sum of these two terms is the score of category b at time t for customer i.

To obtain the optimal parameters $\boldsymbol{\lambda}_\varnothing$ and $\boldsymbol{\mu}$ we use an empirical loss function that averages the number of times that the predictions are incorrect. Although the "incorrectness" criterion varies across applications, in general one wants the loss function to be large if the predictions are incorrect and vice versa. For instance, consider a purchase at time t followed by a purchase {a, b, c, d, e} at $t+1$. When assigning the scores at t to each of the 18 categories in order to predict the categories that will be bought at $t+1$, we would like categories a, b, c, d and e to be scored strictly higher than the rest of the categories. To this end, we construct two sets of categories, $L_{i,t}$ and $K_{i,t}$, where $L_{i,t}$ is the set of (purchased) categories that should be ranked strictly higher than the (other) categories in set $K_{i,t}$. Following this notation, we consider a prediction to be incorrect when a category from set $K_{i,t}$ is scored higher than a category from set $L_{i,t}$. In our application we use the customer's next purchase, $z_{i,t+1}$, as $L_{i,t}$ and the remaining categories, $\mathbb{Z} \setminus z_{i,t+1}$, as $K_{i,t}$.

$$R_{0-1}(f, X^m; \boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} \mathbb{1}_{[f(x_{i,t},k;\boldsymbol{\theta}) \geqslant f(x_{i,t},l;\boldsymbol{\theta})]} \tag{3}$$

In order to smooth the loss function and for ease of derivation, we replace the indicator function with an exponent term based on the inequality $\mathbb{1}_{[b \geqslant a]} \leqslant e^{b-a}$ and a regularization term of the squared ℓ_2-norm of the parameters $\boldsymbol{\theta}$ (the vector consisting of $\boldsymbol{\lambda}_\varnothing$ and $\boldsymbol{\mu}$) that is outside of the summations. Having sums of exponential terms will allow us to derive analytically the gradients of these functions.

$$R_{exp}(f, X^m; \boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} e^{f(x_{i,t},k;\boldsymbol{\theta}) - f(x_{i,t},l;\boldsymbol{\theta})} + \beta ||\boldsymbol{\theta}||_2^2 \tag{4}$$

According to Letham et al. this model generalizes well. However, when predicting the next category that a customer will buy, this model's recommendations are too general; it fails to incorporate the heterogeneity in purchasing behavior across individuals. In Section 4.4 we discuss how we use this general model as the base for personalized model that we propose.

4.3 Optimization of the Base Model

The optimization of the loss function (4) is performed by stochastic gradient descent (SGD), an iterative maximization/minimization algorithm that stochastically approximates gradient descent (GD). SGD is computationally more advantageous than GD when optimizing loss functions that can be expressed as the sum of derivable individual loss functions, as is our case:

$$R_{exp}(f, X^m; \boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} R_{i,exp}(f, X_i; \boldsymbol{\theta}),$$

where $R_{i,exp}(f, X_i; \boldsymbol{\theta})$ is customer i's loss function.

When the number of training sequences m is large, the computation of the loss function $R_{exp}(f, X^m; \boldsymbol{\theta})$ can become long. SGD takes into account each customer sequentially by using a single training sequence at a time instead of all of them, thus reducing the computational time. The algorithm is as follows:

(1) Set initial values for the parameters;
(2) Shuffle the training data;
(3) Compute $\boldsymbol{\theta}_{i+1} := \boldsymbol{\theta}_i - \eta \nabla R_{i,exp}(f, X_i; \boldsymbol{\theta}_i)$, where $\nabla R_{i,exp}(f, X_i; \boldsymbol{\theta}_i)$ is the vector of first-order partial derivatives w.r.t. the parameter vector $\boldsymbol{\theta}$, η is the learning rate, for all customers i in the training data
(4) Repeat steps 2 to 3 until $| R_{exp}(f, X^m; \boldsymbol{\theta}^n) - R_{exp}(f, X^m; \boldsymbol{\theta}^{n+1}) | < \epsilon$, where n represents the nth iteration across all users and ϵ is a chosen threshold value.

Being able to derive the partial derivatives analytically instead of having to approximate them at each iteration is one of the main advantages of defining the loss function as the sum of exponential terms. The following are the partial derivatives required to perform stochastic gradient descent in the application of the base model:

$$\frac{\partial R_i(f, X_i; \boldsymbol{\theta})}{\partial \lambda_{\varnothing,j}} = \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} e^{f(x_{i,t},k;\boldsymbol{\theta}) - f(x_{i,t},l;\boldsymbol{\theta})} (\mathbb{1}_{[k=j]} - \mathbb{1}_{[l=j]}) + 2\beta\lambda_{\varnothing,j}, \tag{5}$$

$$\frac{\partial R_i(f, X_i; \boldsymbol{\theta})}{\partial \mu_j} = \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} \left(e^{f(x_{i,t},k;\boldsymbol{\theta}) - f(x_{i,t},l;\boldsymbol{\theta})} \sum_{p=1}^{t} \sum_{j \subseteq z_{i,p}} \hat{P}(k|j) - \hat{P}(l|j) \right) + 2\beta\mu_j. \tag{6}$$

