FSAN/ELEG815: Statistical Learning

Gonzalo R. Arce

Department of Electrical and Computer Engineering
University of Delaware

2b. Matrix Completion

Outline

Matrix Approximations and Completion

Given an $m \times n$ matrix $\mathbf{Z} = \{z_{ij}\}$, find a matrix $\hat{\mathbf{Z}}$ that approximates $\mathbf{Z}$.

- $\hat{\mathbf{Z}}$ may have simpler structure.
- Missing entries in $\mathbf{Z}$, a problem known as *matrix completion*.

Approach based on optimization:

$$\hat{\mathbf{Z}} = \arg \min_{\mathbf{M} \in \mathbb{R}^{m \times n}} ||\mathbf{Z} - \mathbf{M}||_F^2 \text{ subject to } \Phi(\mathbf{M}) \leq c \qquad (1)$$

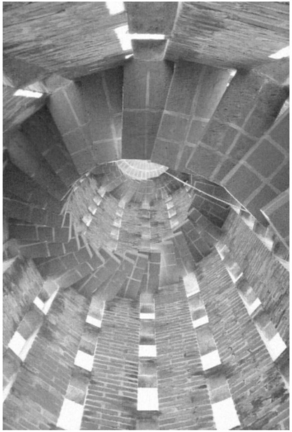
where $||\mathbf{A}||_F^2 = \sum \sum_{i,j} |a_{ij}|^2$ is the Frobenius Norm, and $\Phi(\cdot)$ is a constraint function that encourages $\hat{\mathbf{Z}}$ to be sparse in some sense.

Constraint $\Phi(\mathbf{Z})$	Resulting method
(a) $\|\|\hat{\mathbf{Z}}\|\|_{\ell_1} \leq c$	Sparse matrix approximation
(b) rank$(\hat{\mathbf{Z}}) \leq k$	Singular value decomposition
(c) $\|\|\hat{\mathbf{Z}}\|\|_* \leq c$	Convex matrix approximation

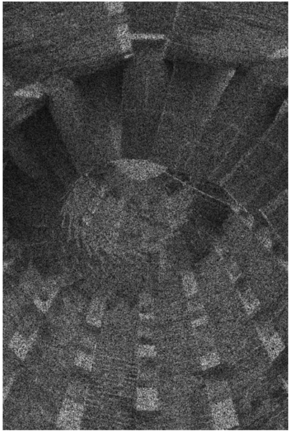
- (a) ℓ_1-norm of all entries of $\hat{\mathbf{Z}}$. Leads to a soft-thresholding $\hat{z}_{ij} = \text{sign}(z_{ij})(|z_{ij}| - \gamma)_+$, where $\gamma > 0$ is such that $\sum_{i=1}^{m} \sum_{j=1}^{n} |\hat{z_{ij}}| = c$.
- (b) Bounds the rank of $\hat{\mathbf{Z}}$, or the number of nonzero singular values in $\hat{\mathbf{Z}}$. Approximation is non-convex, but solution found by computing the SVD and truncating it to its top k components.
- (c) Relaxes the rank constraint to a *nuclear norm* ($||\mathbf{A}||_* = \sum_{i=1}^{min\{m,n\}} \sigma_i$). Solved by computing the SVD and soft-thresholding its singular values.

Motivation: Image Reconstruction from Incomplete Data

Reconstructed image



Incomplete image 50% of the pixels



Matrices with missing elements can be solved exactly using method (c), whereas methods based on (b) are more difficult to solve in general.

Constraint	Resulting method
$\hat{\mathbf{Z}} = \mathbf{L} + \mathbf{S}$, $\Phi_1(\mathbf{L}) \leq c_1$, $\Phi_2(\mathbf{S}) \leq c_2$	Additive matrix decomposition

- Seeks an additive decomposition of the matrix, imposing penalties on both components in the sum.

The Singular Value Decomposition

Given an $m \times n$ matrix $\mathbf{Z}$ with $m \geq n$, its *singular value decomposition* takes the form

$$\mathbf{Z} = \mathbf{U}\mathbf{D}\mathbf{V}^T \tag{2}$$

- $\mathbf{U}$ is an $m \times n$ orthogonal matrix ($\mathbf{U}^T\mathbf{U} = \mathbf{I}_n$) whose columns $\mathbf{u}_j \in \mathbb{R}^m$ are the *left singular vectors*.
- $\mathbf{V}$ is an $n \times n$ orthogonal matrix ($\mathbf{V}^T\mathbf{V} = \mathbf{I}_n$) whose columns $\mathbf{v}_j \in \mathbb{R}^n$ are the *right singular vectors*.
- The $n \times n$ matrix $\mathbf{D}$ is diagonal, with $d_1 \geq d_2 \geq \cdots \geq d_n \geq 0$ known as the *singular values*.

