FSAN/ELEG815: Statistical Learning

Gonzalo R. Arce

Department of Electrical and Computer Engineering
University of Delaware

Matrix Completion

Outline of the Course

1. Review of Probability
2. Stationary processes
3. Eigen Analysis, Singular Value Decomposition (SVD) and Principal Component Analysis (PCA)
4. The Learning Problem
5. Training vs Testing
6. Estimation theory: Maximum likelihood and Bayes estimation
7. The Wiener Filter
8. Adaptive Optimization: Steepest descent and the LMS algorithm
9. Least Squares (LS) and Recursive Least Squares (RLS) algorithm
10. Overfitting
11. Regularization: Ridge and Lasso regression models.
12. Neural Networks
13. Matrix Completion

Outline

Matrix Decompositions, 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:

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

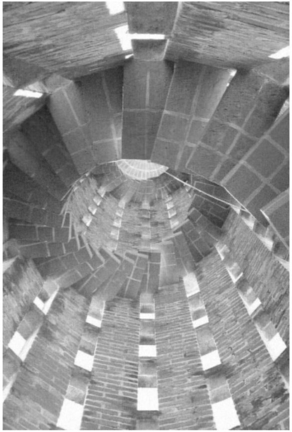
where $||\mathbf{A}||_F^2 = \sum\sum_{i,j} |a_{ij}|^2$ is the Frobenius Norm, and $\Phi(\cdot)$ is a constraint function that encourages $\hat{\mathbf{Z}}$ to be sparse in some sense.

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

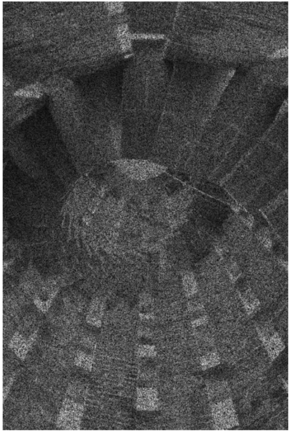
- (a) ℓ_1-norm of all entries of $\hat{\mathbf{Z}}$. Leads to a soft-thresholding $\hat{z}_{ij} = \text{sign}(z_{ij})(|z_{ij}| - \gamma)_+$, where $\gamma > 0$ is such that $\sum_{i=1}^{m} \sum_{j=1}^{n} |\hat{z_{ij}}| = c$.
- (b) Bounds the rank of $\hat{\mathbf{Z}}$, or the number of nonzero singular 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
(d) $\hat{\mathbf{Z}} = \mathbf{UDV}^T$, $\Phi_1(\mathbf{u}_j) \leq c_1$, $\Phi_2(\mathbf{v}_k) \leq c_2$	Penalized SVD
(e) $\hat{\mathbf{Z}} = \mathbf{L} + \mathbf{S}$, $\Phi_1(\mathbf{L}) \leq c_1$, $\Phi_2(\mathbf{S}) \leq c_2$	Additive matrix decomposition

- (d) Imposes penalties on the left and right singular vectors of $\hat{\mathbf{Z}}$. Examples of penalty functions Φ_1 and Φ_2 include the usual ℓ_2 or ℓ_1 norms.
- (e) 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: System Identification in control theory, covariance matrix estimation, 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 $\mathbf{M}$ to make some errors on the observed data:

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

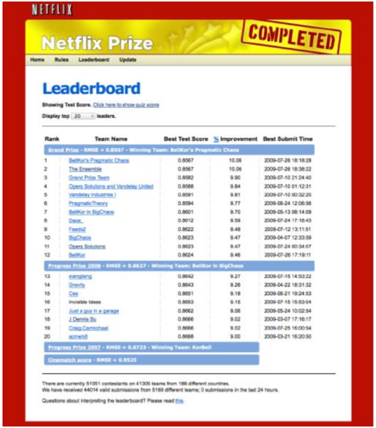
  or equivalently

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

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

Netflix Movie Challenge

- 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

The rating of user i on movie j is given by:

$$z_{ij} = \sum_{\ell=1}^{r} c_{i\ell} g_{j\ell} + w_{ij}, \text{ In Matrix form: } \mathbf{Z} = \mathbf{C}\mathbf{G}^T + W$$