4.4 Individually-Optimized SEP Model

We propose to tailor and extend the base model in order to account for the peculiarities of the problem at hand. In order to capture the heterogeneity in purchasing behaviour across customers, we

introduce individual-specific parameters in the model. In such way, each customer has a model fitted only for his purchase history, that is, the model is now optimized for each customer independently. The fact that this model can be applied independently to each customer makes the problem easy to parallelize, thus reducing the computational burden. Note that the definitions of sets $L_{i,t}$ and $K_{i,t}$ as well as the notation in Section 4.1 remain unchanged.

4.4.1 Individual Confidence Matrix. As a first approach to applying the SEP model individually (per customer), we only use the information of each individual to construct the corresponding individual confidence matrices. As in this case the training sequence consists of only one customer, it is necessary to introduce the individual confidence of a rule, $Conf_i(a \rightarrow b)$. Thus, for person i we define the confidence of rule "$a \rightarrow b$" as:

$$Conf_i(a \rightarrow b) = \hat{P}_i(b|a) = \frac{\#\text{ transitions from } a \text{ to } b}{\#\text{ all possible transitions from } a},$$

where the term *transition* refers to the occurrence of category b in the remaining part of the sequence after category a.

Having defined the new confidence measure, we now introduce the individual scoring function f_I with individual parameters and individual confidence rules:

$$f_I(x_{i,t}, b; \boldsymbol{\lambda}_{i,\varnothing}, \boldsymbol{\zeta}_i) = \lambda_{i,\varnothing,b} + \sum_{j=1}^{t} \sum_{a \subseteq z_{i,j}} \zeta_{i,a} \hat{P}_i(b|a). \tag{7}$$

The loss function to minimize is:

$$\begin{aligned} R_{exp,I}(f_I, X_i; \boldsymbol{\theta}_i) = & \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \\ & \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} e^{f_I(x_{i,t},k;\boldsymbol{\theta}_i) - f_I(x_{i,t},l;\boldsymbol{\theta}_i)} \\ & + \beta ||\boldsymbol{\theta}_i||_2^2, \end{aligned} \tag{8}$$

where $\boldsymbol{\theta}_i$ is a parameter vector consisting of $\boldsymbol{\lambda}_{i,\varnothing}$ and $\boldsymbol{\zeta}_i$.

Although this approach of creating an individual transition matrix for each customer increases the personalization of recommendations, its main drawback is the inability to recommend categories that were not bought previously. While this may not be a serious issue for customers with many purchases, many new customers will have too few purchases in order to construct a reliable individual confidence matrix.

4.4.2 General Confidence Matrix. To account for the general purchasing patterns at an aggregate level, the confidence rules $\hat{P}_G(b|a)$ are now calculated using all customers' purchasing histories. Thus, we rewrite the scoring function as:

$$f_G(x_{i,t}, b; \boldsymbol{\lambda}_{i,\varnothing}, \boldsymbol{\mu}_i) = \lambda_{i,\varnothing,b} + \sum_{j=1}^{t} \sum_{a \subseteq z_{i,j}} \mu_{i,a} \hat{P}_G(b|a), \tag{9}$$

where parameters $\lambda_{i,b,\varnothing}$ can be interpreted as those of the base model. $\mu_{i,a}$ can be interpreted as a parameter to "correct" the general trend of $\hat{P}(b|a)$ to the person i. The loss function for the individual model is constructed in the same way as in the base model, although the set of training sequences now consists of only one sequence – the individual's purchase history.

$$\begin{aligned} R_{exp,G}(f_G, X_i; \boldsymbol{\theta}_i) = & \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \\ & \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} e^{f_G(x_{i,t},k;\boldsymbol{\theta}_i) - f_G(x_{i,t},l;\boldsymbol{\theta}_i)} \\ & + \beta ||\boldsymbol{\theta}_i||_2^2, \end{aligned} \tag{10}$$

where $\boldsymbol{\theta}_i$ is a parameter vector consisting of $\boldsymbol{\lambda}_{i,\varnothing}$ and $\boldsymbol{\zeta}_i$.

In order to optimize this loss function we employ gradient descent as described in Section 4.5.

However, by calculating the confidence rules using only aggregate information of all the customers, some individual patterns may be masked. In order to account for this, we suggest a mixed model.

