So Far…

- It's time for
  - Unsupervised learning
    - We are only given inputs
    - Goal: find "interesting patterns"
    - Discovering clusters
      - Clustering
    - Discovering latent factors
      - Dimensionality reduction
      - Topic modeling
      - Matrix factorization
    - Matrix completion

Matrix Completion

Deng Cai
Zhejiang University

Problem Formulation

- Low rank property of Matrix:
  - A basic characteristic for matrix analogy to the sparsity for vector.
  - It widely exists in the data matrix from real applications.

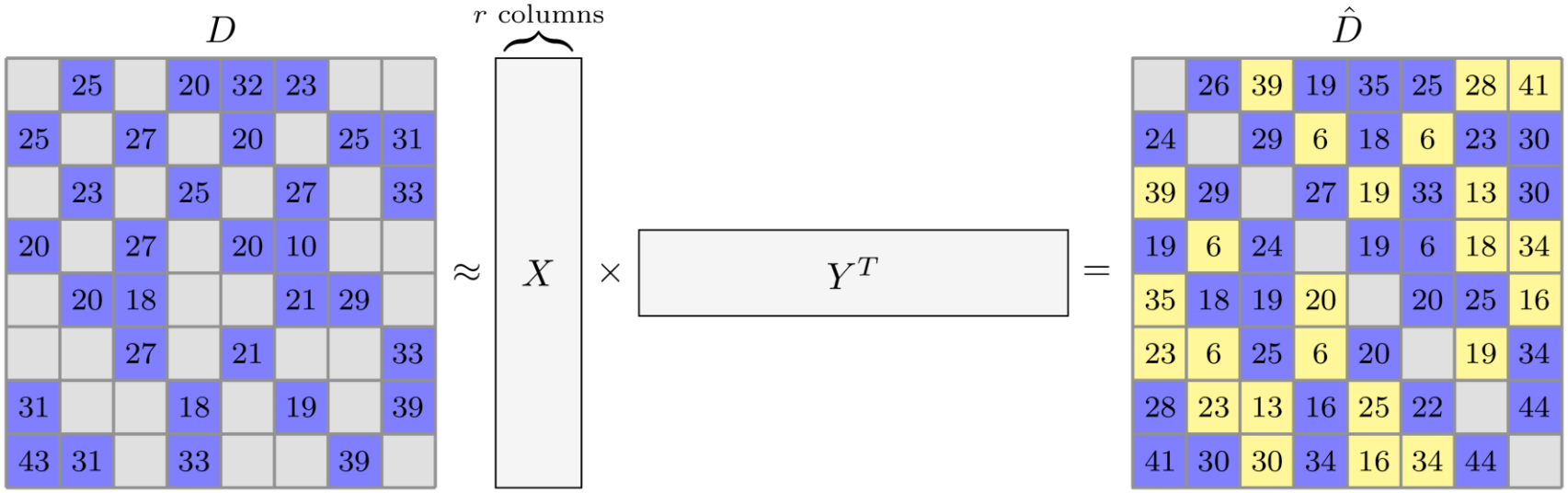
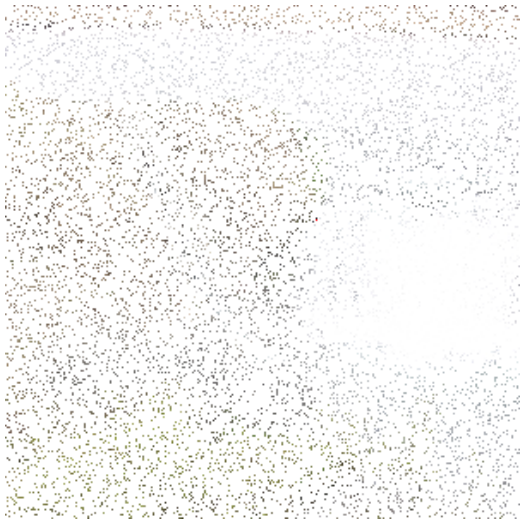


Image Recovery

- In many visual applications, we can only achieve the incomplete visual data. Natural image statistics have shown that images usually have low rank or approximately low rank structure. The missing information can be recovered based on low rank property of image matrix.

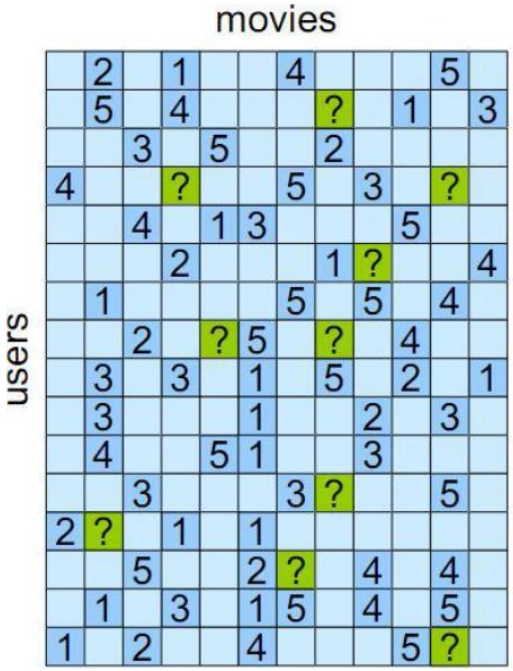


Incomplete Image

Recovered Image

Recommender System

- Recovering the unknown information from very limited observed information. Since the number of factors of the users' preference is limited, recommender system is regarded as low rank matrix recovery problem.



