A Study of Uncertainty in Recommender Systems
DS 6040 – Fall 2022
Angela Boakye Danquah (aab5zd) & Caroline Bush (cjb7tt)
December 12, 2022

Project Description:

Businesses with online platforms often host extremely large catalogs and rely on recommender systems to facilitate efficient browsing. Recommender systems are essential to achieving business goals as they allow consumers to filter through these catalogs and interact with the items they are most likely to enjoy or purchase. The ideal recommender system can direct consumers to items like those they have purchased or interacted with previously and new or more varied items that they may not have discovered on their own. The former type of recommendation is the result of content filtering, which leverages only a single consumer's data and knowledge of product attributes to generate recommendations. Recommendations of new or more varied products, however, are the result of collaborative filtering. This method looks across different consumers to make recommendations of products that are similar to those enjoyed by other users with similar preferences as the target user based on product attributes. It is important to not only estimate the similarities between users and between products, but also to take into consideration the uncertainty around these estimates if one is to make effective recommendations that can also generalize to new users and new items.

Matrix Factorization is a popular method used to construct collaborative filtering recommender systems. This method is typically used as a starting point and can be one of several candidate generators within a larger recommender system. One of the main drawbacks to Matrix Factorization, however, is that it suffers from the cold-start problem: only those items included in the matrix when the model was trained can be recommended. Traditionally, it also fails to measure uncertainty. Several methods have been developed to address the limitations of this model. We first examine the capabilities of Matrix Factorization as a song recommender system. We then expand upon this method with Collaborative Topic Modeling/Regression (CTM), an algorithm introduced by Chong Wang and David Blei, one of the original developers of Latent Dirichlet Allocation (LDA). This method exploits the sentiment analysis capabilities of LDA and Probabilistic Matrix Factorization (PMF) to build a recommender system that can incorporate the 'sentiments' of new items and use them to recommend to users who have engaged with items with similar 'sentiments', enabling the system to perform out-of-matrix prediction. Our data include each song in the "Liked Songs" libraries of 11 Spotify users as well as the audio descriptions made available by Spotify. Our data were collected using Tekore, a Python package that enables connection to Spotify's API.

Probability Model:

In collaborative filtering methods, the objective is to generate customer-specific recommendations that are informed by the engagement history of all customers and the defining characteristics of items in the business' inventory. The development of these models therefore centers around estimating the model parameters that yield the best recommendations as informed by, and later verified by, known consumer actions. Expressed mathematically using Bayes Rule,

$$p(\boldsymbol{\theta}|\boldsymbol{H},\boldsymbol{I}) \propto p(\boldsymbol{H},\boldsymbol{I}|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

where θ are the model parameters, $\boldsymbol{H}$ represents customer history, and $\boldsymbol{I}$ represents inventory characteristics. The parameters, once estimated, may be used to generate predictions.

A complete recommender system should be able to accurately predict which items a given customer will prefer and to rank these items in order to determine the order in which to present them to the customer. By employing Bayesian methods to generate recommendations, each prediction is accompanied by a metric of uncertainty around that recommendation that may be referenced when the system is determining how to rank the recommended items for each user. Estimates of uncertainty in the model parameters also provide the opportunity to evaluate the stability of the model. Our approach, CTM, builds off two models: LDA and PMF.

LDA

LDA determines a fixed number of latent topics present in a corpus of documents based on word frequency. It analyzes word frequency of all words in each document and attempts to maximize the probability of reconstructing that document from a series of draws of topics given a document and words given a topic. The goal of LDA is to maximize the posterior distribution of hidden topics given a document:

$$p(\boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{w}, \alpha, \beta) = \frac{p(\theta, \boldsymbol{z}, \boldsymbol{w}|\alpha, \beta)}{p(\boldsymbol{w}|\alpha, \beta)}$$

in which word w in vocabulary: $w \in \{1, .., Y\}$; document $\mathbf{w}$: $\boldsymbol{w} = (w_1, \dots, w_j)$; Corpus of N documents: $D = \{\boldsymbol{w_1}, \dots, \boldsymbol{w_n})$

