Prediction of Sparse User-Item Consumption Rates with Zero-Inflated Poisson Regression

Moshe Lichman[*]
Department of Computer Science
University of California, Irvine
lichman@gmail.com

Padhraic Smyth
Department of Computer Science
University of California, Irvine
smyth@ics.uci.edu

ABSTRACT

In this paper we address the problem of building user models that can predict the rate at which individuals consume items from a finite set, including items they have consumed in the past and items that are new. This combination of repeat and new item consumption is common in applications such as listening to music, visiting web sites, and purchasing products. We use zero-inflated Poisson (ZIP) regression models as the basis for our modeling approach, leading to a general framework for modeling user-item consumption rates over time. We show that these models are more flexible in capturing user behavior than alternatives such as well-known latent factor models based on matrix factorization. We compare the performance of ZIP regression and latent factor models on three different data sets involving music, restaurant reviews, and social media. The ZIP regression models are systematically more accurate across all three data sets and across different prediction metrics.

KEYWORDS

Consumption Rate Modeling, Repeat Consumption, Explore-Exploit, Zero-Inflated Poisson

ACM Reference Format:
Moshe Lichman and Padhraic Smyth. 2018. Prediction of Sparse User-Item Consumption Rates with Zero-Inflated Poisson Regression. In *WWW 2018: The 2018 Web Conference, April 23–27, 2018, Lyon, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3178876.3186153

1 INTRODUCTION

In many aspects of our daily lives the way we consume products and items has evolved from interactions in a physical world to interactions in digital worlds. We purchase books online instead of shopping at brick-and-mortar stores, stream music and movies online instead of purchasing physical copies, and so on. The digital nature of our consumption provides the opportunity for tailoring of individual user experiences that can benefit both the consumer and the provider. As a consequence, the ability to develop predictive individual-level models for user-item consumption from past observations is increasingly important across a variety of applications.

Building accurate models of consumption in a typical digital environment is challenging for multiple reasons. In particular, as an individual moves forward through time, the items an individual consumes are a combination of (a) items that they have consumed in the past (i.e., repeat consumption), and (b) novel items that they have not consumed in the past (i.e., new consumption). User models in this context must balance these two aspects of behavior.

Individual heterogeneity, in the form of significant variability in behavior across users, further complicates the modeling process. In particular, when the set of possible items to be consumed is large, different users may have very different consumption patterns. Another significant challenge is data sparsity, given that the number of items a user typically consumes is often a very small fraction of the total number of available items.

In this paper we focus on the problem of predicting *rates* of item consumption per unit time (days, weeks, months) for individual users. The prediction of rates is broadly useful in a variety of applications since it allows us to predict not only which items a user will consume, but also how often those items will be consumed. For example, prediction of rates of consumption for specific items and specific sets of users is important in the design and engineering of proxy-caching systems for online streaming media content [19]. For contexts where items have different costs associated with them, predictions of the rates at which a user will consume specific items can be used for estimating the expected value of a customer from the provider perspective. Rates also can be used to help evaluate the expected benefit of interventions such as providing incentives to a user. For example, if some users have a high rate of usage for a particular app on their mobile phones and other users have low rates of usage for the same app, the latter group is likely to be a better target for incentivization than the former [8].

As mentioned above, in many real-world applications consumption behavior is characterized by a combination of repeat and new consumption. For example, some users' behaviors may be highly repetitive in nature, e.g., they tend to visit the same restaurants or listen to the same music artists, and rarely try new items. Other users may have behavior at the other extreme, continuously exploring new items and rarely returning to old items. This trade-off between exploration and exploitation is well known in computer science in the context of reinforcement learning, and is also well-established in cognitive science as a basic trait of how humans interact with the world around them (e.g., [4, 24]).

These observations suggest that in addition to handling significant heterogeneity in terms of individual behavior, the notion that there is a steady-state behavior for many users may be a fallacy in the sense that users are continuing over time to both exploit and explore the choice of items available to them. Rather than having user

[*] Current affiliation: Google Inc.

WWW 2018, April 23–27, 2018, Lyon, France

ACM ISBN 978-1-4503-5639-8/18/04.
https://doi.org/10.1145/3178876.3186153

models that are represented as fixed distributions over items, individual behavior can be thought as a dynamic process over time that is driven by feedback from past item consumption, both positive and negative. To capture these ideas we develop individual-level Poisson-based regression models where the predicted rate that a user will consume an item in the next time period is modeled as a function of an individual's past behavior. In addition, the models use global contextual information (such as item popularity) in order to better generalize to prediction of new items.

The primary contribution of this paper is the development of a systematic approach for modeling user-item consumption rates over time using Poisson-based regression models with zero-inflation. Through a systematic investigation of several user-item consumption data sets from multiple domains, we demonstrate that this modeling approach can capture individual-level user preferences for both old and new items as a function of past behavior and contextual information. We compare the proposed approach to state-of-the-art alternatives both empirically and qualitatively and also show that the proposed approach is scalable to large-scale data sets.

On the surface the problem we address looks very similar to that of the classic recommendation system problem. However, it is important to note that the modeling goals and evaluation criteria in our work are significantly different. Recommender systems focus only on prediction and ranking of new items that a user has not consumed in the past, e.g., for items such as movies or books, where typically an item is only consumed once by a user. In contrast we specifically focus on problems where consumption is a mix of repeated and novel item consumption. In this context a natural approach is to predict the *rates* at which items are consumed and to evaluate how well these rates are predicted, rather than just evaluating the likelihood of whether a user will consume an item or not.

The remainder of this paper proceeds as follows. In Section 2 we explore different user-item consumption sequence data sets, and provide motivation for our modeling approach in Section 3. In Section 4 we describe the proposed ZIP model for understanding and predicting user-item consumption rates and we show how this model is learned using user-item consumption observations in Section 5. Section 6 provides an overview of the existing approaches for modeling user-item consumption data. In Section 7 we compare our proposed model to a variety of state-of-the-art alternatives and interpret the results. Section 8 discusses the scalability of the approach and we conclude with a brief discussion in Section 9.

2 PROBLEM STATEMENT AND USER-ITEM CONSUMPTION DATA

In this paper we consider user-item consumption counts measured in discrete time intervals (by day, week, month, etc.). We define $y_{ij}^t \in \{0, 1, 2, \ldots\}$ as the number of consumptions of item $j \in \{1, 2, \ldots, M\}$ by user $i \in \{1, 2, \ldots, N\}$ in time window $t \in \{1, 2, \ldots, T\}$. In this context the goal of our work is to predict the expected number of items of type j that user i will consume during time $t + 1$, $E[y_{ij}^{t+1} | \ldots]$ given the history of all user-item consumption up through time t. It should be relatively straightforward to extend the approach to continuous time, where each consumption event has its own time-stamp—here we focus on the discrete-time case.