The Singular Value Decomposition

- If columns of $\mathbf{Z}$ are centered (zero mean), then the right singular vectors $\{\mathbf{v}_j\}_{j=1}^{n}$ define the *principal components* of $\mathbf{Z}$.
- The unit vector $\mathbf{v}_1$ yields the linear combination $\mathbf{s}_1 = \mathbf{Z}\mathbf{v}_1$ with highest sample variance among all possible choices of unit vectors.
- $\mathbf{s}_1$ is the *first principal component* of $\mathbf{Z}$, and $\mathbf{v}_1$ is the corresponding *direction* or *loading* vector.

The Singular Value Decomposition

Suppose $r \leq \text{rank}(\mathbf{Z}) = 800$, and let $\mathbf{D}_r$ be a diagonal matrix with all but the first r diagonal entries of $\mathbf{D}$ set to zero. The optimization problem

$$\hat{\mathbf{Z}}_r = \min_{\text{rank}(M)=r} ||\mathbf{Z} - \mathbf{M}||_F \tag{3}$$

has a closed form solution $\hat{\mathbf{Z}}_r = \mathbf{U}\mathbf{D}_r\mathbf{V}^T \triangleq$ the rank-r SVD. $\hat{\mathbf{Z}}_r$ is sparse in the sense that all but r singular values are zero.

800 Singular Values

164 Singular Values


24 Singular Values


12 Singular Values

Matrix Completion

Problem Formulation: Recover an $m \times n$ matrix $\mathbf{Z}$ when we only get to observe $p \ll mn$ of its entries.

- Impossible without additional information!
- Assumption: Matrix is known to be low-rank or approximately low-rank.
- Matrix Completion: Fill the missing entries.
- Used in: machine learning, computer vision...

Optimization Problem

- Observe the entries of the $m \times n$ matrix $\mathbf{Z}$ indexed by the subset $\Omega \subset \{1, \cdots, m\} \times \{1, \cdots, n\}$.
- Seek the lowest rank approximating matrix $\hat{\mathbf{Z}}$ that interpolates the entries of $\mathbf{Z}$

$$\begin{aligned} &\text{minimize} \quad \operatorname{rank}(\mathbf{M}) \\ &\text{subject to} \quad m_{ij} = z_{ij},\ (i,j) \in \Omega, \end{aligned} \tag{4}$$

- Rank minimization problem is NP-hard.
- Forcing interpolation leads to overfitting.

Optimization Problem

- Better to allow **M** to make some errors on the observed data:

$$\begin{aligned} \text{minimize} \quad & \operatorname{rank}(\mathbf{M}) \\ \text{subject to} \quad & \sum_{(i,j)\in\Omega} (z_{ij} - m_{ij})^2 \leq \delta\,, \end{aligned} \tag{5}$$

or equivalently

$$\underset{\operatorname{rank}(\mathbf{M})\leq r}{\text{minimize}} \quad \sum_{(i,j)\in\Omega} (z_{ij} - m_{ij})^2\,, \tag{6}$$

- Both problems are non-convex, and exact solutions are generally not available.

Matrix Completion Using the Nuclear Norm

- Nuclear norm of $\mathbf{M}_{m\times n}$:

$$||\mathbf{M}||_* = \sum_{k=1}^{n} \sigma_k(\mathbf{M}) \tag{7}$$

- Convex relaxation of the rank minimization problem:

$$\begin{aligned} &\text{minimize} && ||\mathbf{M}||_* \\ &\text{subject to} && m_{ij} = z_{ij},\ (i,j) \in \Omega\ , \end{aligned} \tag{8}$$

- Whereas the rank counts the number of nonzero singular values, the nuclear norm sums their amplitude.
- Analogous to the ℓ_1 norm as a relaxation for the ℓ_0 norm as sparsity measure.