4.4.3 Mixed Confidence Matrix. A combination of both the individual and the general confidence matrices should incorporate the advantages of both previous approaches and allow the algorithm to recommend new categories for customers who have few purchases. A linear combination of both approaches results in the scoring function:

$$\begin{aligned} f_M(x_{i,t}, b; \boldsymbol{\lambda}_{i,\varnothing}, \boldsymbol{\mu}_i, \boldsymbol{\zeta}_i) = & \lambda_{i,\varnothing,b} \\ & + \sum_{j=1}^{t} \sum_{a \subseteq z_{i,j}} \Big(\mu_{i,a} \hat{P}_G(b|a) \\ & + \zeta_{i,a} \hat{P}_i(b|a) \Big). \end{aligned} \tag{11}$$

The loss function to minimize can now be written as:

$$\begin{aligned} R_{exp,M}(f_M, X_i; \boldsymbol{\theta}_i) = & \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \\ & \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} e^{f_M(x_{i,t},k;\boldsymbol{\theta}_i) - f_M(x_{i,t},l;\boldsymbol{\theta}_i)} \\ & + \beta ||\boldsymbol{\theta}_i||_2^2, \end{aligned} \tag{12}$$

where $\boldsymbol{\theta}_i$ is a parameter vector consisting of $\boldsymbol{\lambda}_{i,\varnothing}$, $\boldsymbol{\mu}_i$ and $\boldsymbol{\zeta}_i$.

4.5 Optimization of the Individual Model

To optimize the individually-applied SEP models with different confidence matrices we use gradient descent, an iterative minimization/maximization algorithm. This algorithm uses the partial derivatives of the loss function in question and iteratively updates the parameter values until an approximate minimum is reached. The algorithm for person i is as follows:

(1) Set initial values for the parameters;
(2) Compute $\boldsymbol{\theta}_{i,n+1} = \boldsymbol{\theta}_{i,n} - \eta \nabla R_{exp,\cdot}(f_\cdot, X_i; \boldsymbol{\theta}_{i,n})$, where η is learning rate and n is the iterator;
(3) Repeat step 2 until $| R_{exp,\cdot}(f_\cdot, X_i; \boldsymbol{\theta}_{i,n}) - R_{exp,\cdot}(f_\cdot, X_i; \boldsymbol{\theta}_{i,n+1}) | < \epsilon$.

The partial derivatives for this algorithm in our application are:

$$\frac{\partial R_{exp,\cdot}(f_\cdot, X_i; \boldsymbol{\theta}_i)}{\partial \lambda_{i,\varnothing,j}} = \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} e^{f_\cdot(x_{i,t},k;\boldsymbol{\theta}_i) - f_\cdot(x_{i,t},l;\boldsymbol{\theta}_i)} (\mathbb{1}_{[k=j]} - \mathbb{1}_{[l=j]}) + 2\beta\lambda_{i,\varnothing,j}, \tag{13}$$

$$\frac{\partial R_{exp,\cdot}(f_\cdot, X_i; \boldsymbol{\theta}_i)}{\partial \mu_{i,j}} = \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} \left(e^{f_\cdot(x_{i,t},k;\boldsymbol{\theta}_i) - f_\cdot(x_{i,t},l;\boldsymbol{\theta}_i)} \sum_{p=1}^{t} \sum_{j \subseteq z_{i,p}} \hat{P}_G(k|j) - \hat{P}_G(l|j) \right) + 2\beta\mu_{i,j}, \tag{14}$$

$$\frac{\partial R_{exp,\cdot}(f_\cdot, X_i; \boldsymbol{\theta}_i)}{\partial \zeta_{i,j}} = \sum_{t=0}^{T_i-1} \frac{1}{T_i} \frac{1}{|K_{i,t}|} \frac{1}{|L_{i,t}|} \sum_{l \in L_{i,t}} \sum_{k \in K_{i,t}} \left(e^{f_\cdot(x_{i,t},k;\boldsymbol{\theta}_i) - f_\cdot(x_{i,t},l;\boldsymbol{\theta}_i)} \sum_{p=1}^{t} \sum_{j \subseteq z_{i,p}} \hat{P}_i(k|j) - \hat{P}_i(l|j) \right) + 2\beta\zeta_{i,j}. \tag{15}$$

4.6 Baselines

In this paper we use the max confidence algorithm and the item-based collaborative filtering algorithm as baselines. The first algorithm uses confidence rules $Conf(a \rightarrow b) = \frac{\#(b \text{ and } a)}{\#a}$, where a is an itemset and b is an item in the sequence. The right-hand sides of the confidence rules, i.e., the potential future items b in the sequence, are ranked and a list is constructed with these ranked items by descending confidence. This ranked list is used to make predictions and its output gives the recommendations of particular items to the user. Another procedure that can be used for recommendations is the item-based collaborative filtering, which is based on cosine similarity. This method computes the similarities between items based on the ratings that people give to these items. The cosine similarity is intended for settings in which a user i applies a rating $R_{i,b}$ to item b. In our application, the rating reduces to $R_{i,b} = 1$ if sequence i contains item b and 0 otherwise. For each item b the binary vector of ratings $\mathbf{R_b} = [R_{1,b}, ..., R_{m,b}]$ is constructed and then the cosine similarity between every pair of items a and b can be expressed as

$$sim(a,b) = \frac{\mathbf{R_a} \cdot \mathbf{R_b}}{||\mathbf{R_a}||_2 ||\mathbf{R_b}||_2}.$$

