EECS 445 Discussion 10: Collaborative Filtering and Intro to Generative Models

April 1, 2021

1 Matrix Factorization

Recommendation systems were historically the result of substantial human effort. For example, the music service Pandora used to hire experts to create curated playlists that corresponded to specific genres and moods. Collaborative filtering both negates the need for substantial expert knowledge and improves upon the quality of recommendations by solving a regression problem for the expected rating that a user would give to a piece of content using only a metric of similarity between users. For instance, if user A and user B liked very similar music and user A likes song X that B has not heard, a collaborative filtering algorithm would conclude that user B may also like song X. This is the same technique currently used by corporations such as Spotify and Netflix.

Here we will focus on the matrix factorization technique for collaborative filtering. Consider the case of n users and m possible song selections. Let Y_{ai} be the observed rating that user a assigned to song i, where $Y_{ai} \in \{-1, 0, 1\}$. We wish to construct an approximation $\hat{Y}_{ai}$ for *all* values of a, i to satisfy the following optimization problem.

$$\min_{\hat{Y}} J(\hat{Y}) = \min_{\hat{Y}} \frac{1}{2} \sum_{(a,i) \in D} (Y_{ai} - \hat{Y}_{ai})^2 + \frac{\lambda}{2} \sum_{a,i} (\hat{Y}_{ai})^2 \tag{1}$$

Here, D represents the set of all observed ratings and λ is a tuneable hyperparameter. Note that without any constraint on $\hat{Y}$, we end up with the trivial solution that $\hat{Y}_{ai} = 0$ for all unobserved values. We therefore constrain $\hat{Y}$ to be *low-rank*, so that for some $U \in \mathbb{R}^{n \times d}$ and $V \in \mathbb{R}^{m \times d}$, where $d \leq \min(n, m)$, we have that $\hat{Y} = UV^T$. Our new optimization problem is as follows.

$$\min_{U,V} J(U, V) = \min_{U,V} \frac{1}{2} \sum_{(a,i) \in D} (Y_{ai} - \bar{u}^{(a)} \cdot \bar{v}^{(i)})^2 + \frac{\lambda}{2} \sum_{a=1}^{n} \|\bar{u}^{(a)}\|^2 + \frac{\lambda}{2} \sum_{i=1}^{m} \|\bar{v}^{(i)}\|^2 \tag{2}$$

Note that equation 1 and equation 2 are not necessarily equal

With our cost function defined, we now consider how to proceed with constructing an algorithm to minimize this cost. We will use an alternating minimization, meaning that we will hold one minimization parameter constant while optimizing with respect to the other, and then switch. This alternation is repeated multiple times until some stopping criteria is met (e.g., after a fixed number of iterations or when the magnitude of an update is sufficiently small).

Here we will go through one step of the optimization given the following observation matrix, in which rows represent the ratings provided by a user, and columns represent all user's ratings of a particular song. Note that in this case, there are not any missing ratings.

$$Y = \begin{bmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \end{bmatrix}$$

Define our initial matrices U and V to be as follows.

$$U = \begin{bmatrix} 2 & 2 \end{bmatrix}^T \qquad V = \begin{bmatrix} -1 & 1 & -1 \end{bmatrix}^T$$

Discussion Question 1. *What is d, the rank of UV^T?*

Solution: The rank of a matrix product is at most the minimum of the rank of any of the component matrices. Therefore, $d = 1$.

Discussion Question 2. *What is the value of $\frac{1}{2}(Y_{2,3} - \bar{u}^{(2)} \cdot \bar{v}^{(3)})^2$, the error of our approximation of user 2's rating of song 3?*

Solution:

$$\frac{1}{2}(Y_{2,3} - \bar{u}^{(2)} \cdot \bar{v}^{(3)})^2 = \frac{1}{2}(0+2)^2 = 2$$

Discussion Question 3. *Assume we hold V constant. Solve for $\nabla_{\bar{u}^{(2)}} J$.*

Solution:

$$\nabla_{\bar{u}^{(2)}} J = -\sum_{2,i \in D} (Y_{2,i} - \bar{u}^{(2)} \cdot \bar{v}^{(i)})\bar{v}^{(i)} + \lambda \bar{u}^{(2)}$$

Discussion Question 4. *Assume $\lambda = 1$. For what value of $\bar{u}^{(2)}$ is J minimized?*

Solution: We set $\nabla_{\bar{u}^{(2)}} J = 0$ and solve for $\bar{u}^{(2)}$ to get

$$\bar{u}^{(2)} = (I_{1\times 1} + \sum_{2,i \in D} \bar{v}^{(i)} \bar{v}^{(i)T})^{-1} (\sum_{2,i \in D} Y_{2,i} \bar{v}^{(i)}) = -\frac{1}{2}$$