The Bayesian Network in Appendix B illustrates the joint distribution below

$$p(\boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\theta}, |\alpha, \beta) = p(\theta|\alpha)p(w_n|z_n, \beta) = p(\theta|\alpha)\prod_{n=1}^{N} p(z_n|\boldsymbol{\theta})p(w_n|z_n\beta)$$

The marginal distribution of a document is

$$p(\boldsymbol{w}|\alpha, \beta) = \int p(\boldsymbol{\theta}|\alpha)(\prod_{n=1}^{N} \sum_{z_n} p(z_n|\boldsymbol{\theta})p(w_n|z_n\beta))d\theta$$

For our purpose, we are interested in the posterior marginal distribution of each of our songs, which we maximized using variational EM. This distribution then augments the latent song feature matrix ***V***

$$V_j = \theta_j + \varepsilon_j$$

Where j denotes the specific song (document).

PMF Model

PMF attempts to estimate the rating $\widehat{R_{n,j}}$ of a user, *n*, on an item, *j*, from a ratings matrix, ***R***. The ratings matrix ***R*** is factored into matrices *U* and *V*, a latent user matrix and song feature matrix respectively. The probability of the ratings matrix is given by

$$p(R|U, V, \sigma^2) = \prod_{n=1}^{N}\prod_{j=1}^{J}[\mathcal{N}(R_{n,j}|U_n^T V_j, \sigma^2)]R_{n,j}|U_n^T V_j, \sigma^2)]^{I_{n,j}}$$

Where $I_{n,j}$ indicates whether user *n* has item *j* in their library. *J* is the number of songs, and *N* is the number of users. We also place Gaussian priors on the matrices:

$$p(U|\sigma_Q^2) = \prod_{n=1}^{N} \mathcal{N}(U_n|0, \sigma_U^2 \boldsymbol{I}), \;\; p(V|\sigma_V^2) = \prod_{j=1}^{J} \mathcal{N}(V_j|\sigma_V^2 \boldsymbol{I})$$

The log of the posterior with respect to the factorized matrices is

$$\ln\,(p(U, V|R, \sigma^2, \sigma_V^2, \sigma_U^2))$$

$$= -\frac{1}{2\sigma^2}\sum_{n=1}^{N}\sum_{j=1}^{J} I_{n,j}(R_{n,j} - U_n^T V_j)^2 - \frac{1}{2\sigma_V^2} V_j^T V_j - \frac{1}{2}((\sum_{n=1}^{N}\sum_{j=1}^{J} I_{nj}) ln\sigma^2 + NDln\sigma_U^2 + IDln\sigma_V^2) + c$$

<u>Approach:</u>

Our first approach to generating a recommender system was Matrix Factorization, a traditionally non-Bayesian method that is a common component of recommender systems in place today. Matrix Factorization works by decomposing the sparse matrix of customer feedback on inventory items into the product of two matrices. The first matrix describes each customer’s preference for a given number of latent features of the inventory items, and the second describes how each inventory item corresponds to these latent features. By placing each user and inventory item in the context of a common set of underlying features, Matrix Factorization can predict any existing customer’s preference for any item that was in the inventory at the time the model was generated.

Matrix factorization is traditionally non-Bayesian. Instead, matrix decomposition is optimized through the minimization of a loss function; traditionally Stochastic Gradient Descent or Weighted Alternating Least Squares (Google Developers, 2022). Probabilistic Matrix Factorization (PMF) extends traditional MF by placing Gaussian priors on the two factored matrices and assuming the ratings are Gaussians ~ $\mathcal{N}(U^T V, \sigma^2)$ (Salakhutdinov & Mnih, 2008). With fixed hyperparameters for noise and prior variances, PMF maximizes the log-posterior with respect to the two factored matrices, minimizing the sum-of-squares error (Salakhutdinov & Mnih, 2008). A truly Bayesian extension of Matrix Factorization, Bayesian Probabilistic Matrix Factorization (BPMF), places priors on the model hyperparameters and employs sampling methods to obtain the posterior probabilities (Salakhutdinov & Mnih, 2008).

