CSC2506 Assignment 3

Matthew Scicluna

2017-01-13

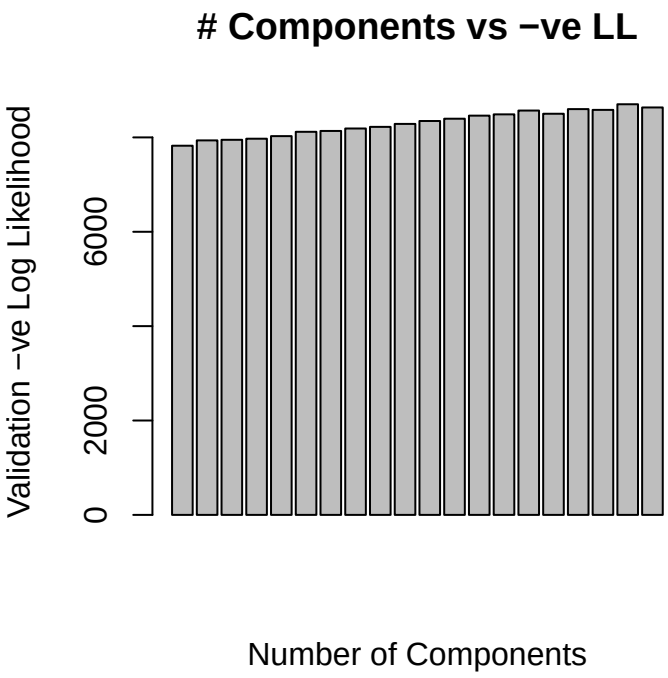

Figure 1: Negative log likelihood of model trained with different numbers of clusters

Training a Mixture Model for Movie Ratings

We trained a multinomial mixture model on the MovieLens data. The dataset consists of ratings of movies (1 to 5) of 843 users of 89 movies. We limited the movies to only ones with at least 200 ratings to avoid overfitting. To import the data we used the `R.matlab` package. Information about the package can be found here. We used a custom written EM algorithm to train the model since our data had missing values in it. Our model was similar to the standard multinomial EM model but with indicators added to handle the missing ratings.

Optimizing the Number of Mixture Components

We trained the model varying the number of mixture components on a subset of the training data. We plotted the number of components in each model with the to see log likelihood of the remaining training data (the validation set). Results can be found in figure 1. We found that the model with the best validation log likelihood had 2 clusters. Surprisingly, increasing the number of components beyond this increased the negative validation data log likelihood.

Consistency of Log Likelihood Across Different Initializations

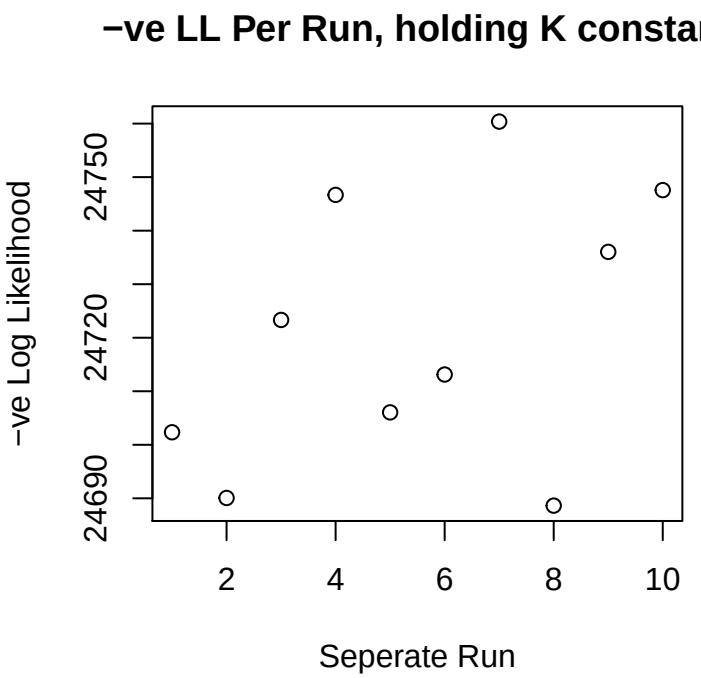

Figure 2: Assessing the consistency of the negative log likelihood after several random restarts

From figure 2 We see that the model can vary across different initializations. This is not surprising since the model is supposed to be sensative to the initial values of the betas.