Dataset	N	M	t	T	% **non-zero**
reddit	1000	1000	week	52	2.5
lastfm	931	19997	month	50	0.5
Yelp	2836	203	2 months	12	1.7

Table 1: Summary of the three datasets used in this paper: number of unique users N, unique items M, time-window t, number of windows T, and the percentage of data points that are non-zero.

Our work is motivated by the challenge of creating a general framework for consumer behavior data across different domains. To that end, we investigate multiple publicly available data sets that represent different types of items and consumption activities. The 3 data sets are summarized and compared in Table 1.

Reddit: reddit is a popular social network with on the order of 1 million topic-focused subgroups (known as subreddits) where users can post, comment, and vote on content. In this work we considered data from a sample of $N = 1000$ users with high activity and $M = 1000$ highly active subreddits throughout 2015. The value of y_{ij}^t is defined as the number of times user i posted (or commented) in subreddit j during a given week t.

Lastfm: lastfm is an online music streaming service that allows users to listen to a selected song or playlist. The particular dataset we use contains the listening actions over time of nearly $N = 1000$ users[1]. We consider artists as items and retain the top $M = 20K$ artists that are most frequently listened to during the period of time of February 2005 to June 2009. In the lastfm dataset, y_{ij}^t represents the number of times that user i listened to a song performed by artist j during a month t.

Yelp: Yelp is a popular review platform that allows users to share their experience with different service providers such as restaurants. The dataset that we use has been widely used as a benchmark across recommendation system studies[2]. For our experiments we focused on the histories of $N = 2836$ unique users and their reviews of $M = 203$ types of restaurants (e.g. fast food, Mexican, sushi, etc.) in the Scottsdale and Phoenix (Arizona, USA) metropolitan areas between June 2014 and June 2016. y_{ij}^t is the number of times user i reviewed a restaurant of type j during every two months t.

3 EXCESS ZEROS AND HETEROGENEITY

Two typical characteristics of sparse user-item data sets are (1) an excess of zeros and (2) heterogeneity across both users and items. We discuss both characteristics below in turn and describe our approach to handling each from a modeling perspective.

3.1 Excess Zeros

A common feature of user-item consumption data sets, particularly when the number of items is large, is a very high rate of zeros, i.e., most users do not consume the vast majority of items. This is certainly true of the 3 data sets we analyze in this paper where roughly 98% to 99% of the entries are zero across the datasets. This

[1] http://www.dtic.upf.edu/~ocelma/MusicRecommendationDataset/lastfm-1K.html

[2] http://www.yelp.com/dataset_challenge

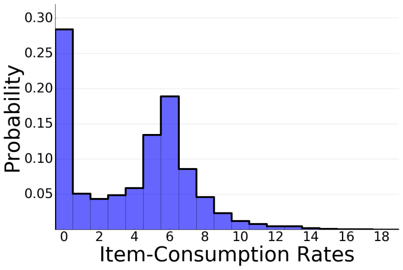

Figure 1: The distribution of item-consumption rates for a sample of user-item pairs with similar *average* rates.

is not surprising: for high-dimensional data sets each user will only be exposed to or be aware of a relatively small fraction of the potential items that they could interact with. In addition, there are practical limits (e.g., from cognitive and economic perspectives) in terms of how many items a user can realistically interact with.

In statistical modeling an overabundance of zeros is often referred to as "zero-inflation" [15, 20], to reflect the phenomenon that the frequency of zeros in the data is significantly higher than what a typical parametric model for count data (such as a Poisson model) can handle. This observation has been made in application contexts as diverse as epidemiology, economics, and manufacturing (e.g., see [2]), but has seen relatively little application to the type of high-dimensional user-item consumption data that we investigate in this pape—exceptions are [9, 16, 18], which we discuss in more detail later in the paper.

To illustrate the phenomenon of zero-inflation Figure 1 shows a histogram of the y_{ij}^t values for a sample of user-item pairs from all datasets with an average consumption rate between 5 and 6 across time (values of $y_{ij}^t = 0$ were excluded in computing the average). The histogram illustrates the variability of y_{ij}^t values across all time windows. We can see that the consumption rate has a bimodal distribution with one mode at $y_{ij}^t = 0$ and additional mode at $y_{ij}^t = 6$. We selected the average rate of 5 to 6 for illustration—similar bimodal patterns occur for different values of average number of user-item consumption. The bimodal nature of this data suggests that user-item rate can be represented as a mixture of two processes: an *exposure process* and a *rate process*.

Exposure Process: The exposure process describes whether or not a user i has been *exposed* to item j at time t. The concept of exposure captures the idea that for large item sets a typical user is likely to be unaware of (or unexposed to) most items in the "item vocabulary" (see also [9, 16, 18]), e.g., in music-listening many artists are unknown to many users. We define $z_{ij}^t \in \{0, 1\}$ as an indicator variable to indicate if user i was exposed to item j at time t. We can model $P(z_{ij}^t = 1)$ via a Bernoulli distribution with parameter π_{ij}^t, where the Bernoulli parameter will be a function of the past history of user i and item j.

Rate Process: Conditioned on exposure, i.e., $z_{ij}^t = 1$, the rate process accounts for the number of times user i consumes item j at time t. A natural and simple distribution for the rate process is the Poisson model, parameterized by the expected consumption rate λ_{ij}^t:

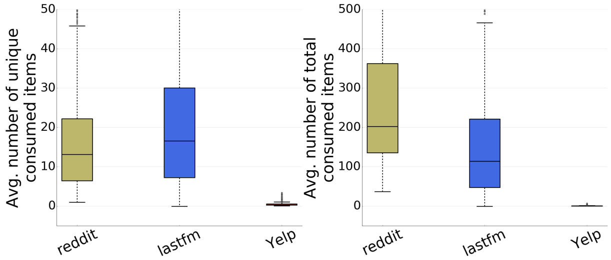

Figure 2: Average number of unique consumed items (left) and total number of consumed items (right) per week for each user across each of the 3 data sets.

$$P(y_{ij}^t = k|\lambda_{ij}^t) = \frac{{\lambda_{ij}^t}^k e^{-\lambda_{ij}^t}}{k!} \tag{1}$$

where $k = 0, 1, 2, \ldots$ is the number of consumptions. There are a number of other alternatives for defining probability distributions over count data that we could have used in our modeling approach, such as the non-negative Binomial distribution (NBD). We chose to use the Poisson distribution since it is straightforward to interpret the model parameters and is simple to implement. Using an NBD model, within the same general modeling framework that we propose here, could in principle lead to more accurate predictive models than the Poisson.

