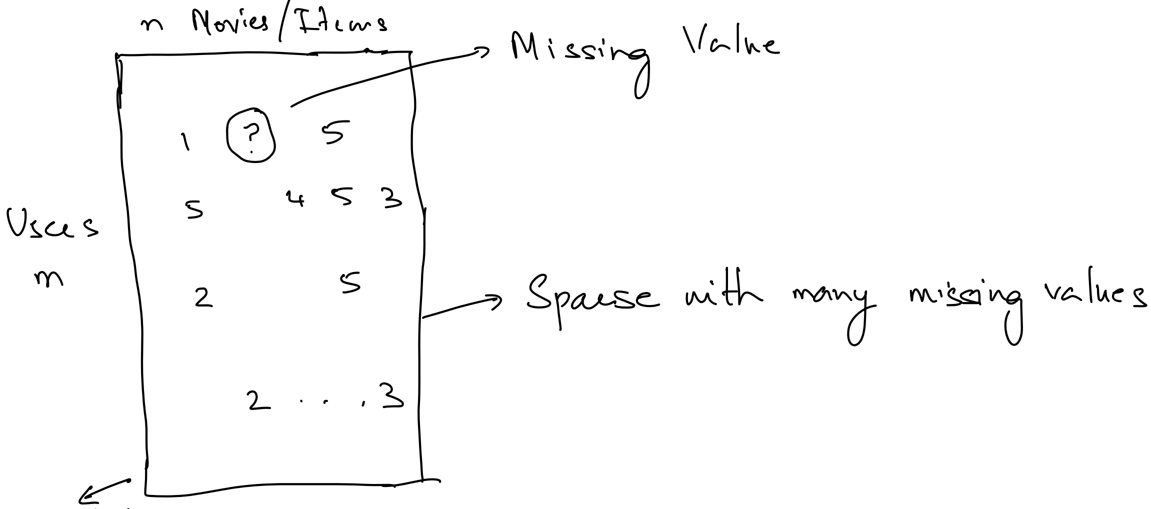
Missing Value Estimation

- Recommender Systems
- Matrix Completion / Matrix Factorization



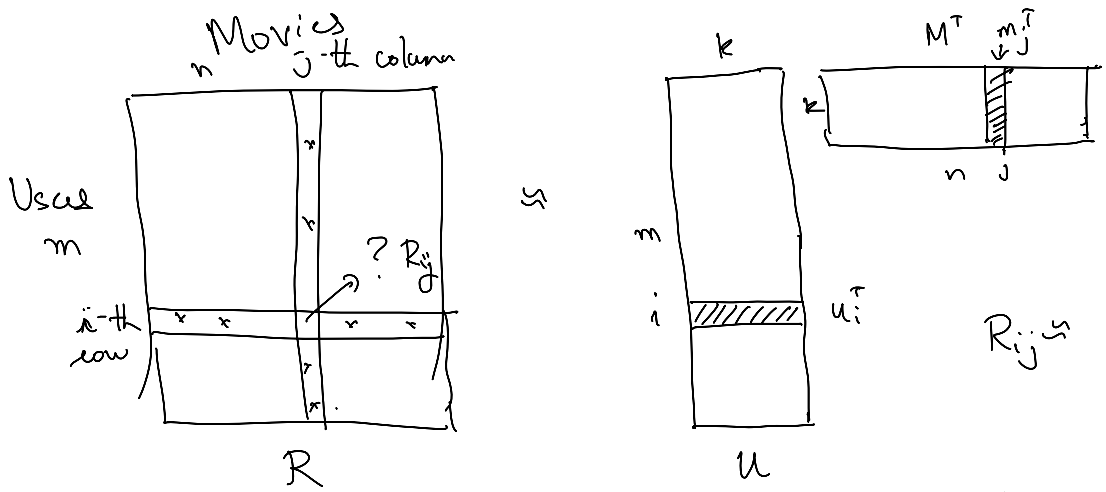
R : Ratings Matrix

<u>Goal</u>: Given ratings, predict missing values (recommendations)

<u>Initial methods</u>: Look at "neighbors" of a user

people who have similar ratings of items that a user has rated

Matrix Factorization for Recommendation :



$k << \min(m,n)$

$R_{ij} \approx u_i^T m_j$

UM^T is a rank-k matrix

$$R \approx UM^T$$

Row i of U is u_i^T : Low-dimensional (k) of user i

Row j of M is m_j^T : Low-dimensional (k) of movie j

Columns of U are latent factors (features) for users

Columns of M are latent factors (features) for movies

How to find u_i & m_j ?