For each item b, the k most similar items are found and they are defined as the neighborhood of b, Nbhd(b;k). In our case we used $k = 3$. In order to make a prediction from a partial sequence $x_{I,t}$, each item b is scored by adding the similarities of all of the observed items that occur both in the sequence and in the neighborhood Nbhd(b;k), and then normalizing it:

$$f_{sim}(x_{i,t}, b; k) := \frac{\sum_{a \in \bigcup_{j=1}^{t} z_{i,j} \cap Nbhd(b;k)} sim(a,b)}{\sum_{a \in Nbhd(b;k)} sim(a,b)}.$$

5 EVALUATION

Prior to the evaluation of the models, it is important to define how their goodness of fit will be assessed, i.e., the accuracy measure. When predicting which categories will be bought in the following purchase, we first rank them according to how likely it is that they will be bought next (we refer to categories by letters due to privacy concerns). After this, we check if any of the first 3 categories in the ranked list were actually bought in the following purchase. We refer to the proportion of times in which at least one of the predicted categories was bought next as Top-3 accuracies.

Before implementing the model that we propose in this paper, we try a few naïve approaches and regard their accuracies as benchmarks. The three most bought categories across all customers are *I*, *R*, and *H*. If we predict that customers will buy these categories in their next purchase, we obtain a Top-3 accuracy of 36.75%. Alternatively, if we predict that each customer will buy the categories which he/she bought the most before, we obtain a Top-3 accuracy of 46.65%, respectively. We thus observe that individual customers' purchase histories contain valuable information for predicting. The general procedure for the evaluation of each model is as follows. First, we construct association rules with each of the 18 product categories b by taking a as a previously visited category and we compute the confidence of these association rules. Then, each of the 18 product categories is scored and these scores are ranked to make the predictions. Last, we compute the Top-3 accuracies by checking whether the predicted categories match the ones that were actually bought.

In order to test the models we split each customer's purchase history into two parts: all purchases except the last one (as training data), and the last purchase (as test data) to verify whether our models perform well. We randomly select 20,000 customers to evaluate the models, which we find to be sufficient to make a comparison of their accuracies. In Appendix A.1 we plot the accuracies of the models for different sample sizes. We use Amazon's AWS cloud computing services to run some of the models and to perform the robustness analysis. Furthermore, we take advantage of the fact that the application of the individually-applied SEP models is an *embarrassingly parallel* problem.

We use a two-step approach to estimate the models: first, the confidence matrix (or matrices in the case of the individual-specific model) is constructed. For the base model, the confidence matrix is constructed with the training sequences of 5,000 customers for which the model is optimized. For the general and the individual models, the general confidence matrices are calculated using training sequences of 20,000 randomly chosen customers.

Secondly, the model is fitted by optimizing the parameters. A maximum of 10 previous baskets are used for optimization as more baskets do not increase the accuracy. For the base model, optimized on 5,000 customers, we use stochastic gradient descent with all

initial parameters 0.1, learning rate $\eta = 0.2$, and regularization parameter $\beta = 0.1$. On the other hand, for the individual-specific models the parameters are optimized through gradient descent. The initial parameters, the learning rate, and the regularization parameter remain the same across the performed experiments and we set $\epsilon = 10^{-5}$ as the convergence criterion.

5.1 Base Model

The confidence matrix for the base model reveals the general transition patterns between categories. By inspecting the heatmap in Figure 2 we can observe that the likelihood of buying a certain category again is higher than switching to other categories. It can also be observed that the transition to I is relatively very likely to happen from any other category. This is in line with the fact that I is the most bought category.

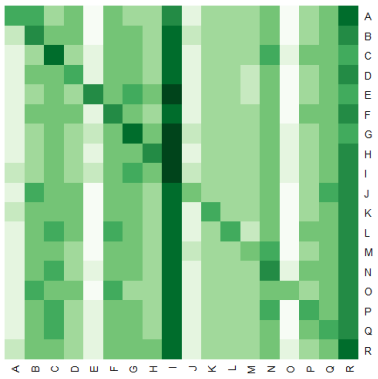


Figure 2: Confidence matrix (a in rows, b in columns)

To test the accuracy of the base model we ran the optimization algorithm on only 10 random samples of 5,000 customers each due to the time it takes to optimize this model (all customers and their associated sequences in the training data need to be visited). All samples were run with learning rate of 0.2 and $\beta = 0.1$. The model achieved a mean accuracy of 40% (Figure 3). However, we argue that due to the lengthy optimization process this algorithm is not suitable for large customer databases. Furthermore, we observe in Figure 4 that the algorithm never recommends certain categories such as D and E. Moreover, the four most popular categories, R, I, G, and C are recommended the most. We thus see that the base model does capture the general trend; however, it lacks personalization.

