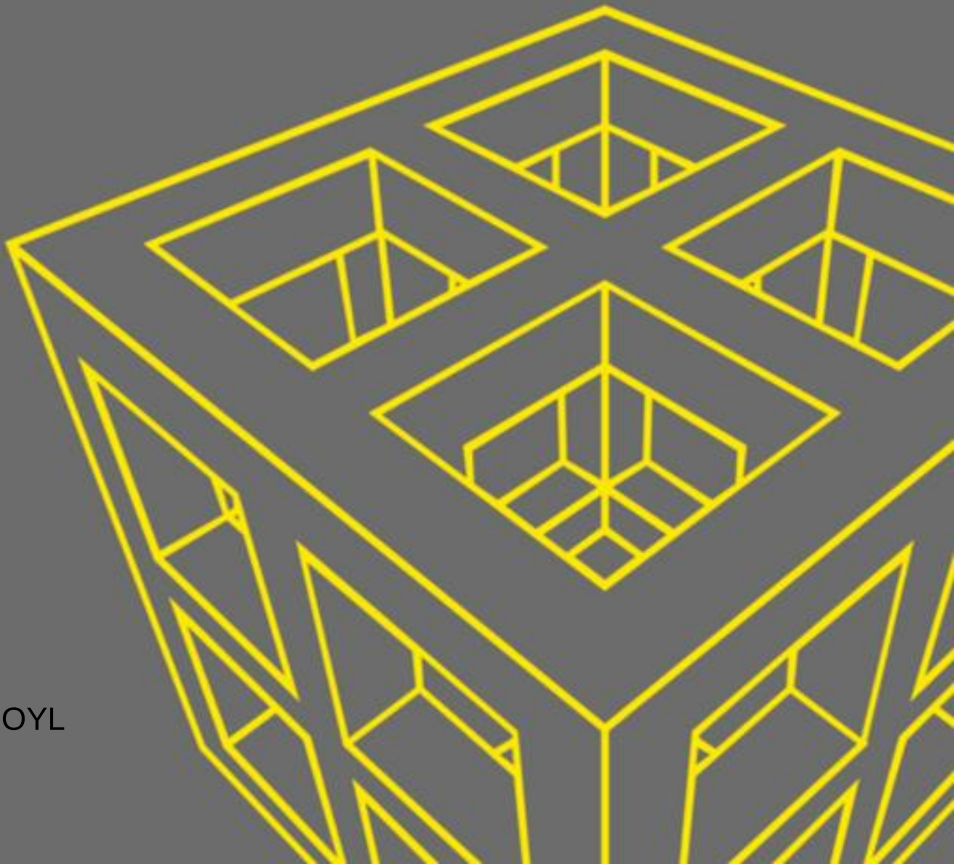
Advanced Topics Collaborative Filtering



By **Roy Levin**

http://researcher.watson.ibm.com/researcher/view.php?person=il-ROYL

Movie Recommendations

	Roy_1	$Inbal_2$	$Hasan_3$	$Lior_4$	$Anat_5$	$Arnon_6$
The God Father$_1$	?	4	?	5	?	?
The Dark Knight$_2$	3	?	?	?	2	5
Pulp Fiction$_3$	5	3	5	4	4	5
40 Year Old Virgin$_4$	2	4	?	?	3	3
Analyze That$_5$	3	5	4	?	4	?
Anger Management$_6$	3	5	?	?	?	5
Black Hawk Down$_7$	5	?	?	4	?	?

Romance	Action	Comedy	
f_1	f_2	f_3	
?	?	?	x_1
?	?	?	x_2
?	?	?	
?	?	?	
?	?	?	
?	?	?	
?	?	?	x_m

?

$$\theta_1 = \begin{matrix} 0 \\ 5 \\ 0 \end{matrix} \qquad \theta_2 = \begin{matrix} 5 \\ 0 \\ 0 \end{matrix} \qquad \cdots \qquad \theta_n = \begin{matrix} 0 \\ 0 \\ 5 \end{matrix}$$

Movie Recommendations

	Roy_1	$Inbal_2$	$Hasan_3$	$Lior_4$	$Anat_5$	$Arnon_6$
The God Father$_1$	?	4	?	5	?	?
The Dark Knight$_2$	3	?	?	?	2	5
Pulp Fiction$_3$	5	3	5	4	4	5
40 Year Old Virgin$_4$	2	4	?	?	3	3
Analyze That$_5$	3	5	4	?	4	?
Anger Management$_6$	3	5	?	?	?	5
Black Hawk Down$_7$	5	?	?	4	?	?