3.2 Data Heterogeneity

Another common feature of high-dimensional user-item data sets is heterogeneity. Figure 2 shows boxplots of the average number of unique items each user consumes (left panel) and the average number of total consumed items for each user (right panel), per week, for each of the 3 data sets. The figure clearly indicates (a) significant variability across users, as well as (b) significant variability across the different data sets. A plausible explanation for user variability is that different users can have significantly different *budgets*, either monetary or non-monetary (e.g. time), for consuming items. In addition there can be significant variation in the *domain-specific cost* for the consumption of a typical item across different data sets, leading to significant differences in the scale of user-item consumption across different domains, i.e., *domain-specific cost offsets*. For example, the effective cost to a user of listening to a song (lastfm) is significantly less than the cost of visiting a restaurant for a meal (Yelp).

Another contribution to data heterogeneity is the natural variation across users (and datasets) of some users to *explore* new items compared to their tendency to *exploit* known items. For example, a user who has a low tendency for exploration will naturally tend to repeat their behavior and the number and identity of unique items this user consumes is likely to remain relatively small and static over time. On the other hand a different user could have a tendency to be easily bored with items (a state that could be detected from recent activity [12]) and a corresponding tendency to often explore new items, where the new items are perhaps strongly influenced by global popularity and trends in the data.

4 POISSON REGRESSION WITH ZERO-INFLATION

Given the prevalence of zero-inflation and heterogeneity in user-item data sets we propose to model the observations y_{ij}^t for user i, item j, at time t, as

(1) a mixture of an exposure process and a rate process, where π_{ij}^t is the mixture weight and where λ_{ij}^t is the expected rate that user i will consume item j at time t (conditioned on being exposed to the item), and
(2) regression models for each of π_{ij}^t and λ_{ij}^t, conditioned on features $\boldsymbol{x}_{ij}^t$.

4.1 Zero-Inflated Poisson Models

The *exposure* and *rate* processes are modeled via a mixture of two components: (a) a delta function at zero and (b) a Poisson distribution. The mixture model weights and Poisson rate parameters, π_{ij}^t and λ_{ij}^t respectively, are user and item-dependent and are implicit functions of the features $\boldsymbol{x}_{ij}^t$—we provide more details on the conditional models for these parameters later in this section.

We can write the probability of $P_{zip}(y_{ij}^t = k|\pi_{ij}^t, \lambda_{ij}^t)$ as:

$$P_{zip}(y_{ij}^t = k) = \begin{cases} (1 - \pi_{ij}^t) + \pi_{ij}^t P_\lambda(k|\lambda_{ij}^t), & k = 0 \\ \pi_{ij}^t P_\lambda(k|\lambda_{ij}^t), & k = 1, 2, \ldots \end{cases} \tag{2}$$

where $P_\lambda(k|\lambda_{ij}^t)$ is the Poisson probability defined in Equation 1. The model above is known as the zero-inflated Poisson (ZIP) regression model, where the regression aspect arises through the conditioning of π_{ij}^t and λ_{ij}^t on the features [15]. In the ZIP model, zeros can be generated either by (a) the Bernoulli random variable π_{ij}^t taking value 0 or (b) π_{ij}^t taking value 1 and the Poisson model generating a value $k = 0$. From a generative perspective these two “routes" for generating zeros can be interpreted as either (a) the user i not being exposed to item j, or (b) the user being exposed but deciding not to consume the item (by drawing a zero from the Poisson distribution).

An alternative to the ZIP model is to use a shifted Poisson process for the rate that has a minimum value of $k = 1$ (rather than $k = 0$). In the statistical literature this is known as a *hurdle model* in the sense that the Poisson model is invoked if the count is greater than the “hurdle" (where here the hurdle value is 0). We empirically compared the hurdle and the ZIP model (results not shown) and found that the ZIP model systematically outperformed the hurdle variant for our 3 data sets in terms of modeling and predicting user-item consumption rates. For this reason we focus on the ZIP model in the rest of the paper.

4.2 Regression Modeling of Mixture Parameters

We model heterogeneity across users and items via generalized linear regression models for both π_{ij}^t and λ_{ij}^t, where the regression models depend on feature vectors $\boldsymbol{x}_{ij}^t$ that vary by user i, item j and time t. The regression models use two constant intercepts, globally-shared and individual-specific, capturing (respectively) the effect of *global domain costs* and heterogeneity in *individual-specific budgets*. In addition, we use four data-driven features, defined in Table 2, that are computed from each individual user's historical data and from contextual information.

Covariate	Notation	Value
Global domain costs	x_0	1
User-specific Budget	x_{i0}	1
Past user-item preference	$x_{ij}^{\bar{t}}$	$\log\left(1 + \frac{\sum_{\tau=1}^{t} y_{ij}^{\tau}}{t}\right)$
Current user-item activity	x_{ij}^t	$\log(1 + y_{ij}^t)$
Historical item popularity	$x_j^{\bar{t}}$	$\log(1 + \frac{\sum_i \sum_{\tau=1}^{t} y_{ij}^{\tau}}{tN})$
Current item popularity	x_j^t	$\log(1 + \frac{\sum_i y_{ij}^t}{N})$

Table 2: Definition of features used in our regression models, based on user and item historical data.

The features capture different aspects of user and item histories and allow the model to capture the balance between *explore* and *exploit* for each individual. **Past user-item preference**, $x_{ij}^{\bar{t}}$, represents the average rate that user i consumes item j over time and can capture the behavior of repetitive users who have a high probability of *exploitation.* **Current user-item activity**, x_{ij}^t, captures (on a log-scale) the recent activity of user i with item j, motivated by recent studies on the effect of recency and boredom in item consumption [1, 10–12]. **Historical item popularity**, $x_j^{\bar{t}}$, reflects the overall popularity of an item and is expected to capture the behavior of users whose *exploration* preferences are affected by conformity [23]. **Current item popularity**, x_j^t, captures current trends in item popularity, allowing the model to reflect the behavior of users driven by trends such as hype as a result of a sale, or the “death” of an item.

The use of features based on a user's past observations to predict the future behavior of the individual is an instance of an observation-driven time-series modeling approach (which we discuss further in the related-work section below). In particular, this allows for an individual's behavior to change over time in a non-stationary fashion. For example some individuals could be permanently in exploration mode to the extent that their future behavior is always different to their past (in terms of specific item consumption). More typical is the case where future behavior is a combination of repeat and novel item consumption, to varying degrees across different individuals.

The features used in this paper (in Table 2) are somewhat general and other features could be used depending on the application. For example, more specific domain-dependent features could also be incorporated, such as static features that provide side-information about users and items [18], or exogenous time-varying features such as seasonality or calendar effects [21].