Discussion Question 5. *What is our new value of $\frac{1}{2}(Y_{2,3} - \bar{u}^{(2)} \cdot \bar{v}^{(3)})^2$ after updating $\bar{u}^{(2)}$?*

Solution:

$$\frac{1}{2}(Y_{2,3} - \bar{u}^{(2)} \cdot \bar{v}^{(3)})^2 = \frac{1}{2}\left(0 - \frac{1}{2}\right)^2 = \frac{1}{8}$$

2 K Nearest Neighbors

In problems like matrix completion, we borrow data from similar rows to infer better estimates of the values for missing entries. In the example of recommendation systems (i.e. Netflix), one method of finding similar users is known as K-Nearest-Neighbors (KNN prediction). The idea is to find the k most "similar" users to the user in question, and make predictions based on these users. This entails the following: 1) Defining a metric of similarity, 2) Calculating the similarity from the user in question to all other users, and 3) Rate movie i for the user in question based on the data from the k most similar users.

First we must define some metric of similarity for two users a and b. Following the example from lecture, let $R(a, b)$ be the set of movies rated by both users a and b. Let the rows of matrix Y represent user data, and each column as specific movie, where Y_{ai} gives the rating of user a for movie i. We write the average rating of user a for movies rated by both users as:

$$\tilde{Y}_{a:b} = \frac{1}{|R(a,b)|} \sum_{j \in R(a,b)} Y_{aj}.$$

Given this, we define the correlation between users a and b as a metric of similarity as follows:

$$sim(a, b) := \frac{\text{N = expected value of how much user ratings vary together}}{\text{D = how much the ratings vary individually}}$$

Where:

$$\text{N} = \sum_{j \in R(a,b)} (Y_{aj} - \tilde{Y}_{a:b})(Y_{bj} - \tilde{Y}_{b:a})$$

And:

$$D = \sqrt{\sum_{j \in R(a,b)} (Y_{aj} - \tilde{Y}_{a:b})^2 \sum_{j \in R(a,b)} (Y_{bj} - \tilde{Y}_{b:a})^2}$$

This metric of similarity ranges in [-1, 1], where 0 represents a complete lack of linear correlation, while -1 and 1 represent a perfect linear relationship.

Discussion Question 6. *Given the following matrix Y, let the first row represent user a, the second row represent user b, and the third row represent user c. Calculate the similarity between users a and b, and between users a and c:*

$$Y = \begin{bmatrix} 3 & 3 & ? & 6 & 8 \\ 2 & 5 & 0 & 10 & 7 \\ 5 & 1 & 3 & 3 & ? \end{bmatrix}$$

Solution:

$$\tilde{Y}_{a:b} = 20/4 = 5, \tilde{Y}_{b:a} = 24/4 = 6$$

$$sim(a, b) = \frac{(3-5)(2-6) + (3-5)(5-6) + (6-5)(10-6) + (8-5)(7-6)}{\sqrt{((3-5)^2 + (3-5)^2 + (6-5)^2 + (8-5)^2) * ((2-6)^2 + (5-6)^2 + (10-6)^2 + (7-6)^2)}}$$

$$= 0.69$$

$$\tilde{Y}_{a:c} = 12/3 = 4, \tilde{Y}_{c:a} = 9/3 = 3$$

$$sim(a, c) = \frac{(3-4)(5-3) + (3-4)(1-3) + (6-4)(3-3)}{\sqrt{\cdots}} = \frac{0}{\sqrt{\cdots}} = 0$$

Finally, let KNN(a, i) represent the K most similar users to a who have also rated movie i. Given the results of our K "most similar" users, we predict on user a's rating of movie i using the following equation:

$$\hat{Y}_{ai} = \bar{Y}_a + \frac{1}{\sum_{b \in KNN(a,i)} |sim(a,b)|} \sum_{b \in KNN(a,i)} sim(a,b)(Y_{bi} - \bar{Y}_b)$$

Discussion Question 7. *Assume we are using the same rating matrix as before, with KNN for K=1 such that b is the most similar user to a. Give a prediction for* $\hat{Y}_{a3}$.