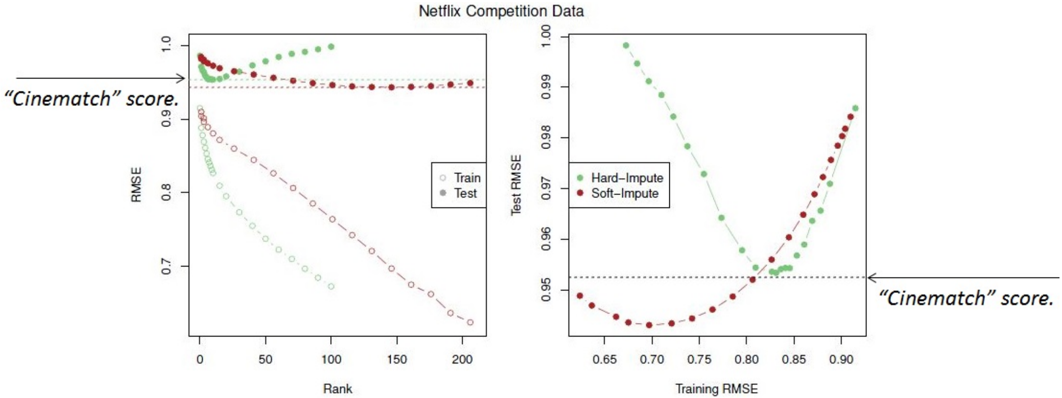
There are r genres of movies, and corresponding to each is a "clique" (small group of people, with shared interests or other features in common) of viewers who like them; a viewer i has a membership weight of c_{il} for the ℓ^{th} clique, and the genre associated with this clique has a score $g_{i\ell}$ for movie j. The overall user rating is obtained by summing these products over ℓ (cliques/genres), and then adding some noise.

	Dirty Dancing	Meet the Parents	Top Gun	The Sixth Sense	Catch Me If You Can	The Royal Tenenbaums	Con Air	Big Fish	The Matrix	A Few Good Men
Customer 1	•	•	•	•	4	•	•	•	•	•
Customer 2	•	•	3	•	•	•	3	•	•	3
Customer 3	•	2	•	4	•	•	•	•	2	•
Customer 4	3	•	•	•	•	•	•	•	•	•
Customer 5	5	5	•	•	4	•	•	•	•	•
Customer 6	•	•	•	•	•	2	4	•	•	•
Customer 7	•	•	5	•	•	•	•	3	•	•
Customer 8	•	•	•	•	•	2	•	•	•	3
Customer 9	3	•	•	•	5	•	•	5	•	•
Customer 10	•	•	•	•	•	•	•	•	•	•

The table shows the data for the 10 customers and 10 movies with the most ratings. (Each rating in the table corresponds to a score z_{ij}

Netflix Movie Challenge

(Left) RMSE over the training and test sets as the rank of the SVD was varied (Hard-impute). Also shown are estimates based on nuclear norm regularization (soft-impute). Training data is doble centered, by removing row and column means ($z_{ij} = \alpha_i + \beta_j + \sum_{\ell=1}^{r} c_{i\ell} g_{j\ell} + w_{ij}$). (Right) Test error only, plotted against training error, for the two methods.