Given the regression features we model the exposure and rate processes parameters in the following way:

Exposure Process: The value of π_{ij}^t is estimated using logistic regression, conditioned on the globally-shared and the individual-specific intercept coefficients η_0 and η_{i0} respectively, as well as the individual-based feature coefficient vector $\boldsymbol{\eta}_i = \{\eta_{i1}, \eta_{i2}, \eta_{i3}, \eta_{i4}\}$.

We denote the data-driven feature vector as $\boldsymbol{x}_{ij}^{t} = \{x_{ij}^{\bar{t}}, x_{ij}^{t}, x_{j}^{\bar{t}}, x_{j}^{t}\}$ and write the logistic function as:

$$\pi_{ij}^{t} = \frac{1}{1 + e^{-(\eta_0 x_0 + \eta_{i0} x_{i0} + \boldsymbol{\eta}_i \boldsymbol{x}_{ij}^{t})}} \tag{3}$$

Rate Process: Similarly, the value of λ_{ij}^{t} is modeled via Poisson regression with a globally-shared and individual-specific coefficient β_0 and β_{i0}, as well as an individual-specific coefficient vector $\boldsymbol{\beta}_i = \{\beta_{i1}, \beta_{i2}, \beta_{i3}, \beta_{i4}\}$. In addition, as proposed in [6] and [14], we added an additional intercept ($x_{j0} = 1$) with an item-specific offset β_{j0} to accommodate heterogeneity across items.

The resulting Poisson regression model can be written as

$$\log \lambda_{ij}^{t} = \beta_0 x_0 + \beta_{i0} x_{i0} + \beta_{j0} x_{j0} + \boldsymbol{\beta}_i \boldsymbol{x}_{ij}^{t} \tag{4}$$

where the feature vector $\boldsymbol{x}_{ij}^{t}$ is defined in the same way as in Equation 3.

5 LEARNING ALGORITHMS

Since the ZIP regression model can be expressed as a two-component mixture model, with $P_\lambda(y_{ij}^{t}|\lambda = 0)$ as the zero-inflation component, the model parameters can be estimated via a standard application of the Expectation-Maximization (EM) algorithm. EM is a general procedure for iterative optimization of a likelihood function with missing information. For mixture models the missing information for each data point y_{ij}^{t} is the identity of which component generated that data point. In particular, for ZIP mixtures this information is missing for all the zeros in the data set, $y_{ij}^{t} = 0$, since these data points could have been generated by either component. For values $y_{ij}^{t} > 0$ the data are unambiguously assigned to the rate component $P_\lambda(y_{ij}^{t}|\lambda_{ij}^{t})$.

The E-step computes the membership probability (equivalent to the expected value of the binary membership indicator) for each data point $y_{ij}^{t} = 0$, conditioned on current estimates of the model parameters. The M-step generates maximum likelihood estimates of the parameters conditioned on the membership probabilities provided by the E-step. Under fairly broad conditions, repeated application of E and M steps is guaranteed to converge to a (local) maximum of the likelihood function.

E-step: In the E-step, for each of the zero-valued data points $y_{ij}^{t} = 0$, we compute the membership probability w_{ij}^{t}, namely the probability that this zero was generated by the rate component. These membership probabilities can be computed by applying Bayes rule to the definition of the mixture model above, $P_{zip}(y_{ij}^{t} = k|\pi_{ij}^{t}, \lambda_{ij}^{t})$, where the parameters $\pi_{ij}^{t}, \lambda_{ij}^{t}$ are the current parameter estimates (from the most recent M-step or their initial values at the first iteration).

$$w_{ij}^{t} = \frac{\pi_{ij}^{t} P_\lambda(y_{ij}^{t}|\lambda_{ij}^{t})}{(1 - \pi_{ij}^{t}) P_\lambda(y_{ij}^{t}|\lambda = 0) + \pi_{ij}^{t} P_\lambda(y_{ij}^{t}|\lambda_{ij}^{t})} \tag{5}$$

Data points with membership weights closer to 1 are more likely (according to the current parameters) to have been generated by the rate component and, conversely, data with weights closer to 0 are more likely to have been generated by the zero-inflated component.

M-step: The M-step optimizes the parameters of the model conditioned on the current estimates of the w_{ij}^{t} membership values. Our ZIP model has two sets of parameters, the logistic regression parameters for the mixture weights $\boldsymbol{\eta} = \{\eta_0, \boldsymbol{\eta}_i\}$, and the rate parameters for the Poisson rate component in the mixture model, $\boldsymbol{\beta} = \{\beta_0, \boldsymbol{\beta}_j, \boldsymbol{\beta}_i\}$. The logistic regression uses the membership weights as targets and the Poisson regression uses weighted regression with the weights being the membership weights.

Neither the logistic or Poisson regression can be performed in closed-form, so we use gradient descent within each M-step to estimate the coefficients for each model. The gradients in both cases (logistic and Poisson) involve dense sums over all $N \times M \times T$ data values, where N, M and T are the number of users, items and time-windows respectively. This is in contrast to sparse estimation methods such as Poisson matrix factorization that can ignore the zeros in the data, effectively working with only a tiny fraction of the full data matrix for highly sparse data. Thus, in order to achieve a scalable algorithm, we use *stochastic gradient descent* (SGD) instead of full gradient methods, inspired by the success of SGD in training of large-scale deep neural networks on large data sets. SGD approximates the exact gradient at each gradient update by estimating the gradient in a stochastic manner using a small randomly-selected subset of rows (“mini-batches") from the data matrix. We discuss the convergence of our EM + SGD method in more detail in Section 8 later in the paper—at this point it is sufficient to note that our implementation is as fast (or faster) in wall-clock time when compared to publicly-available implementations of other competing approaches.

The step size in each SGD step was determined via the ADAM algorithm [13] which provides a systematic way of conditioning the step size on the level of confidence in the gradient. We found empirically that ADAM worked well for our SGD-based optimization problems (and that convergence could be difficult to attain without it) in agreement with work in deep learning where the combination of SGD and adaptive step-size (such as ADAM) is essential to the success of training models on large data sets.

One final note is that rather than maximizing the likelihood we maximized the likelihood times a prior, i.e., maximum a posteriori EM estimation. In log-space this corresponds to maximizing (in the M-step) the log-likelihood plus a regularization term corresponding to the log prior. In our experiments we found that empirically-determined MAP priors were particularly effective. To compute the empirical prior we trained the model using global coefficients (assuming all data belong to a single user) with L2 regularization. The learned coefficients were then used as a common prior for all users.