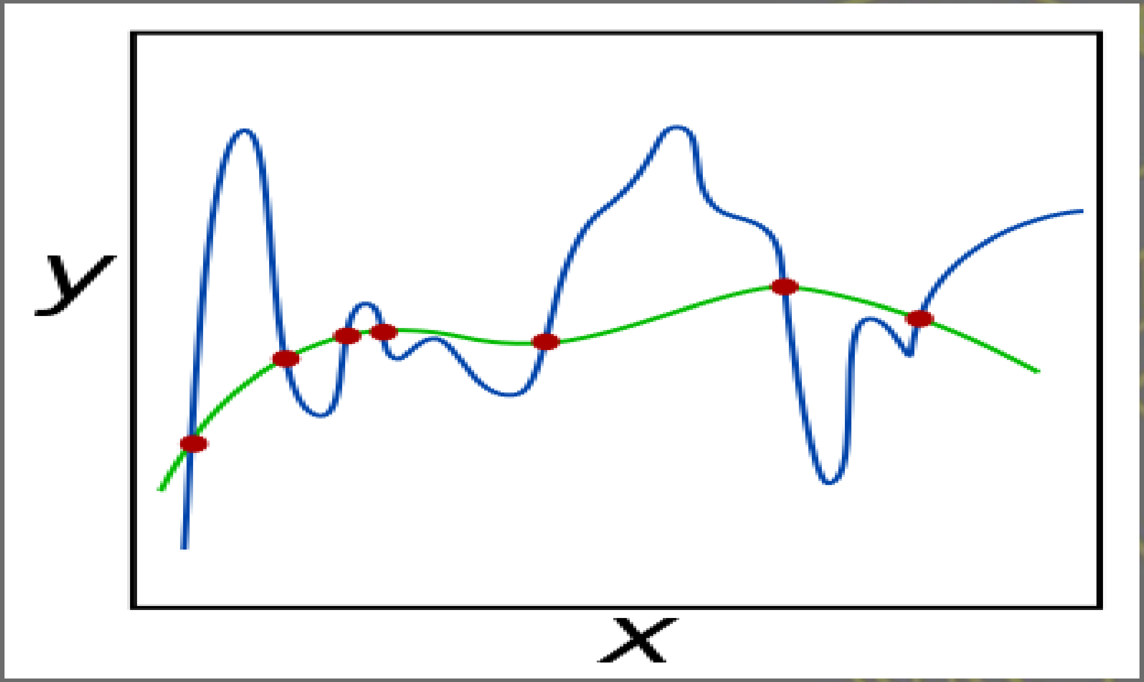
Romance	Action	Comedy	
f_1	f_2	f_3	
?	?	?	x_1
?	?	?	x_2
?	?	?	
?	?	?	
?	?	?	
1	0.6	1	
?	?	?	x_m

$$\theta_1 = \begin{matrix} 0 \\ 5 \\ 0 \end{matrix} \qquad \theta_2 = \begin{matrix} 5 \\ 0 \\ 0 \end{matrix} \qquad \cdots \qquad \theta_n = \begin{matrix} 0 \\ 0 \\ 5 \end{matrix}$$

Formalizing the Problem

Given $\theta^{(1)}, \dots, \theta^{(n_u)}$, to learn $x^{(i)}$:

$$\min_{x^{(i)}} \frac{1}{2} \sum_{j:r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2} \sum_{k=1}^{n} (x_k^{(i)})^2$$



Formalizing the Problem

- Given $\theta_1, \theta_2, \ldots, \theta_n$
- We want to learn all $x_1, x_2, \ldots, x_m$
  - $J(x_1, x_2, \ldots, x_m) =$

$$min_{x_1,x_2,\ldots,x_m} \sum_{i=1}^{m} \sum_{j:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2$$

- Or, alternatively:

Formalizing the Problem

- Or: given $x_1, x_2, \ldots, x_m$
- We want to learn all $\theta_1, \theta_2, \ldots, \theta_n$
  - $J(\theta_1, \theta_2, \ldots, \theta_n) =$

$$min_{\theta_1,\theta_2,\ldots,\theta_n} \sum_{j=1}^{n} \sum_{i:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{j=1}^{n} \sum_{k=1}^{l} \theta_{j,k}^2$$

ALS: Collaborative Filtering

- Minimize $J(x_1, x_2, \ldots, x_m)$ and then $J(\theta_1, \theta_2, \ldots, \theta_n)$