Matrix Completion Using the Nuclear Norm

- Recall $\text{rank}(\mathbf{M}) = \#$ of non-zero singular values of $\mathbf{M}$.
- The nuclear norm $||M||_*$ is the sum of the singular values. It constitutes a relaxation of $\text{rank}(\mathbf{M})$.
- Consider the symmetric matrix $\mathbf{M}$:

$$\begin{pmatrix} x & y \\ y & z \end{pmatrix}$$

  These matrices can be thought of as points in a 3D space, and the coordinate values tell us about the entries in the matrix.
- The singular values for such matrix are:

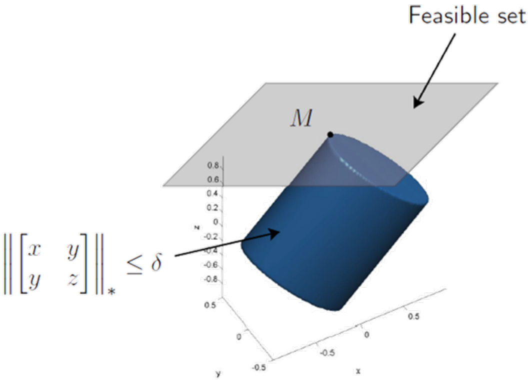
$$s_{1,2} = \frac{1}{\sqrt{2}}\sqrt{x^2 + 2y^2 + z^2 \pm |x+z|\sqrt{(x-y)^2 + 4z^2}} \tag{7}$$

- The unit nuclear norm implies $s_1 + s_2 = 1$. Thus

$$x^2 + 2y^2 + z^2 + 2|y^2 - xz| = 1. \tag{8}$$

- This equation in the 3D plane describes a cylinder.

Matrix Completion Using the Nuclear Norm



The blue cylinder shows the level set of the nuclear norm unit-ball for a symmetric 2×2 matrix. The tangent plane is the feasible set $z = z_0$ for the matrix imputation problem where we observe z and wish to impute x and y. The point M is the solution that we seek, leading to the minimum value for δ

Matrix Completion Using the Nuclear Norm

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

$$||\mathbf{M}||_* = \sum_{k=1}^{n} \sigma_k(\mathbf{M}) \tag{9}$$

- Convex relaxation of the rank minimization problem:

$$\begin{aligned} \text{minimize} \quad & ||\mathbf{M}||_* \\ \text{subject to} \quad & m_{ij} = z_{ij},\ (i,j) \in \Omega\ , \end{aligned} \tag{10}$$

- 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 :

$$\sum_{(i,j)\in\Omega} (z_{ij} - m_{ij})^2 = ||P_\Omega(\mathbf{Z}) - P_\Omega(\mathbf{M})||_F \tag{11}$$

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:

$$\begin{aligned} \text{minimize} \quad & ||\mathbf{M}||_* \\ \text{subject to} \quad & P_\Omega(\mathbf{M}) = P_\Omega(\mathbf{Z}) \ , \end{aligned} \tag{12}$$

- The SVD of a matrix $\mathbf{M}$ of rank r is:

$$\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V} \ , \ \boldsymbol{\Sigma} = \mathrm{diag}(\{\sigma_i\}_{1\leq i\leq r}) \tag{13}$$

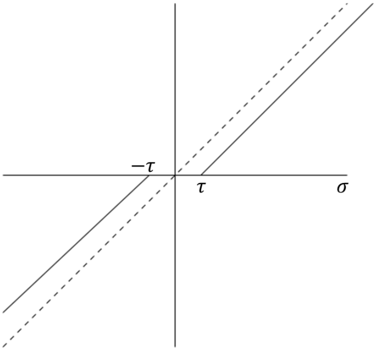
[+] Cai et al. (2010), SIAM Journal on Optimization, Vol. 20, No. 4

Singular Value Thresholding (SVT)

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

$$D_\tau(\mathbf{M}) = \mathbf{U} D_\tau(\mathbf{\Sigma}) \mathbf{V}^* \ , \ D_\tau(\mathbf{\Sigma}) = \mathrm{diag}(\{\sigma_i - \tau\}_+) \tag{14}$$

where t_+ is the positive part of t, $t_+ = \max(0, t)$. Operator applies soft-thresholding to the singular values of $\mathbf{M}$, 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 $k = 1, 2, \ldots,$

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

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

Spectral Regularization

- Problem: Unrealistic to model observed entries as being noiseless.
- Relaxed version of (5)

$$\underset{\mathbf{M}}{\text{minimize}} \quad \frac{1}{2} \sum_{(i,j)\in\Omega} (z_{ij} - m_{ij})^2 + \lambda ||\mathbf{M}||_* \,, \tag{15}$$

- Introduce bias to decrease variance.
- Avoids over-fitting.