6 RELATED WORK

The conceptual basis of our work builds from a rich literature in statistics on modeling of count data [2]. For example, within the framework of generalized linear models the Poisson mean is modeled as $\exp(\sum \beta_k x_k)$ where the β_k's are regression coefficients and the x_k's are the inputs to the model. In the context of longitudinal data (data across multiple individuals) it is common to use fixed and random effects to account for individual-level heterogeneity, e.g., by allowing for individual-specific intercept terms in the mean such as $\exp(\beta_i + \sum \beta_k x_k)$ where β_i is the offset for individual i (e.g., [5], ch.7). The incorporation of time-dependence into such models can typically be categorized into one of two general categories ([2], ch 7.2): *observation-driven* models where the counts are modeled directly as functions of past counts (such as autoregressive models for count data), or *parameter-driven* models where the counts depend on a latent state-space process (such as a hidden Markov model or a linear-Gaussian filter). The models we propose in this paper are in the *observation-driven* category, while the dynamic matrix factorization methods (discussed below) that we compare to in our experiments are in the *parameter-driven* category. The use of zero-inflation is also well-known in statistical modeling of count data (e.g., [7, 15]) and can be combined with other modeling components (as we do in our proposed approach) such as temporal dependence and fixed effects.

While our approach builds on much of the above prior work in statistics, a significant difference is that we model *high-dimensional count vectors* (i.e., a large number of items). These count vectors are orders of magnitude larger in dimensionality than the low-dimensional (often scalar) count data that is often the primary focus in the statistical literature for modeling of count time-series [2]. To handle the optimization challenges of parameter estimation for high-dimensional counts we use techniques from stochastic gradient optimization, which have not (to date at least) seen much application in the statistical literature for count modeling.

Another significant line of related work is in matrix factorization of user-item consumption data. The most well-known approach in this context over the past decade has been the bilinear Gaussian model based on an SVD decomposition (e.g., [14]). The expected target value y_{ij} is usually represented in such models as

$$E[y_{ij}] = \theta_i'\phi_j + \beta_0 + \beta_i + \beta_j, \tag{6}$$

where $\theta_i'\phi_j$ is the inner product of low-dimensional latent vector representations for user u and item i, and β_0, β_i, and β_j are constant, user, and item offsets respectively. In this framework the latent vectors θ_i and ϕ_j and parameters $\beta_0, \beta_i, \beta_j$ are typically estimated from the data using least-squares. This is equivalent to maximizing the likelihood of a Gaussian model for the y_{ij}'s (e.g., [22]). This is a useful approach for data that can be approximated by a symmetric distribution but is not ideal for the types of highly skewed count data we are focusing on in this paper.

More recent work in matrix factorization has built on ideas from non-negative matrix factorization to develop models that are more appropriate for count data, e.g., where the expectation in Equation 6 above represents the mean of a Poisson model for the y_{ij}'s—known as Poisson matrix factorization (PMF). A typical approach is to estimate the parameters within a Bayesian framework (such as variational inference) and to place priors (such as Gamma priors) on the parameters θ_i and ϕ_j [6, 17].

Of particular relevance to this paper is the recently-introduced dynamic Poisson matrix factorization model (DPMF) [3] which models the expected counts as a function of time t as:

$$E[y_{ij}^t] = \theta_{it}'\phi_{jt} + \ldots$$

where t is a discrete time index (such as days, weeks, etc). Here the latent user and item vectors are allowed to evolve dynamically over time, such that predictions for time $t+1$ are a functions of the latent vectors estimated at time t. This DPMF approach (and matrix factorization in general) can be viewed as an instantiation of a *parameter-driven* latent-space model, in contrast to the *observation-driven* model that we pursue here.

Another recent strand of related work (in the non-dynamic PMF context) is the use of zero-inflated models in probabilistic matrix factorization. Liang et al. [16] proposed the framework of *exposure matrix factorization* (ExpoMF) which uses zero-inflation to explicitly account for exposure effects in matrix factorization of large binary user-item data sets. Liang et al. found that ExpoMF systematically outperformed traditional PMF methods that did not account for exposure. In a similar vein, Jain et al. [9] developed a probabilistic matrix factorization framework with zero-inflation to handle exposure effects, for multi-label classification with very large numbers of labels, also finding that explicit modeling of exposure systematically outperforms methods that do not include it. Finally, Liu and Blei [18] recently proposed a zero-inflated exponential family embedding approach for sparse binary and count matrices, which from the perspective of this paper could be effectively viewed as a “cousin" of traditional matrix factorization with its low-dimensional embedding representation of the data.

Our work differs from the matrix factorization and embedding approaches described above, in terms of our focus on (a) prediction of *consumption rates* rather than ratings or binary data, (b) modeling both repeat and novel consumption over time, and (c) the use of user- and item-specific regression models rather than low-dimensional factorizations or embeddings.

There has also been recent work on *continuous-time modeling* of time-stamped user-item data, using Markov approaches [1, 11, 12], Poisson point processes [8], and neural networks [10]. While these papers share a common motivation with our work in terms of analyzing explore/exploit aspects of user consumption, the focus and methodologies are significantly different to what we pursue in this paper, with less emphasis on user-item rate prediction and without the use of zero-inflation or regression models.

7 EXPERIMENTS AND RESULTS

Below we describe the results of comparing the ZIP model to baselines and to a number of well-known approaches from the literature for modeling sparse user-item count data. For all of the experiments described below the model parameters were estimated using data up to time $t-2$, with hyperparameter tuning via grid search using data at time $t-1$, and then evaluated on holdout test data from time t. This was repeated for $t = T-4$ to $t = T$ and the prediction metrics for the 5 test sets were then averaged.

7.1 Performance Metrics

We evaluated our models using four different metrics.

Log-Loss: The log-loss is the average of the negative log-probability (or negative log-likelihood) of each user-item consumption rate in the test data:

$$-\log P = -\frac{1}{N_{\text{test}}}\sum_i\sum_j \log P(y_{ij}^t)$$

where $P(y^t_{ij})$ is the probability of the observed count y^t_{ij}, under the model being evaluated, and where N_{test} is the total number of test points. The log-loss metric is bounded below by zero (attainable only by perfect predictions) and is widely used in the evaluation of machine learning algorithms that produce probabilistic predictions. A model that assigns higher probability, or lower negative log-probability, to the observed test data is preferred over a model that assigns lower probability (or a higher negative log-probability).

Precision, Recall, and F1: Let $\hat{y}^t_{ij}$ denote the expected number of times (according to a particular model) that user i will consume item j during time-window t. For the ZIP model, by the linearity of expectation we have that $\hat{y}^t_{ij} = \hat{\pi}^t_{ij}\hat{\lambda}^t_{ij}$ where $\hat{\pi}^t_{ij}$ and $\hat{\lambda}^t_{ij}$ are point (MAP) parameter estimates learned by the model on the training data.