- $min_{x_1,x_2,\ldots,x_m} \sum_{i=1}^{m} \sum_{j:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2$
- $min_{\theta_1,\theta_2,\ldots,\theta_n} \sum_{j=1}^{n} \sum_{i:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{j=1}^{n} \sum_{k=1}^{l} \theta_{j,k}^2$

- *Alternating Least Squares (ALS):*
- $\boldsymbol{x \to \theta \to x \to \theta \to \ldots \to \theta \to x}$

After reaching convergence then for unrated pairs (user$_i$, movie$_j$) we set:

- **rating(i, j)=$\theta_i x_j$**

SGD: Collaborative Filtering

- Another solution is to combine the two objectives

- $$J(x_1, x_2, \dots, x_m, \theta_2, \dots, \theta_n) = min_{x_1, x_2, \dots, x_m, \theta_1, \theta_2, \dots, \theta_n} \sum_{(i,j):\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \left(\sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2 + \sum_{j=1}^{n} \sum_{k=1}^{l} \theta_{j,k}^2 \right)$$

To solve this Stochastic Gradient Descent (SGD) can be used

Matrix Factorization?

$$\theta^T_{n,k} = \begin{bmatrix} \overrightarrow{\theta_1^T} \\ \theta_2^T \\ \ldots \\ \theta_n^T \end{bmatrix}$$

User feature matrix

$$X_{m,k} = \begin{bmatrix} \overrightarrow{x_1} \\ \overrightarrow{x_2} \\ \ldots \\ \overrightarrow{x_m} \end{bmatrix}$$

Movie feature matrix

$$\boldsymbol{X\Theta^T} = \begin{bmatrix} (\boldsymbol{x_1} \cdot \boldsymbol{\theta_1^T}) & \cdots & (\boldsymbol{x_1} \cdot \boldsymbol{\theta_n^T}) \\ \vdots & \ddots & \vdots \\ (\boldsymbol{x_m} \cdot \boldsymbol{\theta_1^T}) & \cdots & (\boldsymbol{x_m} \cdot \boldsymbol{\theta_n^T}) \end{bmatrix} \approx \boldsymbol{Y_{mxn}} = \begin{bmatrix} \mathbf{5} & \cdots & ? \\ \vdots & ? & \vdots \\ \mathbf{0} & \cdots & \mathbf{3} \end{bmatrix}$$

Find X and Θ s.t. difference is minimized

Finding Related Movies

- For each movie $\boldsymbol{i} \in \{\mathbf{1}, \dots, \boldsymbol{m}\}$ we learned a feature vector $\boldsymbol{x_i} \in \mathbb{R}^{\boldsymbol{l}}$
  - e.g., $\mathrm{x_{i,1}} = \text{romantic}$, $\mathrm{x_{i,2}} = \text{action}$, $\mathrm{x_{i,3}} = \text{comedy}$
  - In practice, these features are commonly not easy to interpret
  - They do however represent the properties that best influence how users rank the movies

Finding Related Movies

- How to find a movie j related to movie i?
  - Find movies j such that $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ is minimized
  - Find top-k x_js which minimize this distance

Alternating Least Squares in Matrix Form

- We need to solve: $[X\Theta^T]_{mxn} \approx Y_{mxn}$
  - (leaving out regularization for simplicity now)

1. Set $X = \hat{X}$
2. Repeat until convergence:
   1. $\Theta^T = \left(X^T X\right)^{-1} X^T Y$
   2. $X = Y\Theta\left(\Theta^T \Theta\right)^{-1}$

- For $A : m \times n \mid m > n$ we have a left inverse: $\underbrace{\left(A^T A\right)^{-1} A^T}_{A_{\text{left}}^{-1}} A = I_n$
- For $A : m \times n \mid m < n$ we have a right inverse: $A \underbrace{A^T \left(AA^T\right)^{-1}}_{A_{\text{right}}^{-1}} = I_m$

User Cold Start Problem

θ_3=?

	Roy$_1$	Inbal$_2$	Hasan$_3$	Lior$_4$	Anat$_5$	Arnon$_6$
The God Father$_1$	?	4	?	5	?	?
The Dark Knight$_2$	3	?	?	?	2	5
Pulp Fiction$_3$	5	3	?	4	4	5
40 Year Old Virgin$_4$	2	4	?	?	3	3
Analyze That$_5$	3	5	?	?	4	?
Anger Management$_6$	4	5	?	?	?	5
Black Hawk Down$_7$	5	?	?	4	?	?

$$min_{x_1,x_2,\dots,x_m,\theta_1,\theta_2,\dots,\theta_n} \sum_{(i,j):\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \left(\sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2 + \sum_{j=1}^{n} \sum_{k=1}^{l} \theta_{j,k}^2 \right)$$

User Cold Start Problem

	Roy_1	$Inbal_2$	$Hasan_3$	$Lior_4$	$Anat_5$	$Arnon_6$
The God Father$_1$	?	4	? 0	5	?	?
The Dark Knight$_2$	3	?	? 0	?	2	5
Pulp Fiction$_3$	5	3	? 0	4	4	5
40 Year Old Virgin$_4$	2	4	? 0	?	3	3
Analyze That$_5$	3	5	? 0	?	4	?
Anger Management$_6$	4	5	? 0	?	?	5
Black Hawk Down$_7$	5	?	? 0	4	?	?