Expected Complete data log likelihood as a function of EM Iteration

For the model with 2 components, we plotted the negative log likelihood as a function of the number of iterations. We see that the model decreases monotonically, as is expected. The results are presented in figure 3. We note that we implemented early stopping after 6 interations to avoid overfitting.

Interpretation of the Mixture Components

Table 1: Demographic profile of each cluster

	1	2
Number of Females	138	108
Number of Males	325	271
Avg Age	34.5	33.3
Most Common Occupation	student	student
2nd Most Common Occupation	educator	other
3rd Most Common Occupation	other	engineer

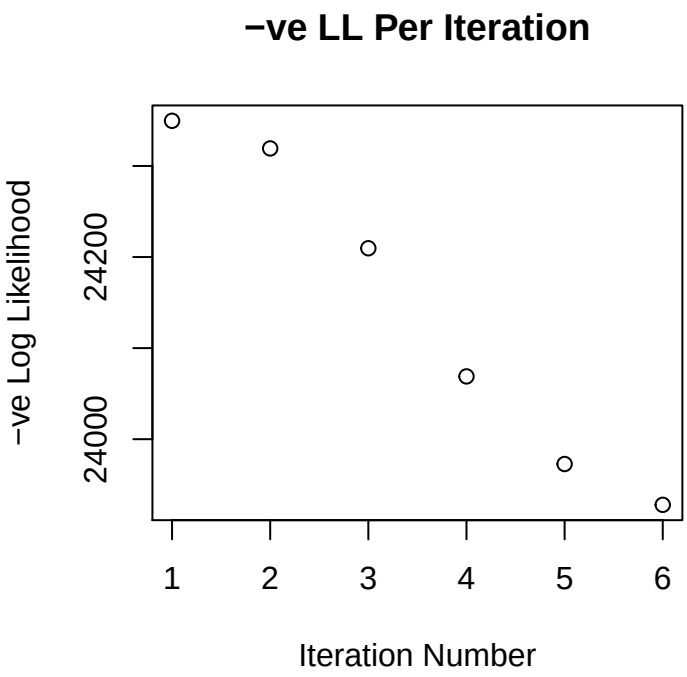

Figure 3: Expected complete (negative) data log likelihood as a function of iteration of the EM algorithm

We analyzed the demographic composition of the 2 clusters our best model. The data was made available from the MovieLens database. We analyzed number of males and females, average age, and the three most common occupations. We did not find any discernable pattern for any number of clusters. For completeness we present the results in table 1.

Dirichlet Prior

We derive EM expressions to train a multinomial mixture model with a Dirichlet prior over the β_{vjk} parameters. $P(\beta_{jk} \mid \phi_k) = D(\beta_{jk} \mid \phi_k)$

Denote the marginal density of the latent variable as $P(Z_i) = \theta_k$ and the posterior ratings distribution as $P(R_i \mid Z_i = k, \beta) = \prod_j \prod_v \beta_{vjk}^{[r_{ij}=v]} = Categorical(\beta_{vjk})$.

First we compute the posterior distribution of the latent variable. This amounts to computing the responisibilities in the E step in the EM algorithm.

$$P(Z_i = k \mid R_i, \beta, \phi, \theta) = P(Z_i = k \mid R_i, \beta, \theta) \tag{1}$$

$$= \frac{P(R_i \mid Z_i = k, \beta, \theta) P(Z_i = k \mid \theta)}{\sum_m P(R_i \mid Z_i = m, \beta, \theta) P(Z_i = m \mid \theta)} \tag{2}$$

$$= \frac{\prod_j \prod_v \beta_{vjk}^{[r_{ij}=v]} \theta_k}{\sum_m \prod_j \prod_v \beta_{vjm}^{[r_{ij}=v]} \theta_m} \tag{3}$$

We see that this does not affect the responsibilities of the latent variable. We denote the responsibility of cluster k to data point i as $\gamma_{ik} := P(Z_i = k \mid R_i, \theta_k, \phi_k)$.