Netflix dataset

- 480,189 users, 17,770 movies, only about 100 million ratings.
- 480,189 X 17,770 matrix that is 99% sparse
- About 8.5 billion potential ratings

Objective function

- Let $M \in \mathbb{R}^{m \times n}$ be a partially observed matrix and $\Omega \subset \{1,2,\cdots,m\} \times \{1,2,\cdots,n\}$ be the set of indices of observed entries, then the goal is to recover X^* by solving the following rank minimization problem:

$$\min_X \operatorname{rank}(X)$$
$$\text{s.t. } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M),$$

where the projection operator $\mathcal{P}_\Omega(\cdot)$ is defined as

$$[\mathcal{P}_\Omega(X)]_{ij} = \begin{cases} X_{ij}, \text{if } (i,j) \in \Omega \\ 0, \text{otherwise} \end{cases}$$

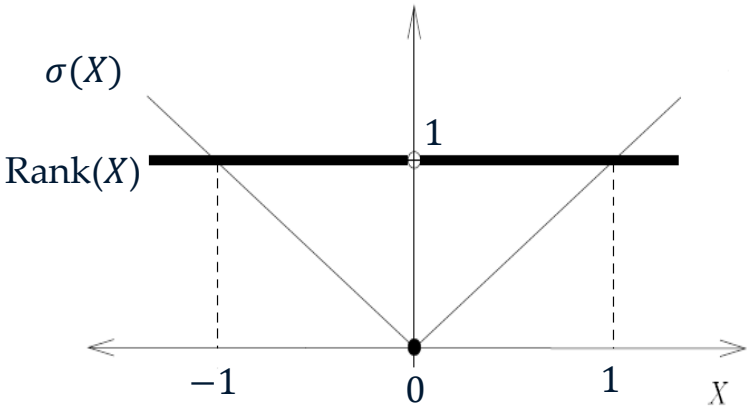
- However, since the rank function is non-convex and discrete, the above rank minimization problem is NP-hard.

Nuclear Norm and Rank

- For a matrix $X \in \mathbb{R}^{m \times n}$, define the nuclear norm of X as

$$\| X \|_* = \sum_{i=1}^{\max\{m,n\}} \sigma_i(X), \text{ where } \sigma_i(X)\text{'s are singular values of } X.$$

- It has been shown that $\| X \|_*$ is the tightest convex lower bound of $\text{Rank}(X)$. More precisely, on the set $\mathcal{S} = \{X \in \mathbb{R}^{m \times n} | \, ||X|| \leq 1\}$, $\| X \|_*$ is the largest convex function f such that $f(X) \leq \text{Rank}(X)$ for all $x \in \mathcal{S}$. Here $||X||$ is the spectral norm, i.e. the largest singular value of X.



For example, when X is a 1×1 (scalar) matrix, it has only one singular value $\sigma(X) = |X|$. $\text{Rank}(X)$=0 if X=0; and $\text{Rank}(X)$=1, otherwise. Then $\sigma(X)$ is the tightest convex lower bound of $\text{Rank}(X)$ for $|X| \leq 1$.

Nuclear Norm Minimizaitin

- By adopting nuclear norm as a surrogate of rank, the original rank minimization problem is converted into a nuclear norm minimization problem:

$$\min_X \operatorname{rank}(X) \quad \text{s.t. } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M) \qquad \Rightarrow \qquad \min_X \| X \|_* \quad \text{s.t. } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M)$$

non-convex convex

- In theory, Candes and Recht have shown that the low rank matrix can be exactly recovered with high probability by solving the above convex problem, even though only a small number of entries are observed.

Algorithms

- Many algorithms have been proposed to solve the nuclear norm minimization problem
  - Proximal Gradient Descent Method
  - Singular Value Thresholding (SVT)
  - Weighted (Truncated) Nuclear Norm Minimization
  - Rank One Pursuit

- We will focus on the Proximal Gradient Descent Method algorithm. Before that, we will introduce the singular value shrinkage operator.