5.2 Individual Model

5.2.1 Individual Confidence Matrix. The model with an individual confidence matrix yields an average accuracy of 47.7% on 1,000 random samples each of 5,000 customers. The main drawback of this approach is the little amount of information available for those customers with few purchases, as can be observed in Figure 5. For those customers that have made many purchases, the number of distinct categories bought is larger, thus it is easier for the model to capture individual preferences and make better recommendations.

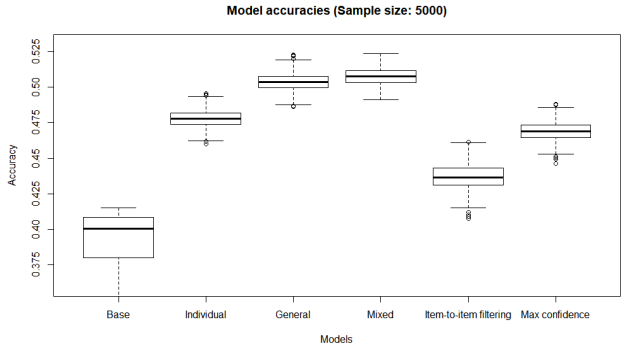


Figure 3: Accuracies of the models and the baselines (1,000 random samples, 10 random samples for the base model)

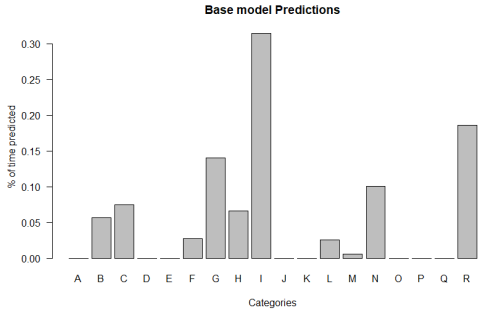


Figure 4: Categories predicted using the base model

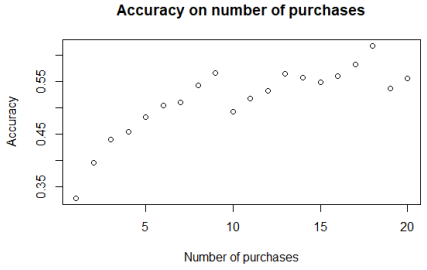


Figure 5: Accuracy for customers with different number of purchases

5.2.2 General Confidence Matrix. The individual-specific model using the general confidence matrix yields a Top-3 accuracy of 50.3%, calculated by taking 1,000 random samples of 5,000 customers. The accuracy of this model is thus better than that of the base model. By optimizing the parameters for each customer separately we are able to capture his/her buying patterns better while still referring to the general confidence matrix to account for the general trend. In Figure 6 we can see how the general confidence matrix looks for a randomly chosen customer. As the customer has never bought

some of the categories, some of the transitions have scores closer to zero (represented in the heatmap in white).

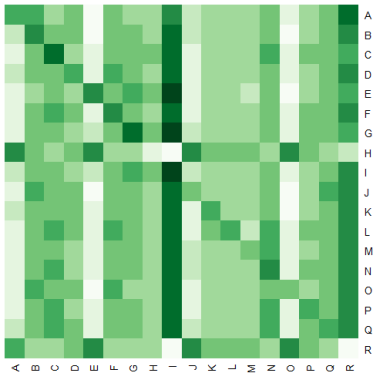


Figure 6: Matrix $\mu_{i,a}\hat{P}(b|a)$ of a randomly chosen customer i

5.2.3 Mixed Confidence Matrix. Finally, the combination of the two previous approaches –the *mixed* model– yields a 50.7% mean accuracy in predicting the next category on 1,000 samples of 5,000 customers. As expected, this method combines the advantages of both the individual and the general confidence matrices. Moreover, the recommended categories include more categories other than the most popular ones.

Figure 7 depicts the predicted categories for the three SEP models (individual, general, and mixed) for individuals. Differently than Figure 4 that reveals that only some of the categories are predicted by the base model, Figure 7 shows that all the categories are predicted by the SEP model, providing thus for a better coverage.

By comparing our models with the baseline algorithms from Section 4.6 we observe that the max-confidence algorithm yields an accuracy of 46.9%. That is 4 percentage points lower than that of the mixed model. The second algorithm, item-based collaborative filtering, achieves a mean accuracy of 43.7%, thus it is 7 points lower than the mixed model. Hence, the mixed model outperforms both baselines by incorporating both individual and general information into the model.

Table 1 displays the mean accuracies of the models as well as the baseline algorithms together with their standard errors. The accuracies were calculated by taking 1,000 samples of 5,000 customers for the individually optimized models and for the baseline algorithms. For the base model – which is optimized on all customers and is very computationally expensive – only 10 random samples of 5,000 customers were taken.

The statistical significance of the differences between the mean accuracies is tested by means of pairwise t-tests. The H_0 hypothesis of the difference between the means being zero is rejected if the p-value is smaller than 0.05. In Table 2 the p-values of such tests are presented. All the p-values are smaller than 0.05, which allows us to state that the mean accuracies of these models are significantly different.