Notation

Given an observed subset Ω of matrix entries, define the projection operator as:

$$[P_{\Omega}(\mathbf{Z})]_{i,j} = \begin{cases} z_{ij} & if \quad (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

P_{Ω} replaces the missing entries in $\mathbf{Z}$ with zeros, and leaves the observed entries alone.
The optimization criterion is then :

$$\sum_{(i,j)\in\Omega} (z_{ij} - m_{ij})^2 = ||P_{\Omega}(\mathbf{Z}) - P_{\Omega}(\mathbf{M})||_F \qquad (9)$$

where $||\cdot||_F$ is the Frobenius norm of a matrix defined as the element-wise sum of squares.

Singular Value Thresholding for Matrix Completion,[+]

- Solves the optimization problem:

$$\begin{array}{ll} \text{minimize} & ||\mathbf{M}||_* \\ \text{subject to} & P_\Omega(\mathbf{M}) = P_\Omega(\mathbf{Z}) \ , \end{array} \tag{10}$$

- The SVD of a matrix **M** of rank r is:

$$\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \ , \ \boldsymbol{\Sigma} = \mathrm{diag}(\{\sigma_i\}_{1\leq i\leq r}) \tag{11}$$

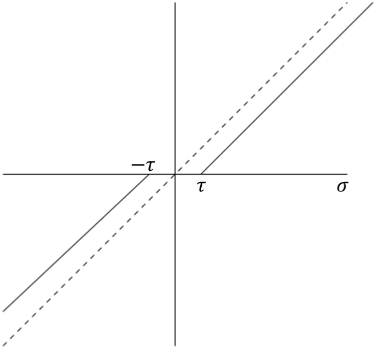
[+] Cai et al. (2010), SIAM Journal on Optimization, Vol. 20, No. 4

Singular Value Thresholding (SVT)

- For each $\tau \geq 0$, the soft-thresholding operator D_τ is defined as:

$$D_\tau(\mathbf{M}) = \mathbf{U} D_\tau(\mathbf{\Sigma}) \mathbf{V}^T \ , \ D_\tau(\mathbf{\Sigma}) = \mathrm{diag}(\mathrm{sgn}(\sigma_i)\{|\sigma_i| - \tau\}_+) \qquad (12)$$

where t, $t_+ = \max(0, t)$. Operator applies soft-thresholding to the singular values of **M**, effectively shrinking them towards zero.



SVT Algorithm - Shrinkage Iterations

Fix $\tau > 0$ and a sequence $\{\delta_k\}$ of positive step sizes. Starting with $\mathbf{Y}^0 = \mathbf{0}$, inductively define for $k = 1, 2, \ldots,$

$$\begin{cases} \mathbf{M}^k = D_\tau(\mathbf{Y}^{k-1}) \\ \mathbf{Y}^k = \mathbf{Y}^{k-1} + \delta_k P_\Omega(\mathbf{Z} - \mathbf{M}^k) \end{cases}$$

$$\begin{aligned} \mathbf{M}^1 &= D_\tau(\mathbf{Y}^0) = 0 \\ \mathbf{Y}^1 &= 0 + \delta_1 P_\Omega(\mathbf{Z} - 0) \\ &= \delta_1 P_\Omega(\mathbf{Z}) \end{aligned}$$

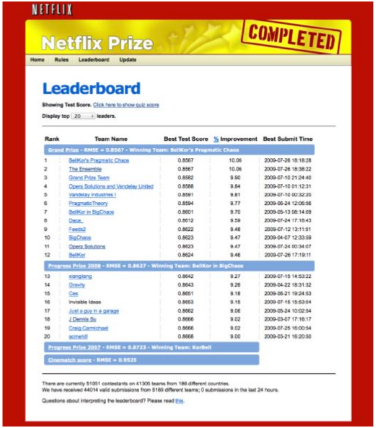
$$\begin{aligned} \mathbf{M}^2 &= D_\tau(\mathbf{Y}^1) = \delta_1 P_\Omega(\mathbf{Z}) \\ \mathbf{Y}^2 &= \delta_1 P_\Omega(\mathbf{Z}) + \delta_1 P_\Omega(\mathbf{Z} - \delta_1 P_\Omega(\mathbf{Z})) \end{aligned}$$

until a stopping criterion is reached. At each step, we only need to compute an SVD and perform elementary matrix operations.

SVT Algorithm - Shrinkage Iterations

Netflix Movie Challenge - Revisited

- Dataset: $n = 17,770$ movies (columns) and $m = 480,189$ customers (rows).
- Customers rated movies on a scale from 1 to 5. Matrix is very sparse with "only" 100 million of the ratings present in the training set.
- Goal: Predict the ratings for unrated movies.