Singular Value Shrinkage Operator

- Consider SVD of a matrix $X \in \mathbb{R}^{m\times n}$ of rank r in reduce form:

$$X = U\Sigma V^T,$$

where $U \in \mathbb{R}^{m\times r}$, $V \in \mathbb{R}^{n\times r}$, $\Sigma = \mathrm{diag}(\sigma_i(X)) \in \mathbb{R}^{r\times r}$,

- For $\tau > 0$, the singular value shrinkage operator is defined as

$$\mathcal{D}_\tau(X) = U\mathcal{D}_\tau(\Sigma)V^T, \mathcal{D}_\tau(\Sigma)=\mathrm{diag}(\max\{\sigma_i - \tau,0\})$$

This operator has the ability to "shrink" singular values towards zero, and the rank is possibly to be reduced

- The singular value shrinkage operator has the following property

Theorem. For $\tau > 0$ and $Y \in \mathbb{R}^{m\times n}$, the singular value shrinkage operator satisfies

$$\mathcal{D}_\tau(Y) = \underset{X}{\mathrm{argmin}} \frac{1}{2} \| X - Y \|_F^2 + \tau \| X \|_*$$

Proximal Gradient Descent

- The original problem is a constrained problem

$$\min_X \| X \|_* \quad \text{s.t. } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M) \tag{1}$$

- Let $F(X) = \lambda \| X \|_* + \| \mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M) \|_F^2$, PGD aims to solve the following unconstrained version

$$\min_X F(X) \tag{2}$$

- In this way, we allow some noises exist on the observed values. This relaxation is widely used in practical problems.
- But the problem (2) is still difficult to solve, since $\| X \|_*$ is not differentiable, so we cannot use gradient descent methods. But $g(X) = \| \mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M) \|_F^2$ is differentiable, and

$$\nabla g(X) = \mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)$$

Proximal Gradient Descent

- Then $g(X)$ near a given point X_k can be approximated as

$$g(X) \approx g(X_k) + \langle \nabla g(X_k), X - X_k \rangle + \frac{t_k}{2} \| X - X_k \|_F^2$$

$$= \frac{t_k}{2} \left(\| X - \left(X_k - \frac{1}{t_k} \nabla g(X_k)\right) \|_F^2 \right) + g(X_k) - \frac{1}{2t_k} \| \nabla g(X_k) \|_F^2$$

This part is irrelevant with X, and thus can be omitted

- Then

$$\begin{aligned}
&\operatorname*{argmin}_X F(X) \\
&\approx \operatorname*{argmin}_X \lambda \| X \|_* + \frac{t_k}{2} \left(\| X - \left(X_k - \frac{1}{t_k} \nabla g(X_k)\right) \|_F^2 \right) \\
&= \operatorname*{argmin}_X \frac{2\lambda}{t_k} \| X \|_* + \left(\| X - \left(X_k - \frac{1}{t_k} \nabla g(X_k)\right) \|_F^2 \right) \\
&\overset{*}{=} \mathcal{D}_{\frac{2\lambda}{t_k}} \left(X_k - \frac{1}{t_k} \nabla g(X_k) \right) \\
&= \mathcal{D}_{\frac{2\lambda}{t_k}} \left(X_k - \frac{1}{t_k} (\mathcal{P}_\Omega(X_k) - \mathcal{P}_\Omega(M)) \right)
\end{aligned}$$

The equation marked * is implied by $\mathcal{D}_\tau(X) = \operatorname*{argmin}_Y \frac{1}{2} \| X - Y \|_F^2 + \tau \| Y \|_*$

Proximal Gradient Descent

- Now we can compare the gradient descent(GD) method of minimizing $g(X)$ and the proximal gradient descent(PGD) method of minimizing $\lambda \| X \|_* + g(X)$:

$$\underset{X}{\operatorname{argmin}} g(X) \qquad\qquad \underset{X}{\operatorname{argmin}} \lambda \| X \|_* + g(X)$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$X_{k+1} = X_k - \frac{1}{t_k}\nabla g(X_k) \qquad\qquad X_{k+1} = \mathcal{D}_{\frac{2\lambda}{t_k}}\left(X_k - \frac{1}{t_k}\nabla g(X_k)\right)$$

gradient descent — proximal gradient descent

- We can see that the existence of the term $\lambda \| X \|_*$ induces additional step of "shrinkage" after standard gradient descent step in each iteration. In this way, the rank of X_{k+1} is very likely smaller than $X_k - \frac{1}{t_k}\nabla g(X_k)$. Thus we can get a rank descent sequence $\{X_k\}$.

Determination of Step Size

- The choice of step size is very important to the convergent rate of PGD. In principle, the step size $\frac{1}{t_k}$ of PGD should be chosen in descending order.
- Some researches have propose a simple strategy: given an initial step size $\frac{1}{t_0}$, we decrease this step size with a multiplicative factor $\gamma < 1$ repeatedly until certain condition is satisfied , and then keep the step size unchanged in following iterations. Since the condition is too complicated, we omit it here.
- In this way, the obtained sequence $\{X_k\}$ will converge to the optimal X^* of $F(X)$ at convergent rate $O(\frac{1}{k})$, which means there exists some constant $L > 0$, such that

$$F(X_k) - F(X^*) \leq \frac{\gamma L \parallel X_k - X^* \parallel_F^2}{2k}$$

For any $k \geq 1$.

Thanks!