Soft SVD

- Consider the SVD $\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ of a rank r matrix $\mathbf{M}$.
- The convex optimization problem

$$\underset{\mathbf{M}}{\text{minimize}} \quad \frac{1}{2}||P_\Omega(\mathbf{Z}) - P_\Omega(\mathbf{M})||_F^2 + \lambda||\mathbf{M}||_* \tag{16}$$

- Solution is the **Soft-thresholded SVD**

$$D_\lambda(\mathbf{M}) = \mathbf{U}\mathbf{\Sigma}_\lambda\mathbf{V}^T \tag{17}$$

where, $\boldsymbol{\Sigma}_\lambda = \mathrm{diag}\{(\sigma_1 - \lambda)_+, \ldots, (\sigma_r - \lambda)_+\}$

Convex Optimization Problem

$$\underset{\mathbf{M}}{\text{minimize}} \quad \frac{1}{2}||P_\Omega(\mathbf{Z}) - P_\Omega(\mathbf{M})||_F^2 + \lambda||\mathbf{M}||_* \tag{18}$$

- This is a semi-definite program (SDP), convex in **M**.
- Complexity of existing off-the-shelf solvers:
  - interior-point methods: $O(n^4)\cdots O(n^5)\cdots$
  - (black box) first-order methods complexity: $O(n^3)$
- Use an iterative soft SVD (next slide), with cost per soft SVD $O\left[(m+n)\hat{\text{A}}\Delta r + |\Omega|\right]$ where r is rank of solution.

Soft-Impute for Matrix Completion,[+]

1. Initialize $\mathbf{Z}^{old}$ and create a decreasing grid $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_K$
2. For $k = 1, \ldots, K$, set $\lambda = \lambda_k$ and iterate until convergence:

Compute $\hat{\mathbf{Z}}_\lambda \leftarrow D_\lambda(P_\Omega(\mathbf{Z}) + P_\Omega^\perp(\mathbf{Z}^{old}))$

Update $\mathbf{Z}^{old} \leftarrow \hat{\mathbf{Z}}_\lambda$

3. Output the sequence solutions $\hat{\mathbf{Z}}_{\lambda_1}, \cdots, \hat{\mathbf{Z}}_{\lambda_k}$

$P_\Omega^\perp$ projects onto the complement of the set Ω. [+]Mazumder et al. Journal of Machine Learning Research 2010

Soft-Impute for Matrix Completion,[+]

- Each iteration requires an SVD of a large dense matrix, even though $P_\Omega(\mathbf{Z})$ is sparse.

$$P_\Omega(\mathbf{Z}) + P_\Omega^\perp(\mathbf{Z}^{old}) \tag{19}$$

- Strategy:

$$P_\Omega(\mathbf{Z}) + P_\Omega^\perp(\mathbf{Z}^{old}) = \underbrace{\left\{P_\Omega(\mathbf{Z}) - P_\Omega(\mathbf{Z}^{old})\right\}}_{\text{Sparse}} + \underbrace{\mathbf{Z}^{old}}_{\text{Low Rank}} \tag{20}$$

- The first component is sparse, with $|\Omega|$ non-missing entries. The second component is a soft-thresholded SVD, so can be represented using the corresponding components.
- Each component's special structure can be exploited to efficiently perform left and right multiplications by a vector, and thereby apply iterative Lanczos methods to compute a (low rank) SVD efficiently.