NETFLIX

Netflix Prize — COMPLETED

Home | Rules | Leaderboard | Update

Leaderboard

Showing Test Score. Click here to show quiz score

Display top 20 leaders.

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries !	0.8591	9.81	2009-07-10 00:32:20
6	PragmaticTheory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BigChaos	0.8601	9.70	2009-05-13 08:14:09
8	Dace_	0.8612	9.59	2009-07-24 17:18:43
9	Feeds2	0.8622	9.48	2009-07-12 13:11:51
10	BigChaos	0.8623	9.47	2009-04-07 12:33:59
11	Opera Solutions	0.8623	9.47	2009-07-24 00:34:07
12	BellKor	0.8624	9.46	2009-07-26 17:19:11
Progress Prize 2008 - RMSE = 0.8627 - Winning Team: BellKor in BigChaos				
13	xiangliang	0.8642	9.27	2009-07-15 14:53:22
14	Gravity	0.8643	9.26	2009-04-22 18:31:32
15	Ces	0.8651	9.18	2009-06-21 19:24:53
16	Invisible Ideas	0.8653	9.15	2009-07-15 15:53:04
17	Just a guy in a garage	0.8662	9.06	2009-05-24 10:02:54
18	J Dennis Su	0.8666	9.02	2009-03-07 17:16:17
19	Craig Carmichael	0.8666	9.02	2009-07-25 16:00:54
20	acmehill	0.8668	9.00	2009-03-21 16:20:50
Progress Prize 2007 - RMSE = 0.8723 - Winning Team: KorBell				
Cinematch score - RMSE = 0.9525				

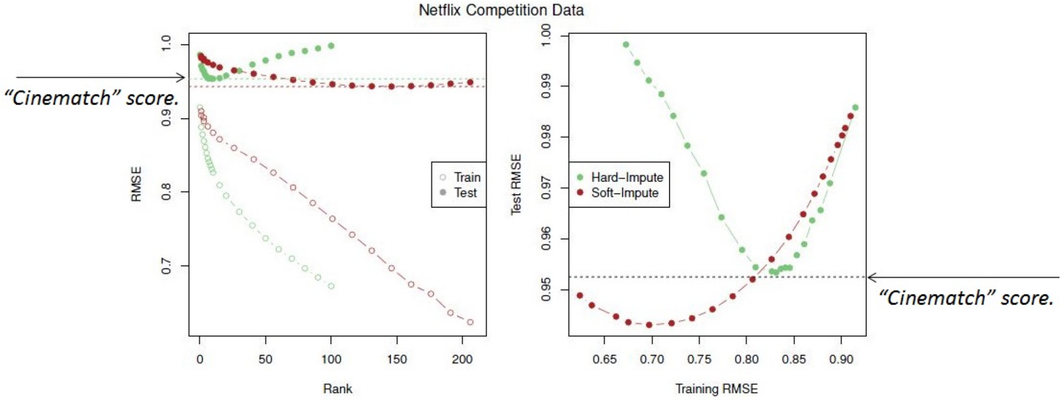
There are currently 51051 contestants on 41305 teams from 186 different countries.
We have received 44014 valid submissions from 5169 different teams; 0 submissions in the last 24 hours.

Questions about interpreting the leaderboard? Please read this.

- (2006) "Cinematch" algorithm used by Netflix RMSE=0.9525 over a large test set.
- Competition started in 2006, winner should improve this RMSE by at least 10%.
- 2009 "Bellkor's Pragmatic Chaos," uses a combination of many statistical techniques to win.

Netflix Movie Challenge

(Left) RMSE over the training and test sets as the rank of the SVD was varied (Hard-impute). Also estimates based on nuclear norm regularization (soft-impute). (Right) Test error only, plotted against training error, for the two methods.



SVT Algorithm Solution

Viewers rated movies on a scale from 1 to 5, 0 for movies that were not rated by the user.