Table 1: Mean accuracies of the models and baseline algorithms

	Mean (Std. error)	Std. dev.
Base	0.3894 (0.0091)	0.0287
Individual	0.4779 (0.0002)	0.0058
General	0.5034 (0.0002)	0.0059
Mixed	0.5073 (0.0002)	0.0056
Item-to-item filtering	0.4371 (0.0003)	0.0088
Max confidence	0.4691 (0.0002)	0.0063

Table 2: 10-sample two tailed paired t-test p-values (note that for the base model the test is unpaired)

	Base	Individual	General	Mixed	Item-based filtering	Max confidence
Base	-	< 0.001	< 0.001	0.001	0.035	< 0.001
Individual	< 0.001	-	< 0.001	< 0.001	0.001	0.015
General	< 0.001	< 0.001	-	0.006	< 0.001	< 0.001
Mixed	0.001	< 0.001	0.006	-	< 0.001	< 0.001
Item-based filtering	0.035	0.001	< 0.001	< 0.001	-	0.003
Max confidence	< 0.001	0.015	< 0.001	< 0.001	0.003	-

5.3 Robustness Analysis

To conclude the analysis of our model, we perform a robustness analysis of the mixed model by evaluating it for different initial parameters, learning rates, and betas.

First, we explore how the accuracies change when varying the initial parameters. Note that the mixed model has three types of parameters: $\boldsymbol{\lambda}_{i,\varnothing}$, $\boldsymbol{\mu}_i$ and $\boldsymbol{\zeta}_i$. Figure 8a shows a heatmap of the accuracies obtained by fixing $\boldsymbol{\lambda}_i = 0.2$ and varying $\boldsymbol{\mu}_i$ and $\boldsymbol{\zeta}_i$ from 0 to 0.4. Note that the subindices are omitted for the sake of readability. It is clearly observed that $\boldsymbol{\mu}$ and $\boldsymbol{\zeta}$ must be greater than 0 when choosing the initial parameters. The heatmap in Figure 8b shows the same as the previous one but leaves out $\boldsymbol{\mu} = 0$ and $\boldsymbol{\zeta} = 0$, making it easier to interpret the differences in accuracy when using initial parameters $\boldsymbol{\mu}$ and $\boldsymbol{\zeta}$ between 0.1 and 0.4. We now observe a clear pattern: the combinations under and on the diagonal yield the best accuracy, although the maximum difference is smaller than 2 percentage points. In conclusion, the initial parameters should be such that $\boldsymbol{\mu}, \boldsymbol{\zeta} > \mathbf{0}$ and $\boldsymbol{\mu} \leq \boldsymbol{\zeta}$ in order to achieve the best predictive accuracy. The heatmaps for $\boldsymbol{\lambda} = 0$ and $\boldsymbol{\lambda} = 0.1$, which are very similar to the one shown here, can be found in the Appendix A.2.

Following the analysis of the model for different initial parameters, we set different learning rates and beta values when evaluating the model and compare the respective accuracies in Figure 9. The differences in accuracy when changing the learning rate are relatively small, although we observe that choosing a lower learning rate is marginally better. We also observe that the mixed model's accuracy does not depend on the value of the β regularization parameter. Nevertheless, this parameter has been shown to be useful in certain cases to avoid overfitting and when the objective functions are ill-posed [4].

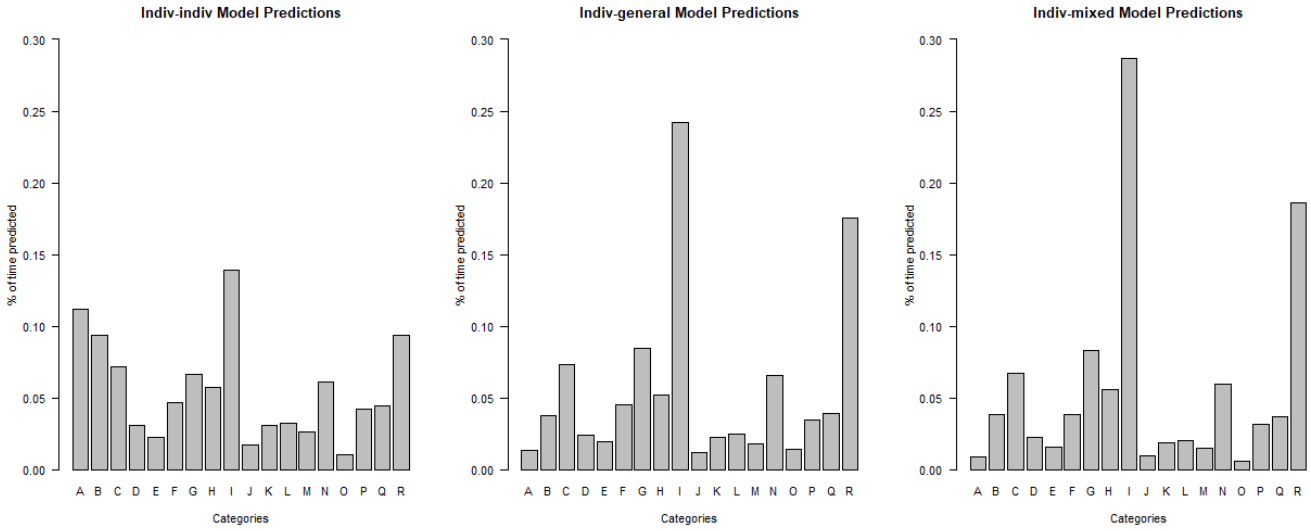
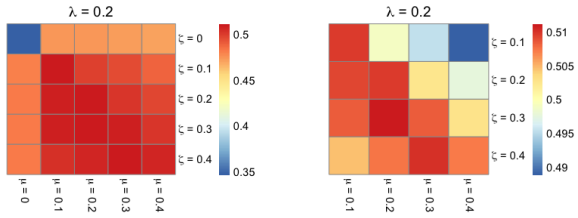