[+]Mazumder et al. Journal of Machine Learning Research 2010

Impediments and Solutions

- How many samples N do we need in order to be able to recover the matrix of dimensions $p \times p$ when $N \ll p^2$?
- It is impossible to recover the matrix exactly if there are no observed entries in some row or column, even if it is rank one.
- Example: Consider the rank one matrix $\mathbf{Z} = \mathbf{e}_1 \mathbf{e}_1^T$ with a single one in its upper left corner:

$$\mathbf{Z} = \begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \quad \mathbf{Z}' = \begin{pmatrix} \boldsymbol{v_1} & \boldsymbol{v_2} & \boldsymbol{v_3} & \boldsymbol{v_4} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix}$$

- If we only observe $N \ll p^2$ entries of this matrix, with the entries chosen uniformly at random, then with high probability, we will not observe the single nonzero entry.
- $\mathbf{Z}' = \mathbf{e}_1 \mathbf{v}^T$, where $\mathbf{v} \in \mathbb{R}^p$ is an arbitrary p vector.

Theoretical Results for Matrix Completion

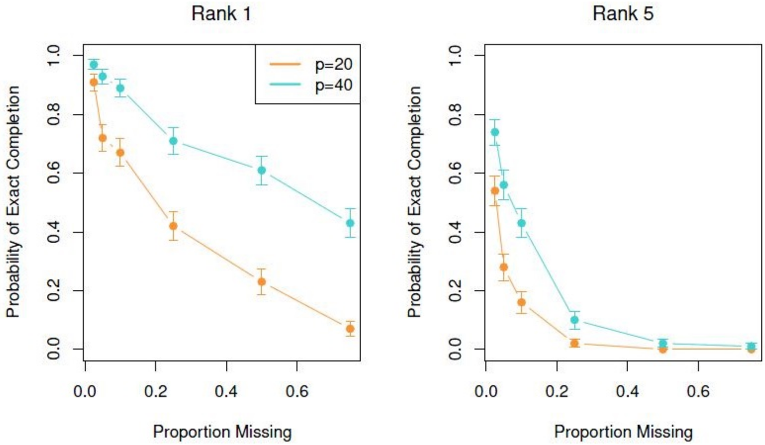
- To exclude troublesome matrices $\rightarrow$ draw matrices from random ensemble e.g. construct a random matrix of the form $\mathbf{Z} = \sum_{j=1}^{r} \mathbf{a}_j \mathbf{b}_j^T$ where the random vectors $a_j \sim N(0, I_p)$ and $b_j \sim N(0, I_p)$ are all independently drawn.
- Gross (2011), shows that the nuclear norm relaxation succeeds in exact recovery if:

$$N \geq Crp \log p, \tag{21}$$

where $C > 0$ is a fixed universal constant.

Theoretical Results for Matrix Completion

- Set to missing a fixed proportion of entries and applied *Soft-Impute* with λ chosen small enough so that $||P_{\Omega}^{\perp}(\mathbf{Z}-\hat{\mathbf{Z}})||_F^2/||P_{\Omega}^{\perp}(\mathbf{Z})||_F^2 < 10^{-5}$.
- Process repeated 100 times for various values of rank r and the proportion set to missing.



Convex matrix completion in the no-noise setting. Shown are probabilities of exact completion (mean $\pm$one standard error) as a function of the proportion missing, for $n \times n$ matrices with n $2\,\{20, 40\}$. The true rank of the complete matrix is one in the left panel and five in the right panel.

Image Inpainting - Convex Optimization Solver

With 70% of the Information.