- Each column j is a different movie
- Each row i is a different viewer
- Each element $a_{i,j}$ represents the rating of movie j by viewer i

	Movie 1	Movie 2	Movie 3	Movie 4	Movie 5
Viewer 1	0	1	0	0	5
Viewer 2	4	2	0	0	0
Viewer 3	0	0	3	3	0
Viewer 4	4	2	0	0	0
Viewer 5	0	0	0	0	5
Viewer 6	0	0	3	3	0
Viewer 7	1	0	0	0	4
Viewer 8	2	1	0	0	4
Viewer 9	1	0	0	0	4

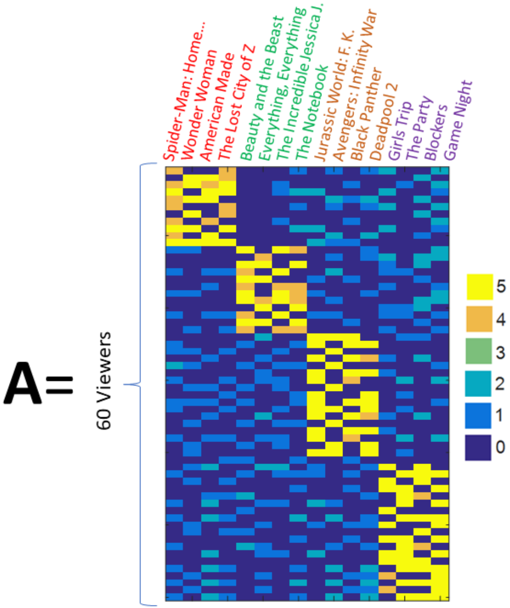
$$\mathbf{A} = \begin{bmatrix} a_{1,1} & \cdots & \cdots & a_{1,n} \\ \vdots & \ddots & \ddots & \vdots \\ a_{m,1} & \cdots & \cdots & a_{m,n} \end{bmatrix}$$

Goal: Use SVT algorithm to predict unobserved data or the rating of a movie that hasn't come out yet.

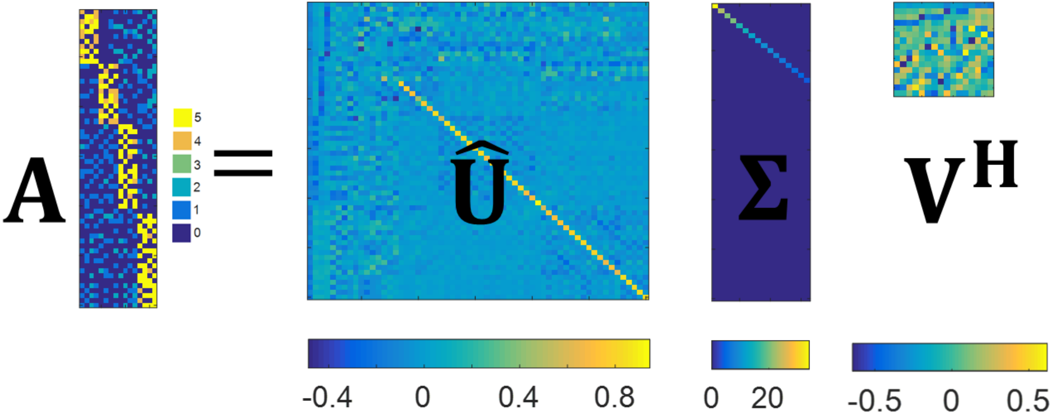
SVT Algorithm Solution

Considering the rating from 60 viewers to 16 movies of 4 different genres(action, romance, sci-fi, comedy), we generate $\mathbf{A} \in \mathbb{R}^{60\times 16}$

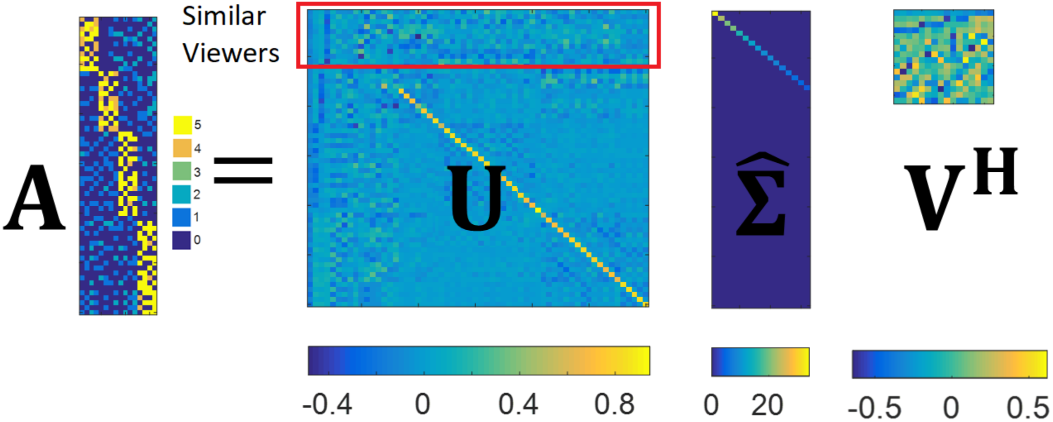
- Viewers rated movies on a scale from 1 to 5, 0 for movies that were not rated by the user.
- Observe the same 4 categories of viewers.