We note that since the Dirichlet distribution is conjugate to the categorical distribution so we can easily compute the joint distribution of the data and the latent variable (Tu, S).

$$
\begin{aligned}
P(Z_i = k, R_i \mid \phi, \theta) &= P(R_i \mid Z_i = k, \phi) P(Z_i = k \mid \theta) && (4)\\
&= P(R_i \mid Z_i = k, \beta_{jk}) P(\beta_{jk} \mid \phi_k) P(Z_i = k \mid \theta) && (5)\\
&= Categorical(\beta_{vjk}) D(\beta_{jk} \mid \phi_k) \theta_k && (6)\\
&= D([r_{ij} = v] + \phi_{vk} - 1)\theta_k && (7)
\end{aligned}
$$

We now compute L, the Expected complete data log likelihood so we can get our β and θ updates. Note that the Expected complete data log likelihood is exactly the observed data log likelihood (Marlin, B).

$$
\begin{aligned}
L &= \sum_i \sum_k P(Z_i = k \mid R_i, \theta_k, \phi_k) log P(Z_i = k, R_i \mid \theta_k, \phi_k) && (8)\\
&= \sum_i \sum_k \left(\sum_v [r_{ik} = v] P(Z_i = k \mid R_i, \theta_k, \phi_k) \right) log P(Z_i = k, R_i \mid \theta_k, \phi_k) && (9)\\
&= \sum_i \sum_k \gamma_{ik} log P(Z_i = k, R_i \mid \theta_k, \phi_k) && (10)\\
&= \sum_i \sum_k \gamma_{ik} \left(log\theta_k + logD([r_{ij} = v] + \phi_{vk} - 1) \right) && (11)\\
&= \sum_i \sum_k \gamma_{ik} \left(log\theta_k + \sum_j \left(\sum_v [r_{ij} = v] + \phi_{vk} - 1 \right) log\beta_{vjk} - logB(\phi_k) \right) && (12)
\end{aligned}
$$

To solve for θ we differentiate L and apply a Lagrange multiplier to ensure the contraints are obeyed:

$$
\begin{aligned}
\frac{\partial}{\partial \theta_k} L + \lambda \left(\sum_k \theta_k - 1 \right) &= \frac{\sum_i \gamma_{ik}}{\theta_k} + \lambda = 0 && (13)\\
&\Rightarrow \lambda \theta_k = \sum_i \gamma_{ik} && (14)\\
&\Rightarrow \theta_k^{MAP} = \frac{\sum_i \gamma_{ik}}{\sum_m \sum_i \gamma_{im}} && (15)
\end{aligned}
$$

Note that (15) follows from (14) upon normalizing θ_k.

To solve for β_{vjk} we again differentiate L and apply the appropriate Lagrange multiplier:

$$\frac{\partial}{\partial \beta_{vjk}} L + \lambda \left(\sum_v \beta_{vjk} - 1 \right) = 0 \tag{16}$$

$$\Rightarrow \frac{\partial}{\partial \beta_{vjk}} \sum_i \sum_k \gamma_{ik} \left(\sum_j \left(\sum_v [r_{ij} = v] + \phi_{vk} - 1 \right) log \beta_{vjk} \right) + \lambda = 0 \tag{17}$$

$$\Rightarrow \frac{\sum_i \gamma_{ik} [r_{ij} = v] + \phi_{vk} - 1}{\beta_{vjk}} = \lambda \tag{18}$$

$$\Rightarrow \beta_{vjk}^{MAP} = \frac{1}{\lambda} \left(\sum_i \gamma_{ik} [r_{ij} = v] + \phi_{vk} - 1 \right) \tag{19}$$

$$\Rightarrow \beta_{vjk}^{MAP} = \frac{\sum_i \gamma_{ik} [r_{ij} = v] + \phi_{vk} - 1}{\sum_m \sum_i \gamma_{ik} [r_{ij} = m] + \sum_m \phi_{mk} - M} \tag{20}$$

Again we see that (20) follows from (19) by normalization of β_{vjk}. From this update formula We can see that adding the Dirichlet prior amounts to adding "pseudo-counts" to the beta. This is beneficial since it acts as a regularizer and ensures that the computations do not become numerically stable (i.e. ensuring that the denominator doesn't vanish to zero).

References

[1] Murphy, K. Machine Learning: A Probablistic Perspective. The MIT Press, Cambridge, 2012.

[2] Marlin, B. Collaborative Filtering: A Machine Learning Perspective (Masters Thesis), 2004.

[3] Tu, S. The Dirichlet-Multinomial and Dirichlet-Categorical models for Bayesian inference.