# Using Collective Matrix Factorization and Tags to improve a Recommender System

Cristopher Arenas

PUC, Campus San Joaquín

November 25, 2015

# Introduction

- Recommender Systems are used to provide recommendation to users about items
- The problem consist in predict for an user $u$ the preference about an item $i$, tipically with a range of values $r$.

# Introduction

- There are methods based in Collaborative Filter to resolve the problem.
- This approach find similarities
  - Users
  - Items
  - Ratings

# Introduction

- One of the major problems with CF is the **sparsity problem**.
- The most of users rating a few items.
- Some Matrix Factorization methods are used to solve the sparsity problem using filling techniques.

# Introduction

- A method called Collective Matrix Factorization will be proposed.
- The use of tag information will be considered.

# Related Work

- Matrix Factorization Techniques consider the factorization of a matrix that relates users and items.
- Traditional approaches try to minimize the error function:

$$E = \sum_{(u,i)\in K} (r_{ui} - q_i^T p_u)^2 + \lambda(||q_i||^2 + ||p_u||^2) \tag{1}$$

# Proposed Method

There are three matrices that relates $m$ users, $n$ items and $p$ tags:

- $U(u, i)$: user-item matrix. Shows the *rating* of user $u$ for an item $i$.
- $T(u, t)$: user-tag matrix. Shows the *preference* of an user $u$ for the tag $t$.
- $G(i, t)$: tag-item matrix. Shows *relevance* between item $i$ and tag $t$.

# Proposed Method

- $U(u, i)$ and $G(i, t)$ are constructed directly with information.
- $T$ is constructed using $U$ and $G$.

$$T(u,t) = \frac{1}{N}\sum_{k=1}^{n} U(u,k) \times G(k,t) \tag{2}$$

# Proposed Method

- Later, submatrices $X$, $Y$ and $Z$ are used to construct two matrices $U'$ and $T'$.

$$U' = XY^T \tag{3}$$

$$T' = XZ^T \tag{4}$$

# Proposed Method

- Gradient Descent Method (GDM) is performed to minimize the error between real values and approximated values:

$$ER(X,Y,Z) = \frac{1}{2}||J \circ (U - XY^T)||_F^2 + \frac{\alpha}{2}||T - XZ^T||_F^2 + \frac{\beta}{2}\left(||X||_F^2 + ||Y||_F^2 + ||Z||_F^2\right) \tag{5}$$

- Gradients are calculated:

$$\nabla_X ER = \left[J \circ (XY^T - U)\right] Y + \alpha(XZ^T - T)Z + \beta X \tag{6}$$

$$\nabla_Y ER = \left[J \circ (XY^T - U)\right] X + \beta Y \tag{7}$$

$$\nabla_Z ER = \alpha(XZ^T - T)X + \beta Z \tag{8}$$

# Proposed Method

**Algorithm 1** Gradient Descent Method

Initialize $X$, $Y$, $Z$ with random number in range (0,1)
$t = 0$
**while** $t < max\_iteration$ **do**
  Get gradients $\nabla_X ER$, $\nabla_Y ER$ and $\nabla_Z ER$.
  $\gamma = 1$
  **while** $(ER(X_t - \gamma\nabla_{X_t}, Y_t - \gamma\nabla_{Y_t}, Z_t - \gamma\nabla_{Z_t}) > ER(X_t, Y_t, Z_t))$ **do**
    $\gamma = \gamma/2$
  **end while**
  $X_{t+1} = X_t - \gamma\nabla_{X_t}$
  $Y_{t+1} = Y_t - \gamma\nabla_{Y_t}$
  $Z_{t+1} = Z_t - \gamma\nabla_{Z_t}$
  $t = t + 1$
**end while**

# Experiments

- Dataset: MovieLens
  - 1.000.209 ratings
  - 3.682 movies
  - 6.040 users
- Tag-Genome
  - 9.734 movies
  - 1.128 tags
- 3.642 movies in MovieLens and Tag-Genome
- 80% training and 20% testing

# Experiments

- 3 experiments:
  1. Latent factors
  2. Prediction: MAE, RMSE
  3. Top-N: nDCG

# Results

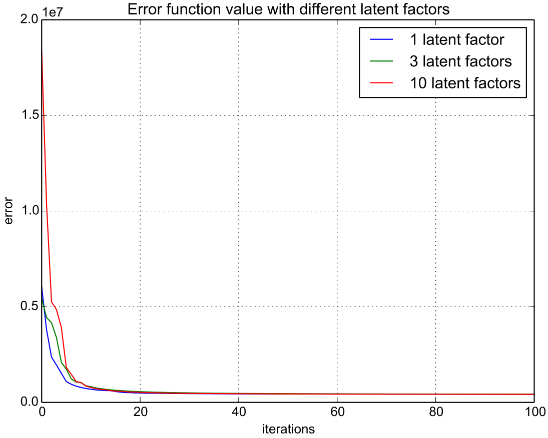


Figure 1 : Error function using three different latent factors.

# Results

Table 1 : MAE and RMSE considering two scenarios.

| Metric | With tags | Without tags |
|---|---|---|
| *MAE* | 0.7236 | 0.6961 |
| *RMSE* | 0.9264 | 0.8945 |

# Results

Table 2 : nDCG@p metric for two scenarios and differents values of p

| $p$ | Best user | | Worst user | | Average | |
|---|---|---|---|---|---|---|
| | With tags | Without tags | With tags | Without tags | With tags | Without tags |
| 1 | 0.7869 | 0.8354 | 0.7869 | 0.8354 | 0.7869 | 0.8354 |
| 3 | 0.7855 | 0.8323 | 0.6431 | 0.6581 | 0.7169 | 0.7337 |
| 5 | 0.7811 | 0.8256 | 0.4256 | 0.5164 | 0.6353 | 0.6740 |
| 10 | 0.9966 | 0.8666 | 0.4147 | 0.5002 | 0.6888 | 0.7157 |
| 20 | 0.9676 | 0.9658 | 0.3974 | 0.4682 | 0.6798 | 0.7066 |

# Conclusions

- Different ways to do matrix factorizations.
- Purpouse of this work: using tags in a collective matrix factorization.
- Latent factors
- About results.
- Future work: exaustive revision.