Figure 7: Predicted categories for the individually-applied SEP models



(a) μ, ζ from 0 to 0.4 **(b) μ, ζ from 0.1 to 0.4**

Figure 8: Top-3 accuracy heatmaps for $\lambda = 0.2$ and different μ and ζ

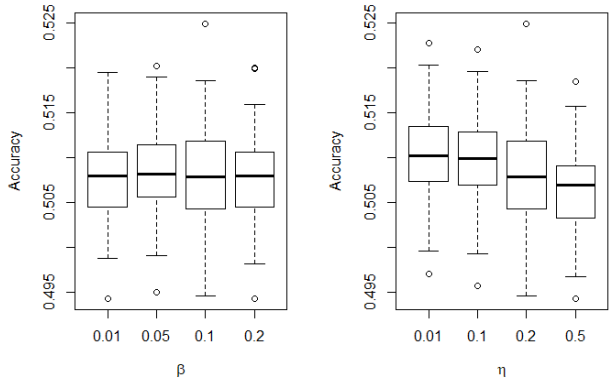


Figure 9: Accuracies of the mixed model for different regularization parameters β and learning rates η

6 CONCLUSION

In this paper we proposed a method to address the problem of sequential event prediction (SEP) in the context of predicting the next product category that a certain customer will buy in a Web shop. We constructed association rules that establish dependency relationships between categories and used them as the building blocks of our model. We showed that by applying the SEP model per customer and thus obtaining customer-specific parameters we can achieve a better accuracy than by having parameters that are common to all customers. The model with individual confidence matrices yielded an accuracy of 47.7%, the one with general confidence matrices, 50.3%, and the mixed model, 50.7%. Furthermore, we showed that our approach outperforms the max confidence and the item-based collaborative filtering models by 4 and 7 percentage points, respectively. By using both individual and general confidence matrices in the so-called *mixed model*, we were able to incorporate both individual and general consumption behavior in the model and achieved an accuracy of 50.7%.

Last, we suggest some directions for further research. For instance, association rules relating multiple items (on the left-hand side) could be used to capture complex relationships in the data, as described in Section 4.1. Our work could also be extended by constructing a *mixed-clustered* model where an individual confidence and a cluster confidence matrix are combined. This cluster confidence matrix would be computed by only taking into account those customers that are similar to each other. In this way, both the sequential and the demographic data of the customers can be incorporated in a model. As a preliminary study, we performed an early-stage exploration of such an approach by clustering only according to the most bought category and we found an increase in accuracy. Letham uses similarity-weighted confidence rules to a similar end [6]. Hence, a theoretical framework for combining clustering and modeling sequential events seems to be a promising continuation of our work.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17, 6 (2005), 734–749.

[2] Dan Ariely, John G Lynch Jr, and Manuel Aparicio IV. 2004. Learning by collaborative and individual-based recommendation agents. *Journal of Consumer Psychology* 14, 1-2 (2004), 81–95.

[3] Franz Böcker and Herbert Schweikl. 1988. Better preference prediction with individualized sets of relevant attributes. *International Journal of Research in Marketing* 5, 1 (1988), 15–24.

[4] Peter Bühlmann and Sara Van De Geer. 2011. *Statistics for high-dimensional data: methods, theory and applications.* Springer Science & Business Media.

[5] Dietmar Jannach, Paul Resnick, Alexander Tuzhilin, and Markus Zanker. 2016. Recommender systems—beyond matrix completion. *Commun. ACM* 59, 11 (2016), 94–102.

[6] Benjamin Letham. 2013. Similarity-weighted association rules for a name recommender system. In *ECML PKDD Discovery Challenge*. ceur-ws.org, 73–80.

[7] Benjamin Letham, Cynthia Rudin, and David Madigan. 2013. Sequential event prediction. *Machine Learning* 93, 2-3 (2013), 357–380.

[8] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing* 7, 1 (2003), 76–80.