In our work with the Spotify data, we employed traditional Matrix Factorization to estimate five latent features among the included tracks. We then generate personalized recommendations for listeners included in the analysis by ranking these ratings and producing the first *K* songs absent from their libraries. Though not a Bayesian method, it was worth exploring to better understand the limits of a frequentist approach in direct comparison to our Bayesian approach, Collaborative Topic Modeling.

Collaborative Topic Modeling expands upon the predictive capabilities of a PMF model by incorporating the results of LDA as a sort of 'prior' on the item-feature matrix $\boldsymbol{V}$ (Blei & Wang, 2011). First, we performed LDA. We chose 15 topics after performing EDA to determine the optimal number of topics. The resulting posterior distribution is intractable. We used variational EM to optimize the posterior distribution of the parameters. We then calculate the augmented item-feature matrix V as $V_j = \theta_j + \varepsilon_j$, where ε is the item-feature matrix computed in the first step of PMF (Blei & Wang, 2011). At this stage, the algorithm simply becomes optimizing the PMF model. The posterior likelihoods of matrices V and U are maximized using stochastic gradient descent (or ascent in our implementation). The rating for each item j and each user n $R_{n,j}$ is then estimated from a Gaussian. $R_{n,j} \sim \mathcal{N}(U^T V, \sigma^2)$. To generate recommendations for a user, the items with the top *K* ratings are selected.

Collaborative Topic Modeling is usually implemented by iteratively optimizing the parameters of LDA and PMF. However, the developers of the algorithm note that similar results are met if the models are optimized sequentially (Blei & Wang, 2011). We therefore chose to construct our model using the posteriors generated by LDA to initialize our PMF model.

Results:

Using data from five out of the eleven of our peers who volunteered their Spotify data, we were able to generate a functional recommender system that uses traditional Matrix Factorization. The underlying code was largely based on the Colab page of a Google Developers tutorial on collaborative filtering (Google 2018), which modified to work with our Spotify data. Once the adapted model was trained, the system could generate any number of track recommendations for a given user (Appendix A, Table 1). These tracks were ranked according to a score calculated from the two estimated matrices; we used cosine as the score (rather than the dot product alone) to control for the popularity of a given track. Scores close to 1.0 indicate that the customer is very likely to enjoy those tracks, and so recommendations are ordered based on this single metric. Unless the model is retrained, the score for any customer-track combination is a fixed prediction.

The addition of LDA, a Bayesian method, enables CTM's out-of-matrix prediction capabilities. The topics defined by LDA not only define the latent features used in the PMF component of the model, but they also present the opportunity to classify new inventory items into an existing, stable framework. For example, the distribution of words associated with Topic 6 include "R&B," "contemporary," and "soul" (Appendix C, Fig. 3). New songs in these genres would likely have high posterior probabilities of association with this topic, and therefore tend to be recommended to those customers who enjoy this type of music. The soft classification of LDA also serves to capture the inherent flexibility of music and musical tastes: songs may be influenced by multiple styles, and listeners are rarely confined to a single genre. Looking at the defining words of each Topic and their representation in the track inventory (Appendix C, Fig. 2), it is evident that LDA was able to use audio features to differentiate between genres and styles of music. The probabilistic (uncertain) classification of songs to these Topics facilitates personalized recommendations based on the types of music a given listener prefers, as indicated by their existing song library.

Using a leave-one-out method, we tested the out-of-matrix predictive abilities of the CTM: one song was excluded from the ratings matrix used to train the initial PMF model, and we used the CTM to generate predictions on this excluded song. Appendix C, Fig. 1 displays the posterior topic probabilities of the track based on its attributes; these probabilities become this song's row in the item-feature matrix used to generate predicted ratings and ultimately generate recommendations. In this way, CTM is able to recommend new inventory items to existing consumers, a highly advantageous feature for businesses that are constantly updating their inventories.