For each pair i, j we can compute the precision and recall of a prediction $\hat{y}^t_{ij}$, relative to the observed value y^t_{ij}, as follows. Precision can be defined in the context of count data as $\frac{\min\{y^t_{ij}, \hat{y}^t_{ij}\}}{\hat{y}^t_{ij}}$, i.e., it is the fraction of user-item consumptions that the model predicted would occur that actually did occur. Similarly, recall can be defined as $\frac{\min\{y^t_{ij}, \hat{y}^t_{ij}\}}{y^t_{ij}}$, which is the fraction of observed user-item consumptions that did occur that the model predicted would occur. These pairwise user-item precision and recall values can be averaged over all user-item pairs to obtain overall precision and recall numbers. Models that systematically underestimate y^t_{ij} (e.g., that predict all zeros) will have high precision but low recall, and vice-versa for models that systematically overestimate y^t_{ij}. For our experimental results below we report the *F1* score, which combines both precision and recall, in the standard fashion as:

$$F1 = 2 \times \frac{\text{Prec} \times \text{Rec}}{\text{Prec} + \text{Rec}}$$

MAE: Mean absolute error between the expected number of times that a user i will consume item j during time-window t and the observed value y^T_{ij}: MAE = $\frac{1}{N_{\text{test}}} \sum_i \sum_j \left| y^t_{ij} - \hat{y}^t_{ij} \right|$

7.2 Poisson Regression with and without Zero-Inflation

We first compare the ZIP Poisson regression model (**ZIP**) to a Poisson regression model (**PR**) without a zero-inflation component. We use the same features for both models. Table 3 shows that the ZIP model systematically outperforms the PR model on holdout data, for all three metrics across all three data sets.

To further analyze the contribution of the zero-inflated component we evaluated each of the models in terms of their ability to predict the zeros. We focused on user-item pairs with $y^t_{ij} = 0$ and computed the Log-Loss for those pairs under each model. The Log-Loss values for the PR model are 0.064, 0.035 and 0.042 in the reddit, lastfm and Yelp data sets respectively. The corresponding values for the ZIP model are an order of magnitude lower: 0.004, 0.004 and 0.017. This significant improvement is directly attributable to the presence of the zero-inflation component in the ZIP model.

	Log-Loss		*F1*		*MAE*	
Dataset	PR	ZIP	PR	ZIP	PR	ZIP
reddit	0.30	**0.14**	0.70	**0.82**	0.40	**0.23**
lastfm	0.20	**0.08**	0.08	**0.25**	0.08	**0.04**
Yelp	0.09	**0.07**	0.09	**0.15**	0.07	**0.04**

Table 3: Log-Loss, F1 measure and MAE on the test data for the PR and ZIP models across different data sets. Lower values are better for Log-Loss and MAE and higher values are better for F1. Best performing methods indicated in bold font.

7.3 Comparing ZIP to Baselines and Matrix Factorization

Below we describe results obtained from comparing the ZIP model to a set of simple baselines and to several well-known approaches in the literature based on matrix factorization and embeddings for count data.

GR (Global Rate): This is defined as the global rate at which each item is consumed in the training data, computed by averaging across all users and time-stamps:

$$\hat{\lambda}^{GR}_{ij} = \hat{\lambda}_j = \frac{1}{N}\frac{1}{t}\sum_{i=1}^{N}\sum_{\tau=1}^{t} y^\tau_{ij} \qquad 1 \leq j \leq M$$

MPE (Mean Posterior Estimate): This is the mean posterior estimate (MPE) of each user-item rate, with a conjugate Bayesian $Gamma(\gamma_0, \gamma_1)$ prior, based on the counts in the training data:

$$\hat{\lambda}^{MPE}_{ij} = \frac{\sum_{\tau=1}^{t} y^\tau_{ij} + \gamma_0}{t + \gamma_1}$$

where γ_0 and γ_1 are determined via grid search to optimize the log-loss of the validation data for the MPE model.

PMF (Poisson Matrix Factorization): This is a latent factor matrix factorization model with a Poisson distribution for the observed counts. In our results we used a state-of-the-art Bayesian PMF version implementation by Liang et al. [17]. We fit the model to the aggregated data across all time windows. At prediction time the predicted rates from the model were divided by the number of time windows in the training data set to scale the rate for prediction for a single time window.

DPMF (Dynamic Poisson Matrix Factorization): This is an extension of the PMF model that learns latent-space decomposition from a sequence of user-item consumption counts [3]. The latent-space vectors for users and items are estimated for each time-window jointly by modeling the change in θ^t_i and ϕ^t_j between different time steps t using a Kalman filter.

ZIE (Zero-Inflated Exponential Family Embeddings): This is an exponential family embedding algorithm that uses a zero-inflation component to model exposure and effectively downweight zero counts when learning item embeddings from sparse binary or count

	GR	MPE	PMF	DPMF	ZIE	ZIP
reddit	2.984	0.248	0.362	0.845	4.214	**0.136**
lastfm	0.213	0.132	0.145	0.154	0.171	**0.075**
Yelp	0.078	0.084	0.076	0.076	0.074	**0.069**

Table 4: Log-Loss on the test data for different algorithms across different data sets. Lower scores are better. Best-performing methods indicated in bold font.

	GR	MPE	PMF	DPMF	ZIE	ZIP
reddit	0.07	0.62	0.63	0.57	0.01	**0.82**
lastfm	0.04	0.21	0.12	0.18	0.02	**0.25**
Yelp	0.10	0.11	0.13	0.10	0.11	**0.15**

Table 5: F1-scores on the test data for different algorithms across different data sets. Higher scores are better. Best-performing methods indicated in bold font.

	GR	MPE	PMF	DPMF	ZIE	ZIP
reddit	0.996	0.401	0.342	0.473	0.661	**0.228**
lastfm	0.057	0.049	0.051	0.043	0.136	**0.038**
Yelp	0.042	**0.035**	0.040	0.049	0.041	**0.035**

Table 6: MAE on the test data for different algorithms across different data sets. Lower scores are better. Best-performing methods indicated in bold font.

data [18]. We fit the model to the aggregated data across all time windows using Poisson distribution and scaled to a single time window at prediction time. The exposure covariates in this model are individual-specific and represent external information (such as demographic variables). In the absence of additional meta-data about the individuals for the data sets used in this paper, we used a single intercept for each item.