[9] Puneet Manchanda, Asim Ansari, and Sunil Gupta. 1999. The "shopping basket": A model for multicategory purchase incidence decisions. *Marketing Science* 18, 2 (1999), 95–114.

[10] Tyler H McCormick, Cynthia Rudin, and David Madigan. 2012. Bayesian hierarchical rule modeling for predicting medical conditions. *The Annals of Applied Statistics* (2012), 652–668.

[11] Phillip E Pfeifer. 2005. The optimal ratio of acquisition and retention costs. *Journal of Targeting, Measurement and Analysis for Marketing* 13, 2 (2005), 179–188.

[12] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. 2018. Sequence-Aware Recommender Systems. *Comput. Surveys* 51, 4 (2018), 66:1–66:36.

[13] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW 2016)*. ACM, 811–820.

[14] Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). 2015. *Recommender Systems Handbook.* Springer.

[15] Cynthia Rudin, Benjamin Letham, Ansaf Salleb-Aouissi, Eugene Kogan, and David Madigan. 2011. Sequential event prediction with association rules. In *Proceedings of the 24th Annual Conference on Learning Theory (COLT 2011)*. JMLR.org, 615–634.

[16] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2015. Learning hierarchical representation model for nextbasket recommendation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2015)*. ACM, 403–412.

[17] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Defu Lian. 2022. A Survey on Session-based Recommender Systems. *ACM Comput. Surv.* 54, 7 (2022), 154:1–154:38.

[18] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2016)*. ACM, 729–732.

A ADDITIONAL PLOTS

The appendixes present accuracies of the individually-applied SEP models and the accuracy heatmaps for additional λ values.

A.1 Accuracy and Sample Sizes

Figure 10 shows the accuracies of the individually-applied (mixed, general and individual) SEP models, each for 2,000 samples of different sizes.

A.2 Top-3 Accuracy Heatmaps for $\lambda = 0, 0.1$

Figure 11 and Figure 12 show the top-3 accuracies for $\lambda = 0$ and $\lambda = 0.1$ when varying μ and ζ from 0 to 0.4 and zooming in the interesting ranges of these two last parameters.

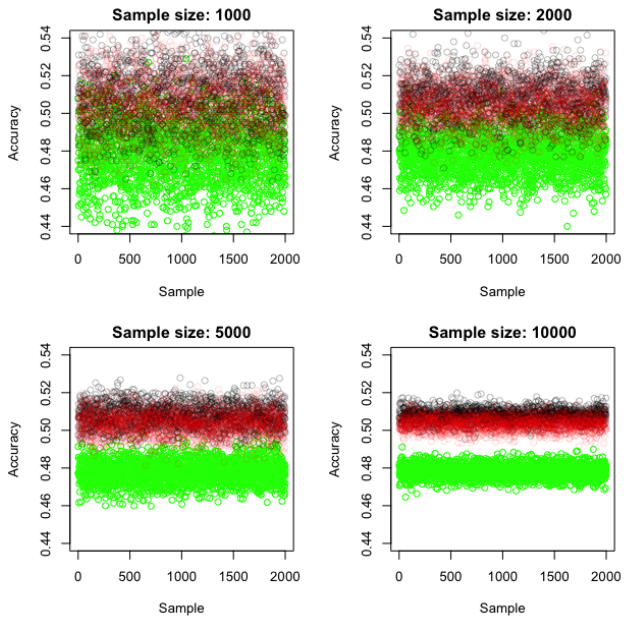
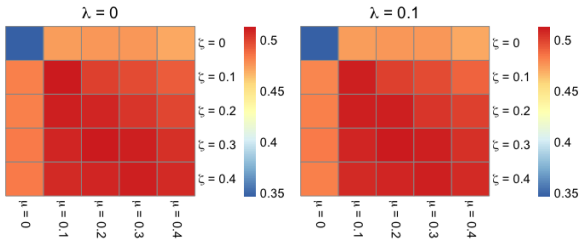
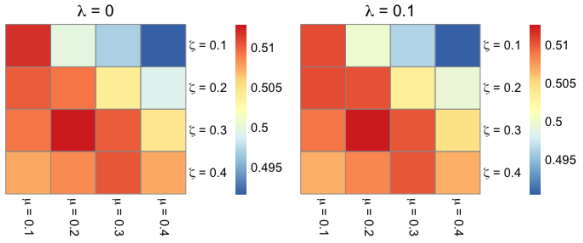


Figure 10: Accuracies of the individually-applied SEP models taking 2,000 random samples of different sizes (mixed model in black, general model in red [dark gray in black and white printing] and individual model in green [light gray in black and white printing])



(a) Heatmap: $\lambda = 0$ **(b) Heatmap: $\lambda = 0.1$**

Figure 11: Top-3 accuracy heatmaps for different λ and with μ, ζ from 0 to 0.4



(a) Heatmap: $\lambda = 0$ **(b) Heatmap: $\lambda = 0.1$**

Figure 12: Top-3 accuracy heatmaps for different λ and with μ, ζ from 0.1 to 0.4