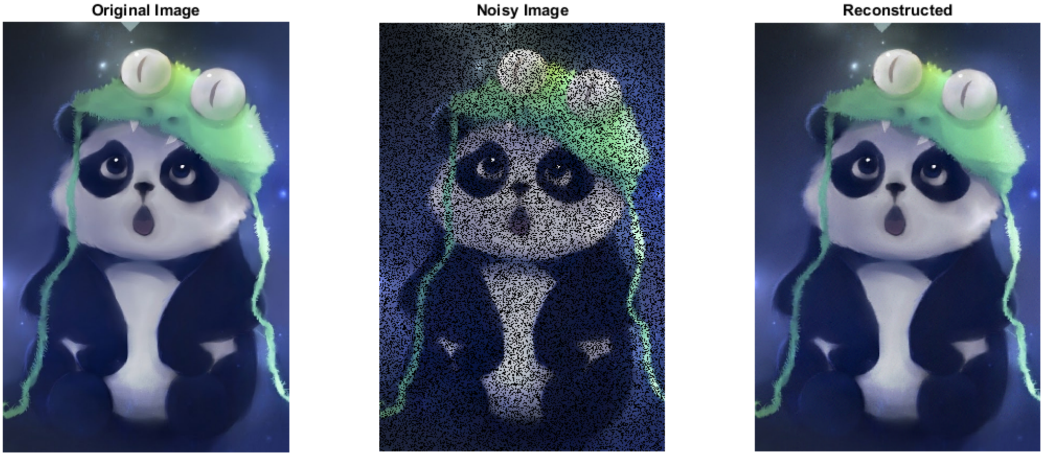
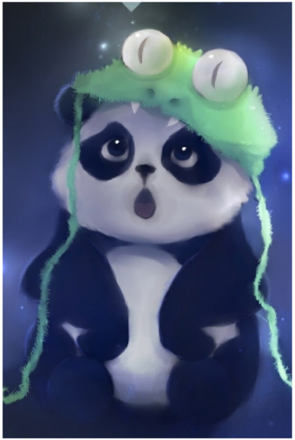


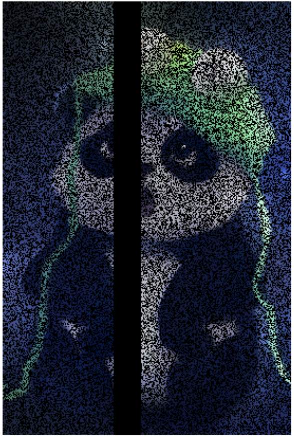
Image Inpainting - Convex Optimization Solver

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

Original Image



Noisy Image



Reconstructed

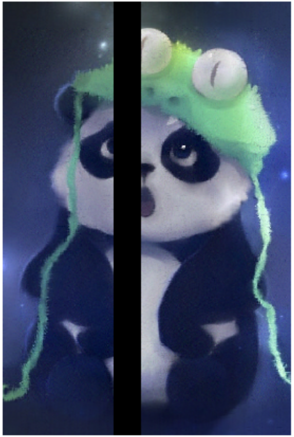
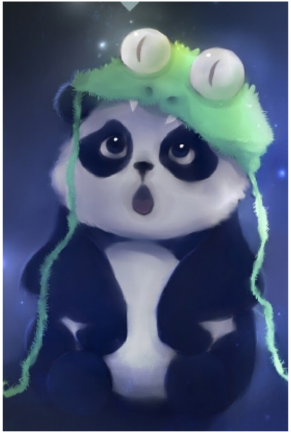


Image Inpainting - Convex Optimization Solver

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

Original Image



Noisy Image

Reconstructed

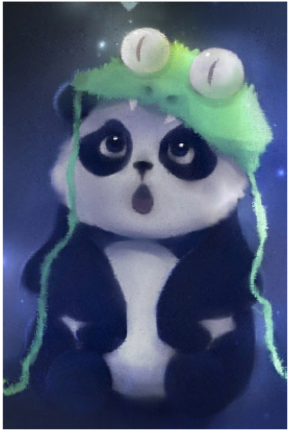
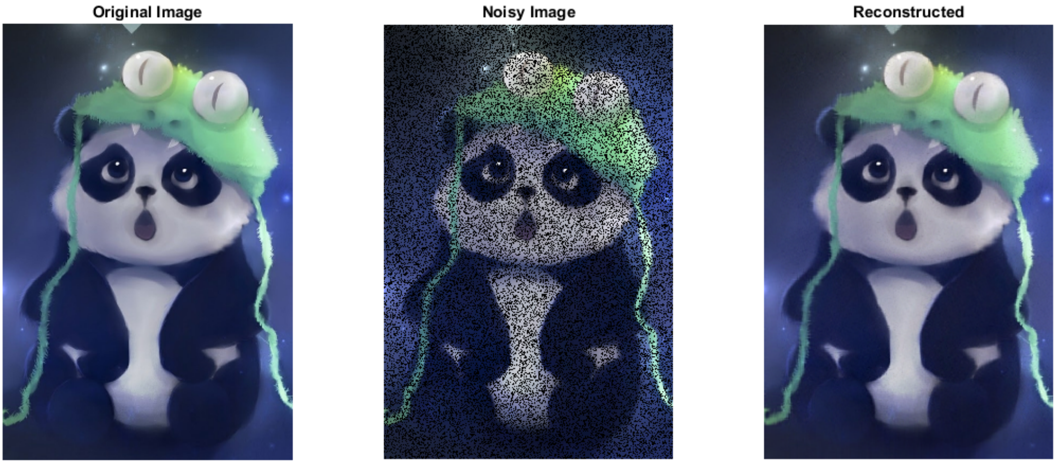