Conclusions:

Recommender systems use a variety of methods to identify items a customer is likely to enjoy and rank those items to present to the customer as targeted suggestions. Both methods employed here, Matrix Factorization and Collaborative Topic Modeling, fulfill this basic function. However, it is also necessary to evaluate the usability of the recommendations made and the limitations of each method in a real-world context.

The Matrix Factorization model successfully generated personalized recommendations that were grounded in data spanning across users and the entire library of tracks. While our recommender system was able to rank recommended tracks, these rankings were based on point estimates alone. It is therefore impossible to consider the uncertainty around each prediction when considering how best to present candidate tracks to the customer. Furthermore, the lack of information about the uncertainty in the matrix factorization prevents even basic evaluation of the model fitting process itself. Implementing a Bayesian method, such as Bayesian Probabilistic Matrix Factorization (BPMF) or Collaborative Topic Modeling (CTM) is a logical next step in generating an effective recommender system that employs Matrix Factorization. We could also expand this work beyond simple presence/absence data, and instead ground the recommender system in more detailed implicit feedback such as the number of times a track has been played.

As with Matrix Factorization, the Collaborative Topic Model was able to generate personalized recommendations using latent features of each song in the library of tracks and the consumers' preferences for these latent features. Critically, these latent features are the posterior probabilities of classification to each of the 15 Topics generated by LDA. The use of LDA and stable Topics as the latent features in PMF enables the system to incorporate and recommend new tracks to existing customers without needing to retrain the model with each addition. Furthermore, LDA's soft classification of songs into the 15 Topics is ideal for music and musical tastes, which are rarely confined to a single genre or style. By allowing for probabilistic classification, CTM inherently incorporates the uncertainty in how much a track conforms to a given genre and in which genres appeal to a given listener. One drawback to CTM, however, is that its use of LDA demands that track attributes are concatenated into a single string. Many of the audio features present in our data were continuous numerical features and some information was lost by discretizing these features. To mitigate this issue, we would consider adapting another method to replace LDA that can handle continuous features. Ideally, this method would preserve LDA's soft classification into fixed topics, but allow greater flexibility in the input data.

References:

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of machine Learning research*, *3*(Jan), 993-1022.

Google (2018). Recommendation Systems with TensorFlow. *Google Machine Learning Education.* Accessed via https://github.com/google/eng-edu/blob/main/ml/recommendation-systems/recommendation-systems.ipynb

Google Developers (2022). Recommender Systems: Matrix Factorization, *Google Machine Learning Education.* Accessed via https://developers.google.com/machine-learning/recommendation/collaborative/matrix

Mnih, A., & Salakhutdinov, R. R. (2007). Probabilistic Matrix Factorization. *Advances in neural information processing systems*, *20*.

Rendle, S., Freudenthaler, C., Gantner, Z., and Schmidt-Thieme, L. (2009). BPR: Bayesian Personalized Ranking from Implicit Feedback. *UAI*, 452-461.

Russo, M. D. (2020, February 19). *A guide to collaborative topic modeling recommender systems*. Medium. Retrieved December 11, 2022, from https://towardsdatascience.com/a-guide-to-collaborative-topic-modeling-recommender-systems-49fd576cc871

Salakhutdinov, R. & Mnih, A. (2008). Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo. *Proceedings of the 25th International Conference on Machine Learning*. Helsinki, Finland.

Wang, C., & Blei, D. M. (2011, August). Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 448-456).