Solution:

$$\hat{Y}_{a3} = 5 + \left(\frac{1}{.69} * (.69)(0 - 4.8) \right) = 0.2$$

3 Generative Models

We often make the assumption that our input data is sampled from some distribution (i.e., $\bar{x}^{(i)} \sim p(\bar{x})$). Thus far in the course, we have not dealt directly with this underlying distribution and instead focused on identifying key factors of the data to place it into a category or regress its features to estimate a value. A machine learning model that performs classification or regression is called a *discriminative model*. Here's an entirely different framework: let's use a *generative model* to try to learn $p(\bar{x})$ directly! Generative models parameterize a probability distribution and learn parameter values from observed data. In other words, we want to create a probability distribution $p_{\bar{\theta}}(\bar{x})$ as similar as possible to $p(\bar{x})$. Generative models are typically divided into two categories: models that aim to explicitly represent the probability density of the data, and models that implicitly represent the probability density as a black box. Models using explicit representations of density first define a probabilistic structure and learn parameters for that structure (e.g., assume the data is normally distributed and learn the mean and variance). Implicit representations of density make no assumptions of structure, and learn both the structure and values in one—usually uninterpretable—real-valued vector. Once trained, generative models allow for new data points to be sampled directly from the learned distribution: $\hat{x} \sim p_{\bar{\theta}}(\bar{x})$.

3.1 Maximum Likelihood Estimation

The point of Maximum Likelihood Estimation is to find the parameters of a probabilistic model $\bar{\theta}$ that maximize the likelihood of the data X being sampled from the distribution. This is a core principle used in generative models.

$$\begin{aligned}\bar{\theta}^* &= \underset{\bar{\theta}}{\operatorname{argmax}}\, p_{\bar{\theta}}(X) \\ &= \underset{\bar{\theta}}{\operatorname{argmax}}\, \ln p_{\bar{\theta}}(X) \\ &= \underset{\bar{\theta}}{\operatorname{argmin}}\, -\ln p_{\bar{\theta}}(X)\end{aligned}$$

Discussion Question 8. *Let $x \sim \text{Binomial}(x; N, p)$ and $\bar{\theta} = (N, p)$ where x is one observation, N is total number of Bernoulli observations, n is number of successes observations and p is probability of success. What is the maximum likelihood estimate of p given observations $X = [x^{(1)}, \ldots, x^{(n)}]$? The binomial distribution is as follows:*

$$\text{Binomial}(x; N, p) = \binom{N}{x} p^x (1-p)^{N-x} \tag{3}$$

Solution: We first compute the log-likelihood as follows:

$$\begin{aligned}
\ln \ell(X) &= \ln \prod_{i=1}^{n} \text{Binomial}(x^{(i)}; N, p) \\
&= \sum_{i=1}^{n} \ln \Big(\text{Binomial}(x^{(i)}; N, p) \Big) \\
&= \sum_{i=1}^{n} \ln \left(\binom{N}{x^{(i)}} p^{x^{(i)}} (1-p)^{N-x^{(i)}} \right) \\
&= \sum_{i=1}^{n} \ln \binom{N}{x^{(i)}} + x^{(i)} \ln p + (N - x^{(i)}) \ln(1-p)
\end{aligned} \tag{4}$$

Next, we optimize w.r.t. p:

$$\begin{aligned}
\nabla_p \ln \ell(X) &= \sum_{i=1}^{n} \frac{x^{(i)}}{p} - \frac{N - x^{(i)}}{1-p} \\
\frac{1}{p} \sum_{i=1}^{n} x^{(i)} &= \frac{1}{1-p} \sum_{i=1}^{n} N - x^{(i)} \\
\frac{1-p}{p} \sum_{i=1}^{n} x^{(i)} &= Nn - \sum_{i=1}^{n} x^{(i)} \\
\sum_{i=1}^{n} x^{(i)} \left(\frac{1-p}{p} + \frac{p}{p} \right) &= Nn \\
p &= \frac{1}{Nn} \sum_{i=1}^{n} x^{(i)}
\end{aligned} \tag{5}$$

We achieve the intuitive result that the MLE estimate of the parameter is just an average of the observations.

Discussion Question 9. *Let $x \sim \text{Normal}(x; \mu, \sigma^2)$ and $\bar{\theta} = (\mu, \sigma^2)$ where x is one observation, μ is the mean of the normal distribution, and σ^2 is the variance of the normal distribution. What are the maximum likelihood estimates of μ and σ^2? The univariate normal distribution is as follows:*

$$\text{Normal}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}(x-\mu)^2) \tag{6}$$

Solution: We first compute the log-likelihood as follows:

$$\begin{aligned}
\ln \ell(X) &= \ln \prod_{i=1}^{n} \text{Normal}(x^{(i)}; \mu, \sigma^2) \\
&= \sum_{i=1}^{n} \ln \Big(\text{Normal}(x^{(i)}; \mu, \sigma^2) \Big) \\
&= \sum_{i=1}^{n} \ln \left(\frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}(x^{(i)} - \mu)^2) \right) \\
&= \sum_{i=1}^{n} -\ln(\sqrt{2\pi\sigma^2}) + (-\frac{1}{2\sigma^2}(x^{(i)} - \mu)^2) \\
&= \sum_{i=1}^{n} -\frac{1}{2} \ln(2\pi\sigma^2) + (-\frac{1}{2\sigma^2}(x^{(i)} - \mu)^2)
\end{aligned} \tag{7}$$