Hyperparameters for PMF, DPMF, and ZIE were determined via grid search on the validation data. One hyperparameter of particular interest is the number of factors or dimensions used by these models. We found that the matrix factorization techniques, PMF and DPMF, had the best predictive performance when using extremely high numbers of factors, to the point of almost memorizing the data. Rather than using very high dimensional representations, in keeping with typical matrix factorization experiments in the literature, we limited the number of factors for the models to a moderate range of 200 to 500 dimensions. For the ZIE method, grid search on the validation resulted in 50 to 100 embedding dimensions being approximately optimal for prediction across the different data sets, with little to no improvement above 100.

Tables 4, 5 and 6 show the *Log-Loss*, *F1*, and *MAE* scores on the test data, for each of the baselines and PMF, DPMF and ZIE models, compared to the ZIP model. The ZIP model is significantly more accurate than the other methods for all metrics for all data sets, except for the MAE score for the MPE model on the Yelp data set.

For both the reddit and lastfm data sets the margins of improvement of the ZIP model over the PMF, DPMF and ZIE models are quite large. There are two likely reasons for this improvement. The first is that the zero-inflation component in the ZIP model provides a more flexible way to handle excess zeros than PMF or DPMF. The second reason is that the the user-specific features in the regression approach (such as the history variable of what specific items a user consumed in the past) allows the regression model to more accurately model individual-level details than the matrix factorization (MF) or embedding approaches. These approaches are constrained by the dimensionality of their latent spaces, limiting the level of detail (e.g., specific combinations of items) available for modeling individual users. We explore both of these in more detail below.

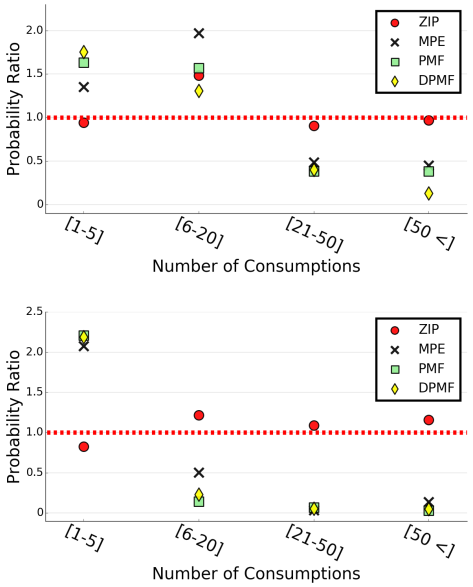

Figure 3: The ratio of probability for the number of consumptions for selected users from reddit (top) and lastfm (bottom) at time T assigned by the evaluated models compared to the ground truth (GT). The number of consumptions predicted by the ZIE model were all lower than 1. As a result the ratios of probabilities for the ZIE model are omitted from the plot for clarity. Best viewed in color.

Modeling excess zeros: Modeling the zeros provides the ZIP model with a principled way of down-weighting the zeros in the process of learning the rate parameter. As a result the rate-process coefficients are free to fit a larger range of y_{ij}^t values (in particular, high numbers of consumptions). In Figure 3 we plot the ratio between (a) the predicted number of $\hat{y}_{ij}^t$ counts, across different ranges of y, for the different models, and (b) the ground truth number for those values in the test data. These plots are for users with high variance in their y_{ij}^t values, for both the reddit (top) and lastfm (bottom) data sets. We can see that in order to fit the excess of zeros, the baselines tend to systematically and significantly overestimate the low rates and to underestimate the high rates, relative to ground truth. In contrast, the proposed ZIP regression model (green squares) tends to be much more accurate (i.e., much closer to 1 than the other methods across both data sets).

Balancing Explore-Exploit: By separately modeling the *exposure* and *rate* processes, the ZIP model is able to capture the heterogeneity across users in terms of their *explore/exploit* behavior. In particular, the exposure process coefficient η_{i0} corresponds to the estimated user-specific budget and captures the number of unique items a user will consume. As η_{i0} increases, the probability that a user will be exposed to an item and consume it is also predicted to increase, corresponding to a higher predicted tendency for *exploration*. To illustrate this, in Figure 4 we plot the correlation between the value of η_{i0} and the number of unique items the user consumed in the lastfm (left) and reddit (right) data sets. The clear positive correlation between the two (shown as the regression line in red) demonstrates the ability of our model to achieve an appropriate balance between *exploration* and *exploitation*, resulting in better individual-level predictive models. In addition, the individual-level coefficients provide interpretable detail about each specific user, quantifying their individual tendency for exploration within the context of a broad rate-prediction model.

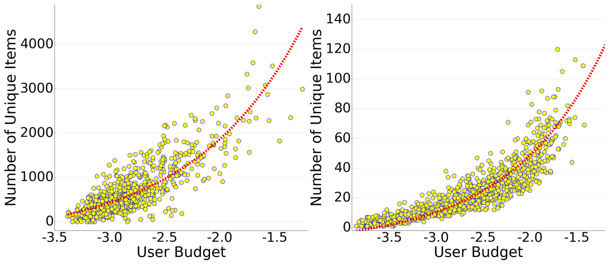

Figure 4: Number of unique items each user consumed as a function of the user-specific budget coefficient (η_{i0}) in the *lastfm* (left) and *reddit* (right) data sets. The red line indicates the exponential curve fitted to the scatter plot.

8 SCALABILITY

In fitting our regression models our dense data matrix can be thought of as having $N \times M \times T$ rows and d columns (for the features), where d is the number of coefficients in the regression, N, M and T are the number of users, items and time-windows respectively. Thus, direct gradient optimization would be $O(dNMT)$ per gradient computation. Using SGD, the time complexity of a single gradient step is $O(d \times R)$ where R is the minibatch size (i.e., the number of data points selected for computing each stochastic estimate of the full gradient).

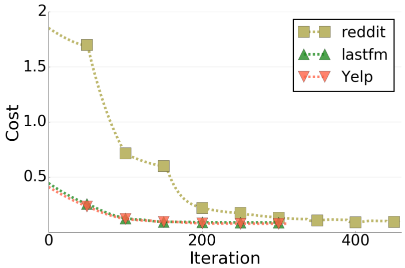

Figure 5: Optimization cost value (negative log-Likelihood) at each SGD iteration for each dataset. Markers denote the point in the iterative process where E-steps were performed. Best viewed in color.

If we think of NMT as the effective total number of rows in the full data set, then to gain the benefits of SGD we need to select R such that $R << NMT$. In typical applications of SGD with dense data the minibatch size can be quite small, e.g., $R = 10, R = 100$. However, with highly skewed data (as in the user-item data sets of interest here), the minibatch sizes need to be significantly larger to ensure that there are enough non-zeros in each minibatch. We found that a minibatch size of $R = 50,000$ worked well in terms of relatively fast and reliable convergence. The time complexity of a single E-step is $O(d \times N \times M \times T)$, i.e., proportional to the number of rows, making it the most expensive part in the algorithm in terms of time complexity. It is possible that some efficiency could be gained here via an approximate E-step but we did not investigate this here given that we execute far more gradient steps (within the M-step) than E-steps.