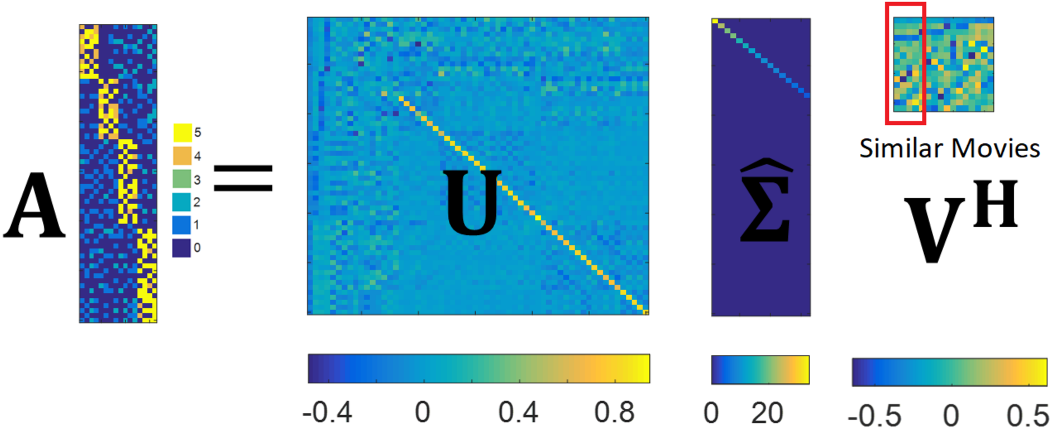
SVT Algorithm Solution



SVT Algorithm Solution



SVT Algorithm Solution



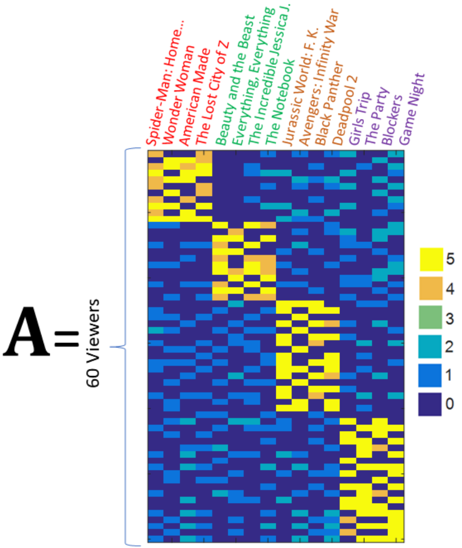
SVT Algorithm Solution

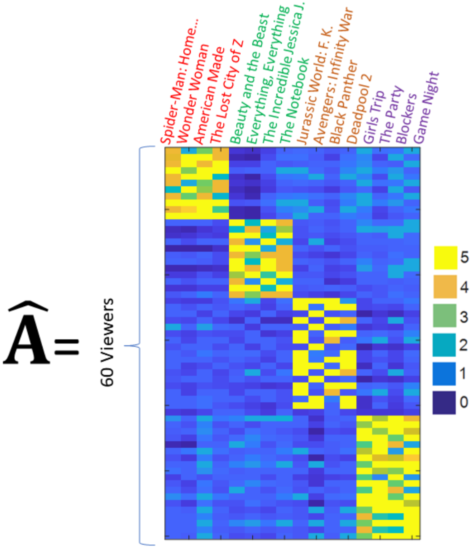
Use SVT Algorithm to estimate not rated movies (zero entries in $\mathbf{A}$), solving the optimization problem:

$$\begin{aligned} &\text{minimize} \quad ||\hat{\mathbf{A}}||_* \\ &\text{subject to} \quad P_\Omega(\hat{\mathbf{A}}) = P_\Omega(\mathbf{A}) \ , \end{aligned}$$

Note: The ratings matrix $\mathbf{A}$ is expected to be low-rank since user preferences can be described by a few categories (k), such as the movie genres.