Next, we optimize w.r.t. μ:

$$\begin{aligned}
\nabla_\mu \ln \ell(X) &= \sum_{i=1}^{n} (\frac{1}{\sigma^2}(x^{(i)} - \mu)) \\
0 &= \frac{1}{\sigma^2} \sum_{i=1}^{n} ((x^{(i)} - \mu)) \\
0 &= \sum_{i=1}^{n} x^{(i)} - \sum_{i=1}^{n} \mu \\
0 &= \sum_{i=1}^{n} x^{(i)} - n\mu \\
n\mu &= \sum_{i=1}^{n} x^{(i)} \\
\mu &= \frac{1}{n} \sum_{i=1}^{n} x^{(i)}
\end{aligned} \tag{8}$$

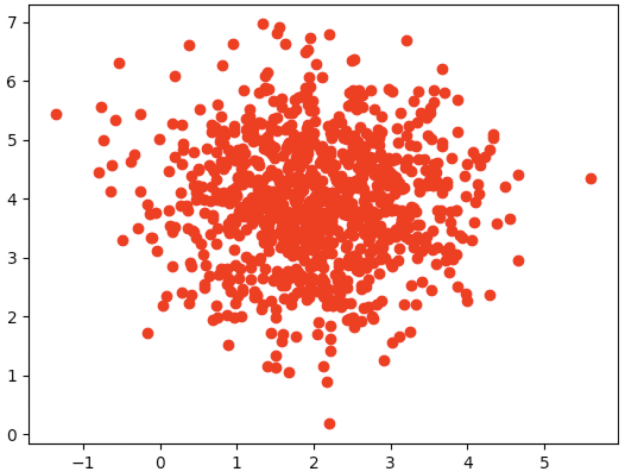
We achieve the intuitive result that the MLE estimate of the mean is just an average of the observations. Next, we optimize w.r.t. σ^2:

$$\begin{aligned}
\nabla_{\sigma^2} \ln \ell(X) &= \sum_{i=1}^{n} -\frac{1}{2\sigma^2} + \frac{(x^{(i)} - \mu)^2}{2(\sigma^2)^2} \\
0 &= \sum_{i=1}^{n} -\frac{1}{2\sigma^2} + \frac{(x^{(i)} - \mu)^2}{2(\sigma^2)^2} \\
0 &= \sum_{i=1}^{n} -\frac{1}{2\sigma^2} + \sum_{i=1}^{n} \frac{(x^{(i)} - \mu)^2}{2(\sigma^2)^2} \\
0 &= -\frac{n}{2\sigma^2} + \sum_{i=1}^{n} \frac{(x^{(i)} - \mu)^2}{2(\sigma^2)^2} \\
\frac{n}{2\sigma^2} &= \sum_{i=1}^{n} \frac{(x^{(i)} - \mu)^2}{2(\sigma^2)^2} \\
n &= \sum_{i=1}^{n} \frac{(x^{(i)} - \mu)^2}{\sigma^2} \\
\sigma^2 &= \sum_{i=1}^{n} \frac{(x^{(i)} - \mu)^2}{n}
\end{aligned} \tag{9}$$

3.1.1 Spherical Gaussian distribution

Let's say that we are given a bunch of data points in $\mathbb{R}^2$, shown below. You can assume that each data point were sampled i.i.d. We want to fit some probability distribution to these data points that best describes the data. By observation, we may notice that the data points look like they were sampled from a Spherical Gaussian distribution, since the points are distributed evenly around some center point. We want to determine the parameters of the Spherical Gaussian distribution such that the resulting distribution fits the data the best, *i.e.,* maximizes the likelihood.

This is a problem we will cover more thoroughly in lecture next week, but just to introduce it, the PDF of the Spherical Gaussian is as follows: $\mathcal{N}(\bar{x}|\bar{\mu}, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp^{[-\frac{1}{2}(\bar{x}-\bar{\mu})^T \Sigma^{-1}(\bar{x}-\bar{\mu})]}$.



Discussion Question 10. *Intuitively, what is the mean $\bar{\mu}$ of this distribution?*

Solution: Just by looking at the plot of points, we might guess that the mean of the distribution is just the center of all the points, which is at approximately $\bar{\mu} = [2, 4]$.