Appendix A:

Table 1: Personalized track recommendations for User 0, as generated by the Matrix Factorization model fit using Spotify data.

```
user_recommendations(model, 0, measure=COSINE, exclude_rated = True, k=10)
```

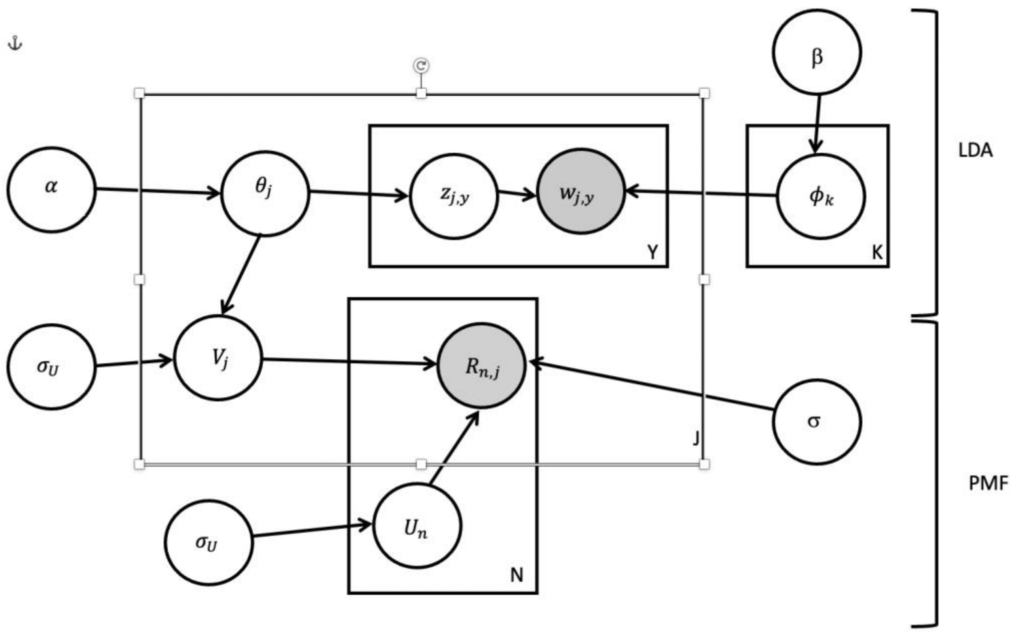
	cosine score	track_no	track_id	title	artist
3740	0.985464	3740	7pamxApUs3CE8t7tKmUJ6Z	The Bird	Anderson .Paak
1877	0.971316	1877	1o7S218Z4CG6IVCOudaPcT	Lucy's Love (feat. Lil Wayne)	SiR
4217	0.965695	4217	6sxptembJVty4sNtcPMAVz	Man! I Feel Like A Woman!	Shania Twain
2182	0.958516	2182	09PaypEcFmih85xXedBBKP	War	SiR
3182	0.952783	3182	6oF3Es1YzzmLKjGBfThUvD	Worst Behavior	Drake
4283	0.947920	4283	3TNSVsiFngfe68UJpMq1oS	Starting Over	Chris Stapleton
3646	0.945584	3646	63wKtmUg754umO8SglEL79	Jealous	Beyoncé
1814	0.945145	1814	4naDApqjX8QsqdTj4hReMu	Wishing On A Star	Will Downing
5703	0.940168	5703	5R6bRLwPyEj7LLMSJW766C	Girls Just Wanna Have Fun	Miley Cyrus
1700	0.938780	1700	6CUWzTRmC1GjjSDTtiVfsl	None Of Your Business	Salt-N-Pepa

Table 2: Personalized track recommendations for User 0, as generated by the CTM model

	track	rating
0	(a sky like i\ve never seen (from the amazon o...	1.572039
1	(sita ram, alice coltrane)	1.552400
2	(tried and true, ween)	1.544628
3	(jupiter, gallant)	1.535673
4	(white water - bonus track, angel olsen)	1.528718
5	(jupiter, jenny hval)	1.526376
6	(tripasia, cloonee)	1.523265
7	(royal and desire, animal collective)	1.517512
8	(trop beau, lomepal)	1.512514
9	(tres notas para decir te quiero, vicente amigo)	1.512275

Appendix B:

Figure 1: Bayesian Network representation of the joint distribution describing Collaborative Topic Modeling.



Appendix C

Figure 1: Posterior distribution of held-out observation given topic.