Figure 5 shows convergence plots, where the y-axis is the cost function (Log-Loss) at each iteration. Each iteration on the x-axis marks a single stochastic gradient (minibatch) step and the markers indicate the point in the algorithm where E-steps occurred (each M-step consists of multiple gradient steps). We see from the convergence plots that the algorithm converges quickly for each of the three data sets in our experiments. We implemented our algorithm in Python (with Cython to speed-up)[3]. Each mini-batch iteration in our implementation ran in a matter of few milliseconds and the relatively expensive E-step took 7, 62, and 1 seconds on average for the reddit, lastfm, and Yelp data sets respectively. Our implementation used a single core—it is relatively straightforward to distribute the computation of the gradients and membership weights by using multiple cores, rendering the algorithm scalable to much larger data sets than what we used here.

9 CONCLUSIONS

We proposed and investigated a framework using zero-inflated Poisson regression for prediction of consumption rates in high-dimensional user-item data sets. The approach is motivated by applications where user consumption is a mix of repeat (exploitation) and novel (exploration) behavior over time. The regression component of the model allows for detailed modeling of individual users based on their histories and provides an alternative to more widely-used latent variable models such as matrix factorization. Experimental results indicate that the proposed approach can systematically outperform existing alternatives such as PMF, DPMF and ZIE for the problem of predicting the rates at which specific users consume specific items. There are a number of natural directions for further explorations of models of this type, including for example modeling of the dynamic changes between time windows within the regression framework, potentially further enhancing the predictive capabilities of the proposed ZIP regression model.

10 ACKNOWLEDGEMENTS

The work in this paper was supported in part by the National Science Foundation award IIS-1320527 and by a Google Faculty Award.

[3] https://github.com/UCIDataLab/ZIP-Regression

REFERENCES

[1] Austin R Benson, Ravi Kumar, and Andrew Tomkins. 2016. Modeling user consumption sequences. In *Proceedings of the 25th International Conference on World Wide Web*. ACM Press, 519–529.

[2] A Colin Cameron and Pravin K Trivedi. 2013. *Regression Analysis of Count Data*. Cambridge University Press.

[3] Laurent Charlin, Rajesh Ranganath, James McInerney, and David M Blei. 2015. Dynamic Poisson factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM Press, 155–162.

[4] Nathaniel D Daw, John P O'Doherty, Peter Dayan, Ben Seymour, and Raymond J Dolan. 2006. Cortical substrates for exploratory decisions in humans. *Nature* 441 (2006), 876–879.

[5] Peter Diggle, Patrick Heagerty, Kung-Yee Liang, and Scott Zeger. 2002. *Analysis of Longitudinal Data*. Oxford University Press.

[6] Prem Gopalan, Jake M Hofman, and David M Blei. 2015. Scalable recommendation with hierarchical Poisson factorization. In *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence*. 326–335.

[7] Daniel B Hall. 2000. Zero-inflated Poisson and binomial regression with random effects: a case study. *Biometrics* 56, 4 (2000), 1030–1039.

[8] Seyed Abbas Hosseini, Keivan Alizadeh, Ali Khodadadi, Ali Arabzadeh, Mehrdad Farajtabar, Hongyuan Zha, and Hamid R Rabiee. 2017. Recurrent Poisson factorization for temporal recommendation. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM Press, 847–855.

[9] Vikas Jain, Nirbhay Modhe, and Piyush Rai. 2017. Scalable generative models for multi-label learning with missing labels. In *Proceedings of the 34th International Conference on Machine Learning*. PMLR, 1636–1644.

[10] How Jing and Alexander J. Smola. 2017. Neural survival recommender. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM Press, 515–524.

[11] Komal Kapoor, Vikas Kumar, Loren Terveen, Joseph A. Konstan, and Paul Schrater. 2015. "I like to explore sometime": adapting to dynamic user novelty preferences. In *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM Press, 19–26.

[12] Komal Kapoor, Karthik Subbian, Jaideep Srivastava, and Paul Schrater. 2015. Just in time recommendations: modeling the dynamics of boredom in activity streams. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*. ACM Press, 233–242.

[13] Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

[14] Y. Koren, R. Bell, and C. Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[15] Diane Lambert. 1992. Zero-inflated Poisson regression, with an application to defects in manufacturing. *Technometrics* 34, 1 (1992), 1–14.

[16] Dawen Liang, Laurent Charlin, James McInerney, and David M. Blei. 2016. Modeling user exposure in recommendation. In *Proceedings of the 25th International Conference on World Wide Web*. ACM Press, 951–961.

[17] Dawen Liang, John William Paisley, and Dan Ellis. 2014. Codebook-based scalable music tagging with Poisson matrix factorization. In *Proceedings of the Fifteenth International Society for Music Information Retrieval Conference*. 167–172.

[18] Li-Ping Liu and David M Blei. 2017. Zero-inflated exponential family embeddings. In *International Conference on Machine Learning*. PMLR, 2140–2148.

[19] Reza Rejaie, Haobo Yu, Mark Handley, and Deborah Estrin. 2000. Multimedia proxy caching mechanism for quality adaptive streaming applications in the internet. In *Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2010)*, Vol. 2. IEEE Press, 980–989.

[20] Martin Ridout, Clarice GB Demétrio, and John Hinde. 1998. Models for count data with many zeros. In *Proceedings of the XIXth International Biometric Conference*. 179–192.

[21] Matthias W Seeger, David Salinas, and Valentin Flunkert. 2016. Bayesian intermittent demand forecasting for large inventories. In *Advances in Neural Information Processing Systems 29*. Curran Associates, Inc., 4646–4654.

[22] Ajit P Singh and Geoffrey J Gordon. 2008. A unified view of matrix factorization models. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer Berlin Heidelberg, 358–373.

[23] Yingzi Wang, Nicholas Jing Yuan, Defu Lian, Linli Xu, Xing Xie, Enhong Chen, and Yong Rui. 2015. Regularity and Conformity: Location Prediction Using Heterogeneous Mobility Data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1275–1284.

[24] Robert C Wilson, Andra Geana, John M White, Elliot A Ludvig, and Jonathan D Cohen. 2014. Humans use directed and random exploration to solve the explore–exploit dilemma. *Journal of Experimental Psychology: General* 143, 6 (2014), 2074–2081.