SVT Algorithm Solution





SVT Algorithm Solution

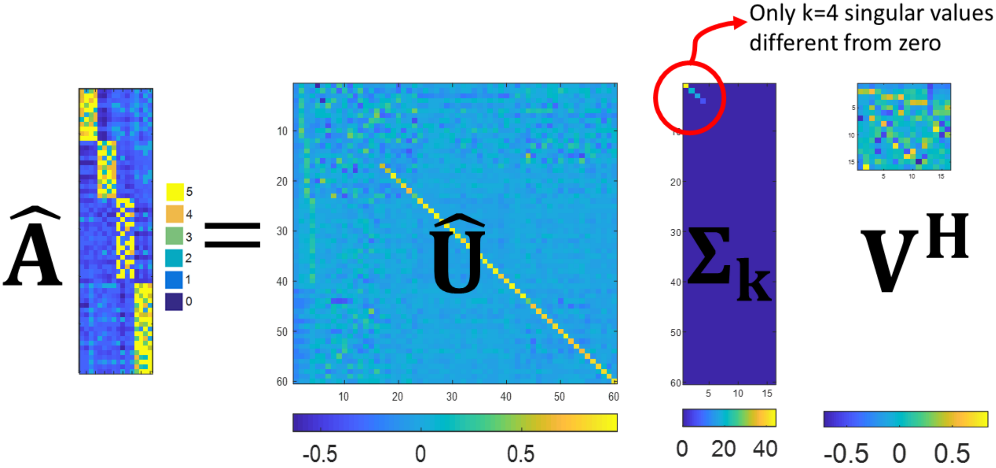


Image Inpainting - Convex Optimization Solver

With 70% of the Information.

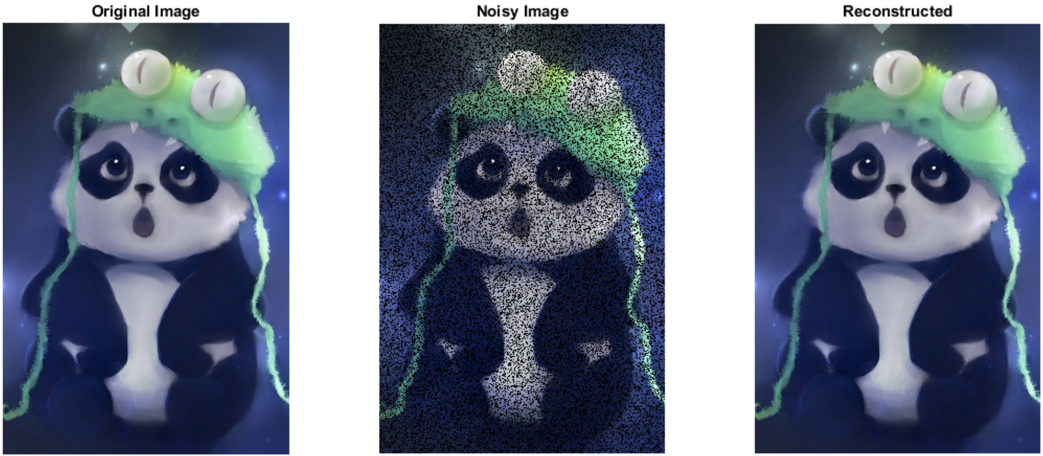


Image Inpainting - Convex Optimization Solver

With 50% of the Information. And multiple columns missing.