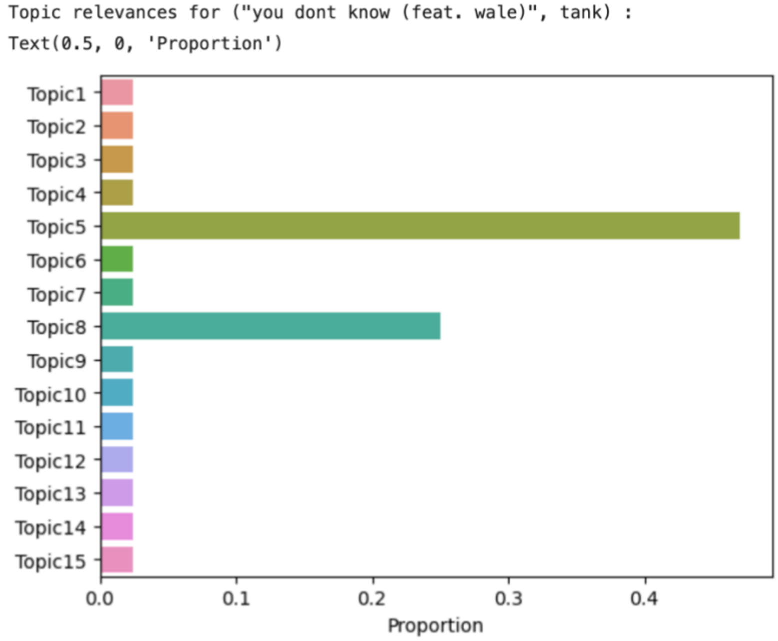


Figure 2: Posterior Marginal Distribution of Topics

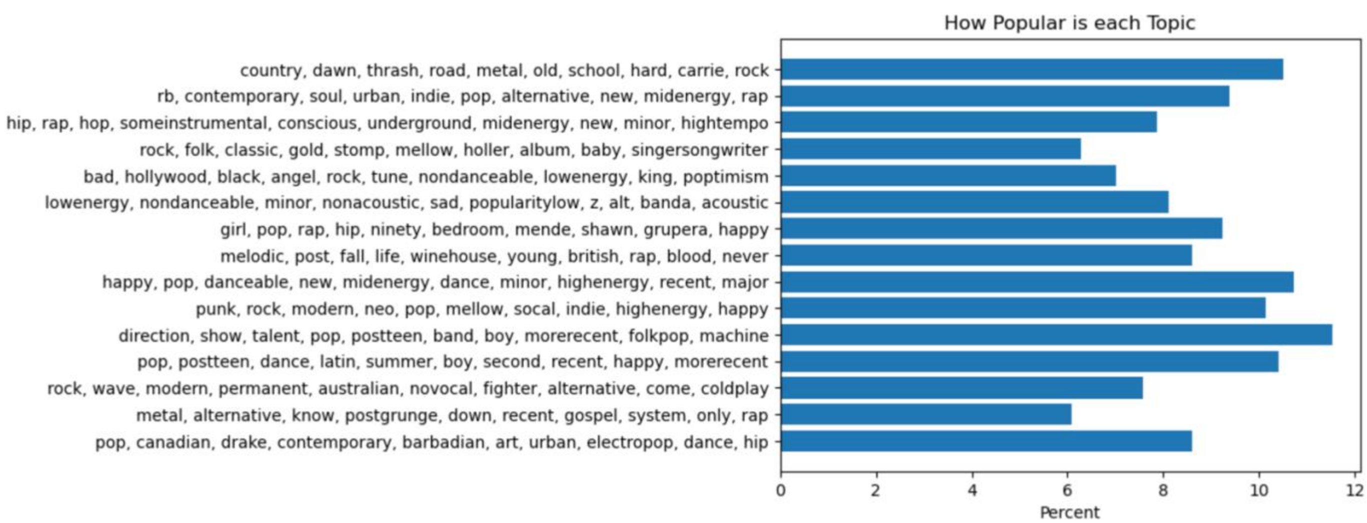


Figure 3: Posterior Distribution of Topic 2