$R_{ij} \approx u_i^T m_j$, K is set of all known ratings

$K = \{(i,j) : R_{ij} \text{ is known}\}$

$$\min_{\substack{u_1,u_2,\dots,u_m \\ m_1,m_2\dots m_n}} \boxed{\sum_{(i,j)\in K} \left(R_{ij} - u_i^T m_j\right)^2} + \lambda \sum_{i=1}^{m} \|u_i\|_2^2 + \lambda \sum_{i=1}^{n} \|m_j\|_2^2$$

→ Regularization

$$\|$$

$$\boxed{\min_{U,M} \|W \odot (R - UM^T)\|_F^2 + \lambda \|U\|_F^2 + \lambda \|M\|_F^2}$$

$$W_{ij} = \begin{cases} 1 & \text{if } (i,j) \in K \text{ (known ratings)} \\ 0 & \text{otherwise} \end{cases}$$

$\odot$ is the elementwise product (Hadamard product)

If K is the set of all (i,j) pairs

$$\min_{U,M} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(R_{ij} - u_i^T m_j\right)^2 \equiv \min_{U,M} \|R - UM^T\|_F^2$$

and the solution would be given by the k-truncated SVD.

But in our case, we only know ratings $(i,j) \in K$.

We do not know R_{ij} for all i, j , $1 \le i \le m$, $1 \le j \le n$

Alternating Minimization

Start with some U (as guess)

repeat K times:
- Fix U and compute M
- Fix M and compute U

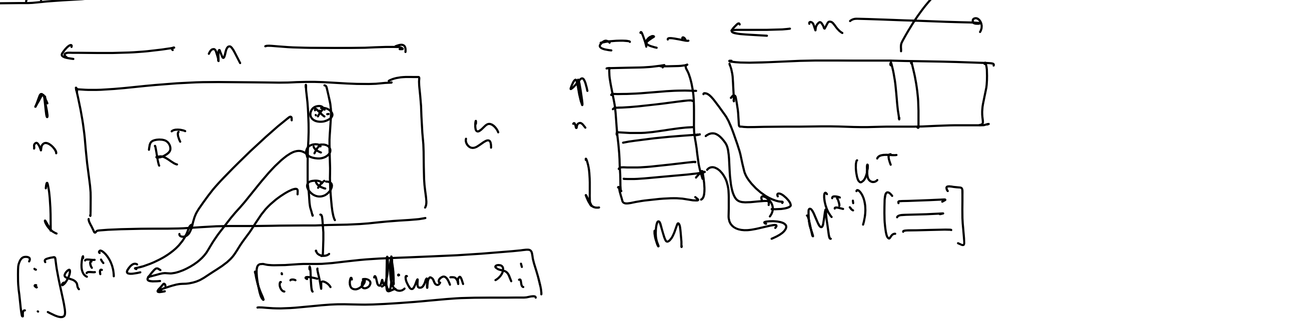
$U_0 \to M_1 \to U_1 \to M_2 \to U_2 \dots$

$$\boxed{M_i = \arg\min_{M} \|R - U_{i-1} M^T\|_F^2 + \lambda \|M\|_F^2}$$

$$U_i = \arg\min_{U} \|R - U M_i^T\|_F^2 + \lambda \|U\|_F^2$$

$R \approx UM^T \quad , \quad R^T \approx MU^T$

Suppose we fix M



$$r_i \approx M u_i$$

$$\min_{u_i} \| r_i - \cancel{M u_i} \|_2^2 + \lambda \| u_i \|_2^2$$ – Ridge Regression Problem

Index Set I_i represents indices of movies rated by user i

$$\min_{u_i} \left\{ J_i = \left(\frac{1}{2} \| r_i^{(I_i)} - M^{(I_i)} u_i \|_2^2 + \lambda \| u_i \|_2^2 \right) \right\}$$

$$\frac{\partial J_i}{\partial u_i} = -M^{(I_i)T}\left(r_i^{(I_i)} - M^{(I_i)} u_i\right) + \lambda u_i$$

$$\frac{\partial J_i}{\partial u_i} = 0 \Rightarrow \left(\underbrace{M^{(I_i)T} M^{(I_i)}}_{k \times k \text{ matrix}} + \lambda I\right) u_i = \underset{k \times |I_i|}{M^{(I_i)T}} \; \underset{|I_i|}{r_i^{(I_i)}}$$

Suppose $|I_i| = n_i$ $\qquad M^{(I_i)}$ is $|I_i| \times k = n_i \times k$

Forming this matrix requires $(k^2 n_i)$ operations

Solving the $k \times k$ linear system requires $O(k^3)$ operations

Total Complexity of finding U from M:

$$O\left(k^2 \left(\sum_i n_i\right)\right) = O(\text{No. of ratings} \cdot k^2) + O(mk^3)$$

Alternating Minimization / Least Squares

↪ Expensive (computation is cubic in k)

Other Alternatives : Stochastic Gradient Descent (SGD)

Coordinate Descent . . .