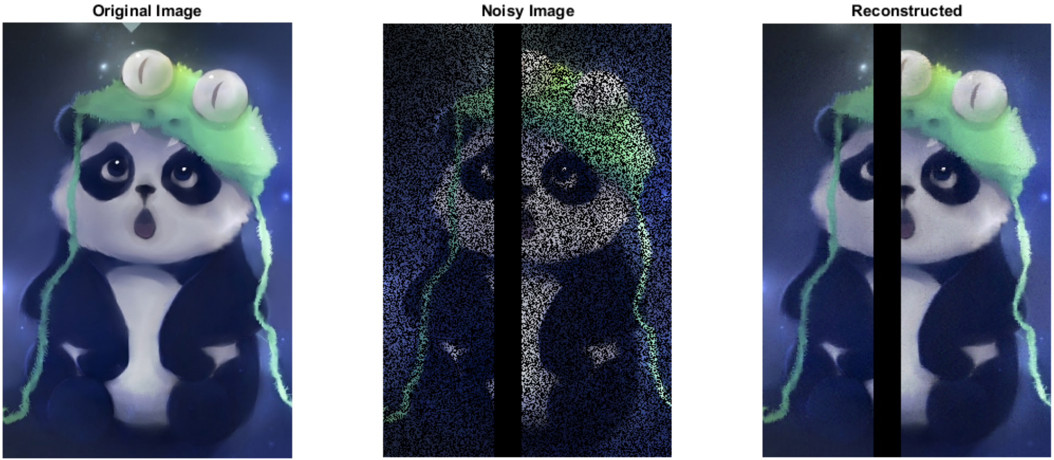


Image Inpainting - Convex Optimization Solver

With 50% of the information. PSNR=35.9 dB.

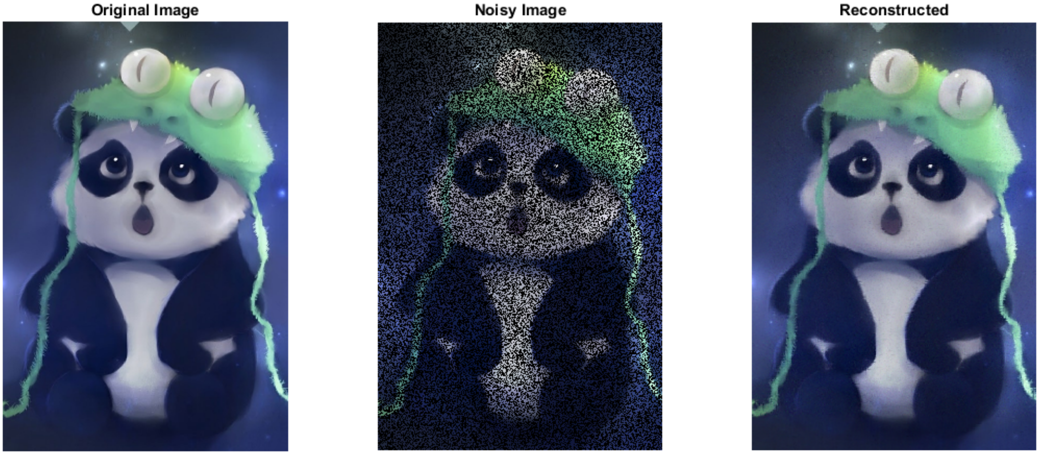
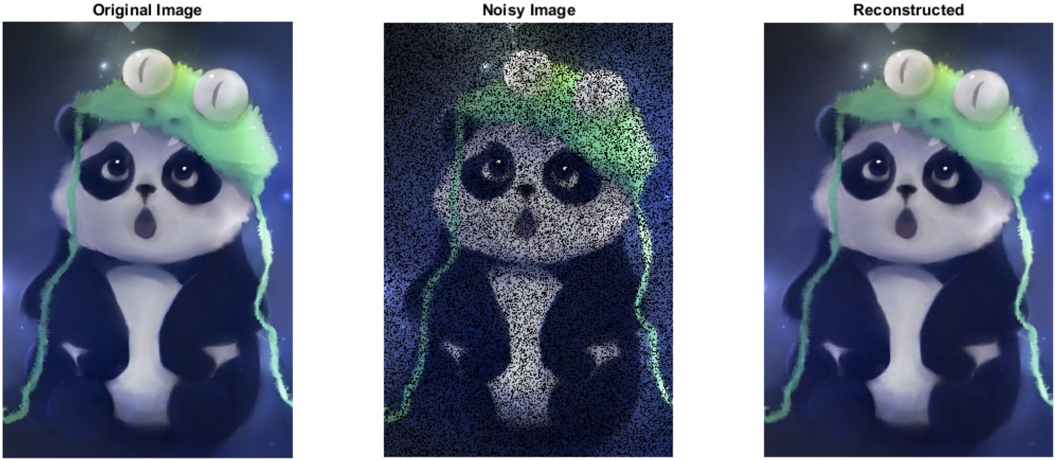


Image Inpainting - SVT Algorithm[+]

With 50% of the Information. PSNR=38.1 dB.



[+]Cai et al. (2010), SIAM Journal on Optimization, Vol. 20, No. 4

Text Removal - Convex Optimization Solver