Collaborative Filtering: A Machine Learning Perspective

Chapter 6: Dimensionality Reduction

Benjamin Marlin

Presenter: Chaitanya Desai

Topics we'll cover

- Dimensionality Reduction for rating prediction:
  - Singular Value Decomposition
  - Principal Component Analysis

Rating Prediction

Problem Description:

- We have M distinct items and N distinct users in our corpus
- Let ${r_y}^u$ be rating assigned by user u for item y. Thus, we have (M dimensional) rating vectors $\boldsymbol{r}^u$ for the N users
- Task:
  - Given the partially filled rating vector $\boldsymbol{r}^a$ of an active user a, we want to estimate $\hat{r}_y^a$ for all items y that have not yet been rated by a

Singular Value Decomposition

- Given a data matrix D of size $N \times M$, the SVD of D is $D = U\Sigma V^T$ where $U_{N\times N}$ and $V_{M\times M}$ are orthogonal and $\Sigma_{N\times M}$ is diagonal
- columns of U are eigenvectors of DD^T and columns of V are eigenvectors of D^TD
- Σ is diagonal and entries comprise of eigenvalues ordered according to eigenvectors (i.e. columns of U and V)

Low Rank Approximation

- Given the solution to SVD, we know that $\hat{D} = U_k \Sigma_k {V_k}^T$ is the best rank k approximation to D under the Frobenius norm
- The Frobenius norm is given by

$$F(D - \hat{D}) = \sum_{n=1}^{N} \sum_{m=1}^{M} (D_{nm} - \hat{D}_{nm})^2$$

Weighted Low Rank Approximation

- D contains many missing values

Weighted Low Rank Approximation

- D contains many missing values
- SVD on a matrix is undefined if there are missing values

Weighted Low Rank Approximation

- D contains many missing values
- SVD on a matrix is undefined if there are missing values
- Solution?

Weighted Low Rank Approximation

- D contains many missing values
- SVD on a matrix is undefined if there are missing values
- Solution?
- Assign 0/1 weights to elements of D

Weighted Low Rank Approximation

- D contains many missing values
- SVD on a matrix is undefined if there are missing values
- Solution?
- Assign 0/1 weights to elements of D
- Our goal is to find $\hat{D}$ so that the **weighted** Frobenius norm is minimized

$$F_w(D - \hat{D}) = \sum_{n=1}^{N} \sum_{m=1}^{M} W_{nm}(D_{nm} - \hat{D}_{nm})^2$$

Weighted low rank approximation

- *Srebro* and *Jaakkola* in their paper *Weighted Low Rank Approximations* provide 2 approaches to finding $\hat{D}$
  - Numerical Optimization using Gradient Descent in U and V
  - Expectation Maximization (EM)

Generalized EM

Given a joint distribution $p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$ over observed variables $\boldsymbol{X}$ and latent variables $\boldsymbol{Z}$, governed by parameters $\boldsymbol{\theta}$, the goal is to maximize the likelihood function $p(\boldsymbol{X}|\theta)$ with respect to θ

1. Choose an initial setting for the parameter θ^{old}
2. **E step** Evaluate $p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$
3. **M step** Evaluate $\boldsymbol{\theta}^{new}$ given by $\boldsymbol{\theta}^{new} = \arg\max_\theta Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$ where $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_Z p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})\ ln\ p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$ represents the expectation of the complete data log-likelihood for some general parameter $\boldsymbol{\theta}$
4. If convergence criterion is not satisfied, let

$$\boldsymbol{\theta}^{old} \leftarrow \boldsymbol{\theta}^{new}$$

and return to step 2

EM for weighted SVD

- **E step**: Fill in the missing values of D from the low rank reconstruction $\hat{D}$ forming a complete matrix X.

(1) $$X = W \odot D + (1 - W) \odot \hat{D}$$

EM for weighted SVD

- **E step**: Fill in the missing values of D from the low rank reconstruction $\hat{D}$ forming a complete matrix X.

(1) $$X = W \odot D + (1 - W) \odot \hat{D}$$

- **M step**: Find low rank approximation using standard SVD on X, which is completely specified.

$$[U, \Sigma, V^T] = SVD(X)$$

$$\hat{D} = U_k \Sigma_k V_k^T$$

The complete algorithm

Input: R, W, L, K
Output: Σ, V

$\hat{R} \leftarrow 0$
while($F_W(R - \hat{R})$ not converged) **do**
  $X \leftarrow W \odot R + (1 - W) \odot \hat{R}$
  $[U, \Sigma, V^T] = SVD(X)$
  $U \leftarrow U_L, \Sigma \leftarrow \Sigma_L, V \leftarrow V_L$
  $\hat{R} \leftarrow U \Sigma V^T$
  if ($L > K$**) then**
    Reduce L
  end if
end while

ating Prediction using the new concept spa

- Given $\hat{R}$, rating value for user u for item y is simply $\hat{r}_y^u$ = $\hat{R}_{uy}$

...ting Prediction using the new concept spa...

