Homework 3
Hengxu Lin (hl3541)
November 7, 2022

1 Problem: Hierarchical Poisson Factorization

1.1 Data

- **Recap Description**. The Expedia dataset is a private dataset obtained from Wharton Customer Analytics. The data is comprised of a search transaction dataset and a clickstream dataset. I use the search transaction dataset in this project, which includes user search record per hotel on Expedia.
- **Selected Features.** I follow the feature selection as previous HWs. Specifically in this homework, I focus on the traveler rating of the hotel. We could model the rating data to discover customer preference. In this homework I use Hierarchical Poisson Factorization (HPF) to decompose the rating matrix, and give analysis about the prediction (recommendation). After processed null value, I obtain a matrix with 2592 users $\times$ 924 item.

1.2 Problem Formulation

Given a user-item rating matrix $Y_{n_U \times n_I}$, we assume that each user could be represented by K latent preference components, meanwhile, each item is represented by K latent components as well, which could be regarded as "score" (attribute) over these components. The data generating process is:

1. For each user u:
   - Sample activity from prior $\xi_u \sim \text{Gamma}(a_1, \frac{a_1}{b_1})$ -> modeled by κ_u
   - Sample preference for each component $\theta_{uk} \sim \text{Gamma}(a, \xi_u)$ -> modeled by γ_{uk}
2. For each item i:
   - Sample popularity from prior $\eta_i \sim \text{Gamma}(c_1, \frac{c_1}{d_1})$ -> modeled by τ_i
   - Sample attribute for each component $\beta_{ik} \sim \text{Gamma}(c, \eta_i)$ -> modeled by λ_{ik}
3. For each user per item, sample rating

$$y_{ui} \sim \text{Poisson}(\theta_u^\intercal \beta_i) \text{ -> modeled by } \phi_{ui}$$

The posterior is approximated by variational parameters $\kappa_u, \gamma_{uk}, \tau_i, \lambda_{ik}, \phi_{ui}$, by the mean-field family assumption, the variational distribution is

$$q(\beta, \theta, \eta, \xi, z) = \prod_{i,k} q(\beta_{ik}|\lambda_{ik}) \prod_{u,k} q(\theta_{uk}|\gamma_{uk}) \prod_{i} q(\eta_i|\tau_i) \prod_{u} q(\xi_u|\kappa_u) \prod_{u,i} q(z_{ui}|\phi_{ui})$$

1.3 Variational Inference Optimization

We use conditional ascent variational inference (CAVI) to optimize the variational parameters. I check the convergence by the log joint:

$$\log p(y|\theta, \beta) = \sum_{y_{ui}>0} \left[y_{ui} \log(\theta_u^\intercal \beta_i) - \log(\Gamma(y_{ui})) \right] - \Big(\sum_u \theta_u^\intercal \Big) \Big(\sum_i \beta_i \Big)$$

I also compute the distance (MSE) between the prediction ϕ_{ui} and true rating y_{ui}.

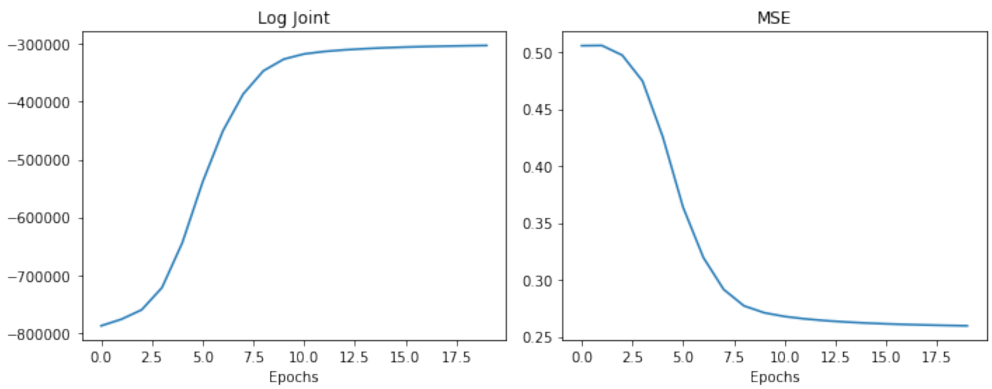

Figure 1: Log Joint Probability of the Posterior

1.4 Evaluation

First I check different hyper-parameter of K, which is the dimension of latent components of users and items. As K increases, the log joint is increasing and training loss is decreasing.