Mean Normalization

$$\mathbf{Y} = \begin{bmatrix} \mathbf{5} & \mathbf{3} & ? \\ \mathbf{4} & ? & ? \\ ? & \mathbf{3} & \mathbf{2} \end{bmatrix}$$

$$\boldsymbol{\mu} = \begin{bmatrix} \mathbf{4} \\ \mathbf{4} \\ \mathbf{2.5} \end{bmatrix} \rightarrow \boldsymbol{Y} = \begin{bmatrix} \mathbf{1} & \mathbf{-1} & ? \\ \mathbf{0} & ? & ? \\ ? & \mathbf{0.5} & \mathbf{-0.5} \end{bmatrix}$$

Now, each movie has an average rating of **zero**

How do we now make a prediction of the rating of user *j* on movie *i* ?

$$\theta_j^T \cdot x_i + \mu_i$$

CF Using Mean Normalization

	Roy$_1$	Inbal$_2$	Hasan$_3$	Lior$_4$	Anat$_5$	Arnon$_6$
The God Father$_1$	?	4	? 4.5	5	?	?
The Dark Knight$_2$	3	?	? 3.3	?	2	5
Pulp Fiction$_3$	5	3	? 4.2	4	4	5
40 Year Old Virgin$_4$	2	4	? 3	?	3	3
Analyze That$_5$	3	5	? 4	?	4	?
Anger Management$_6$	4	5	? 4.6	?	?	5
Black Hawk Down$_7$	5	?	? 4.5	4	?	?

Using Spark MLlib

```
// case Rating(userId, itemId, rate)

val ratings: RDD[Rating] = ...
val model = ALS.train(
  ratings,
  rank,
  numIterations,
  lamda = 0.01)

val rate: Double = model
  .predict(myUserId, someProductId)
```

More methods here …
Is this one useful for recommendation?

CF With Implicit Feedback

- In the **explicit** case, recall:
  - Given $\theta_1, \theta_2, \ldots, \theta_n$
  - We want to learn all $x_1, x_2, \ldots, x_m$
    - $J(x_1, x_2, \ldots, x_m) = $
      $$\min_{x_1, x_2, \ldots, x_m} \sum_{i=1}^{m} \sum_{j:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2$$
- Similarly in the **implicit** case:
    - $J(x_1, x_2, \ldots, x_m) = $
      $$\min_{x_1, x_2, \ldots, x_m} \sum_{i=1}^{m} \sum_{j:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2$$

Not rating anymore but rather 1 if streamed 0 else

CF With Implicit Feedback

- In the **explicit** case, recall:
  - Given $\theta_1, \theta_2, \ldots, \theta_n$
  - We want to learn all $x_1, x_2, \ldots, x_m$
    - $J(x_1, x_2, \ldots, x_m) =$

$$min_{x_1,x_2,\ldots,x_m} \sum_{i=1}^{m} \sum_{j:\exists r(i,j)} \left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2$$

- Similarly in the **implicit** case:
    - $J(x_1, x_2, \ldots, x_m) =$

$$min_{x_1,x_2,\ldots,x_m} \sum_{i=1}^{m} \sum_{j:\exists r(i,j)} \left(1 + \alpha \cdot r_{i,j}\right)\left(\theta_j^T \cdot x_i - y_{i,j}\right)^2 + \lambda \cdot \sum_{i=1}^{m} \sum_{k=1}^{l} x_{i,k}^2$$

Not rating anymore but rather 1 if streamed 0 else

References

- https://www.youtube.com/watch?v=B-JjbXNGnP4&list=PLnnr1O8OWc6ZYcnoNWQignIiP5RRtu3aS&index=3
- http://www.slideshare.net/MrChrisJohnson/collaborative-filtering-with-spark
- https://en.wikipedia.org/wiki/Regularization_(mathematics)
- http://www.slideshare.net/erikbern/collaborative-filtering-at-spotify-16182818/49-Learning_from_feedback_is_actually