- Given $\hat{R}$, rating value for user u for item y is simply $\hat{r}_y^u = \hat{R}_{uy}$
- Map the new user's profile into the K dimensional latent space

ting Prediction using the new concept spa

- Given $\hat{R}$, rating value for user u for item y is simply $\hat{r}_y^u$ = $\hat{R}_{uy}$
- Map the new user's profile into the K dimensional latent space
  - If r is the user's rating vector in the original space and l is the user's vector in the latent space, then $r = l\Sigma V^T$ and thus $l = rV\Sigma^{-1}$

Algorithm (rating vector estimation)

Input: $\boldsymbol{r}^a, w^a, \Sigma, V, K$
Output: $\hat{\boldsymbol{r}}^a$

$\hat{\boldsymbol{r}}^a \leftarrow 0$
while($F_{w^a}(r^a - \hat{\boldsymbol{r}}^a)$ not converged) **do**
    $x \leftarrow w^a \odot \boldsymbol{r}^a + (1 - w^a) \odot \hat{\boldsymbol{r}}^a$
    $\boldsymbol{l}^a \leftarrow xV\Sigma^{-1}$
    $\hat{\boldsymbol{r}}^a \leftarrow \boldsymbol{l}^a \Sigma V^T$
end while

Results of SVD

Data Sets:

- EachMovie:
  - Collected by the Compaq Systems Research Center over an 18 month period beginning in 1997.
  - Base data set contains 72916 users, 1628 movies and 2811983 ratings.
  - Ratings are on a scale from 1 to 6.
- MovieLens
  - Collected by GroupLens research group at the University of Minnesota
  - Contains 6040 users, 3900 movies, and 1000209 ratings collected from users who joined the MovieLens recommendation service in 2000
  - Ratings are on a scale from 1 to 5.

Results of SVD...

- Results reported are *NMAE*
- $NMAE = \frac{MAE}{E[MAE]}$
- $MAE = \frac{1}{N}\sum_{n=1}^{N} |\hat{r}_y^u - r_y^u|$
- Greater than 1 indicates worse than random

Principal Component Analysis

- The idea is to discover latent structure in the data
- Let $A = \frac{1}{N-1} D^T D$ be the co-variance matrix of D
- $A(i,j)$ indicates co-variance between items i and j
- We are interested in retaining K dimensions of highest variance
- This is the subspace spanned by the K largest eigenvectors of A

Rating Prediction with PCA

- Cannot do PCA when D has missing data
- Goldberg, Roeder, Gupta and Perkins propose an algorithm called **Eigentaste** in their paper *Eigentaste: A Constant Time Collaborative Filtering Algorithm*

Eigentaste

- Pick a set of items called the *gauge set* that all users must rate

Eigentaste

- Pick a set of items called the *gauge set* that all users must rate
- Map the new data into a lower dimensional latent space and retain only the 1st 2 components

Eigentaste

- Pick a set of items called the *gauge set* that all users must rate
- Map the new data into a lower dimensional latent space and retain only the 1st 2 components
- Cluster users in this 2 dimensional latent space using divisive hierarchical clustering

Eigentaste

- Pick a set of items called the *gauge set* that all users must rate
- Map the new data into a lower dimensional latent space and retain only the 1st 2 components
- Cluster users in this 2 dimensional latent space using divisive hierarchical clustering
- Compute mean rating vector μ_c for each cluster c

Eigentaste

- Pick a set of items called the *gauge set* that all users must rate
- Map the new data into a lower dimensional latent space and retain only the 1st 2 components
- Cluster users in this 2 dimensional latent space using divisive hierarchical clustering
- Compute mean rating vector μ_c for each cluster c
- For a new user, map the user's rating profile (r^a) for the gauge set into the 2 dimensional concept space ($\hat{r}^a$) and determine what cluster the user belongs to.

Eigentaste

- Pick a set of items called the *gauge set* that all users must rate
- Map the new data into a lower dimensional latent space and retain only the 1st 2 components
- Cluster users in this 2 dimensional latent space using divisive hierarchical clustering
- Compute mean rating vector μ_c for each cluster c
- For a new user, map the user's rating profile (r^a) for the gauge set into the 2 dimensional concept space ($\hat{r}^a$) and determine what cluster the user belongs to.
- If an item ($\hat{r}^a_y$) is not rated, assign it the mean rating vector's value for that item (i.e. $\hat{r}^a_y = \mu_{cy}$)

Problems with Eigentaste?

- Gauge Set must be the same for all users and all users must rate all gauge items

Problems with Eigentaste?

- Gauge Set must be the same for all users and all users must rate all gauge items
- Rating some items is easier and faster than others (e.g. jokes vs. books)

Problems with Eigentaste?

- Gauge Set must be the same for all users and all users must rate all gauge items
- Rating some items is easier and faster than others (e.g. jokes vs. books)
- Selection of items (items that are good discriminators)