After the optimization we obtain the decomposed matrices (i.e. latent parameters) θ, β, which yield the predicted rating (i.e. recommendation) $\phi = \theta^\intercal \beta$. I also evaluate the performance of accuracy over top N items. The top N item accuracy is defined as:

$$\text{Acc} = \frac{N_{\Omega_{\text{Top N recommend}} \cap \Omega_{\text{Top N rating}} \neq \emptyset}}{N_{items}}$$

	k=3	k=5	k=10	k=20	k=50
top1	0.076389	0.075231	0.057099	0.033565	0.077932
top5	0.594522	0.670910	0.691744	0.734568	0.756173
top10	0.761188	0.916667	0.937500	0.957176	0.972222

Table 1: Top N Accuracy of the Recommendation

A number of $k >= 10$ is suitable in terms of log likelihood and accuracy. As we can see, larger k increases the expressiblity of the model and hence increase top N accuracy. However, HPF still has space for improvement, as we can also see that the top 1 accuracy is far low (<10%) even we increase the number of parameters.

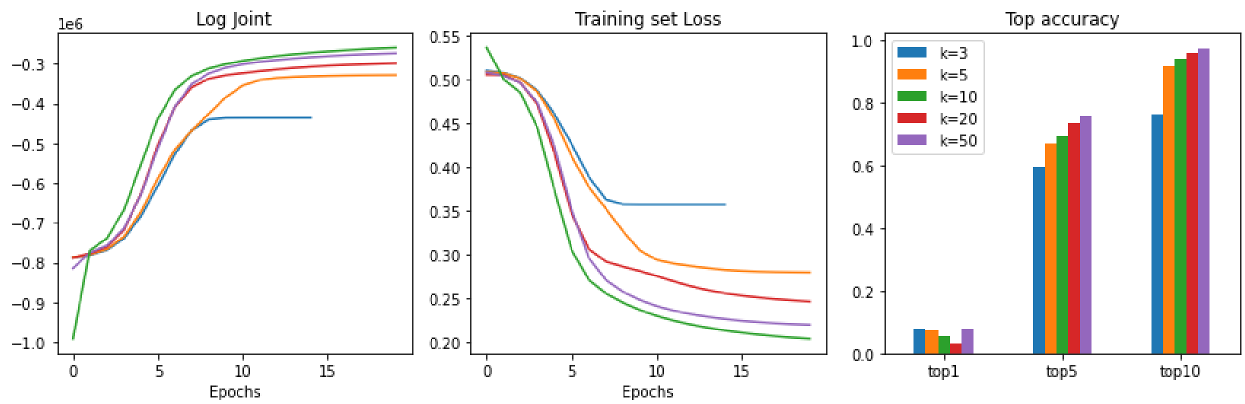

Figure 2: HPF Over Different latent dimension K

Finally I plot three random item recommendations. The recommendation is the predicted rating given by HPF, in contrast to the true rating. The overlap of the recommendation and rating shows our model's performance.

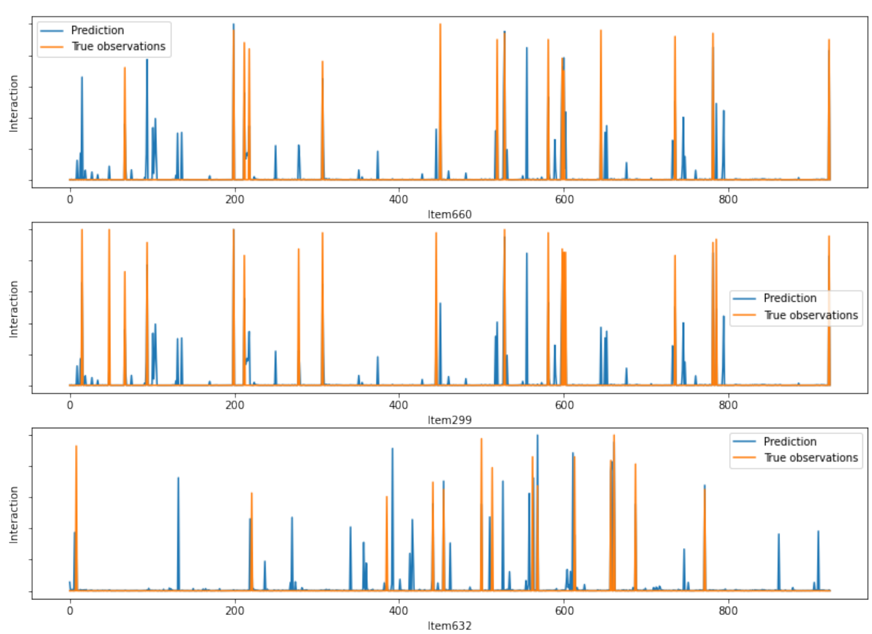

Figure 3: Recommendation v.s. True Rating over Items