Image Inpainting - SVT Algorithm[+]

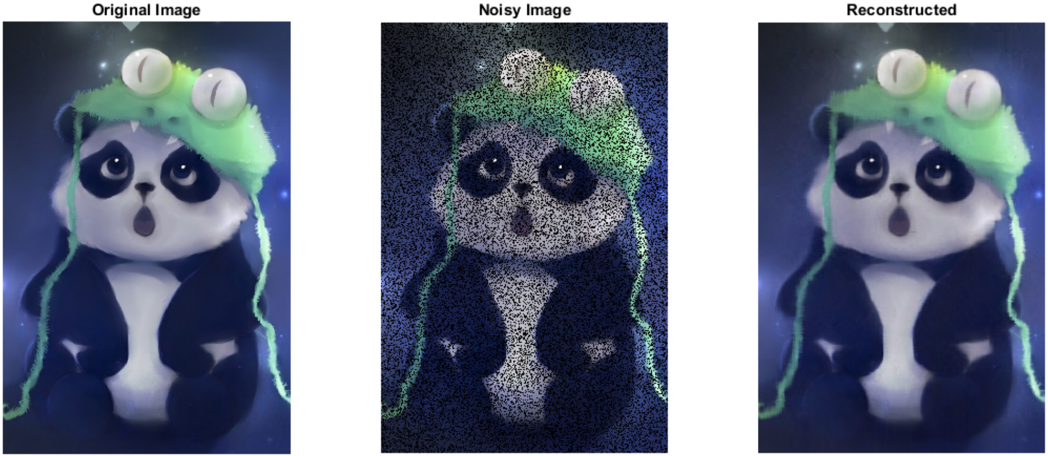
With 50% of the Information. PSNR=38.1 dB.



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

Image Inpainting - Soft Impute Algorithm+

With 50% of the Information. PSNR= 35.7 dB.



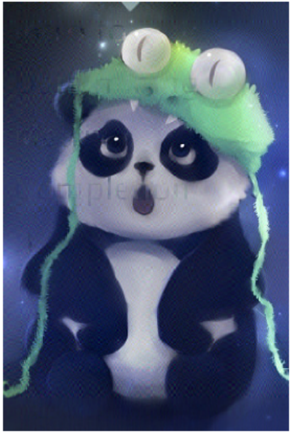
+Mazunder et al. Journal of Machine Learning Research 2010

Text Removal - Convex Optimization Solver

Original Image

Noisy Image

Reconstructed



Additive Matrix Decomposition

- Objective: Decompose a matrix into the sum of two or more matrices: $\mathbf{Z} = \mathbf{L}^* + \mathbf{S}^*$.
- Components should have complementary structures: eg. sum of a low-rank matrix with a sparse matrix.
- Applications: Factor analysis, Robust forms of PCA and matrix completion, and multivariate regression.
- These applications can be described in a noisy linear observation model $\mathbf{Z} = \mathbf{L}^* + \mathbf{S}^* + \mathbf{W}$, where the pair $(\mathbf{L}^*, \mathbf{S}^*)$ specifies the additive matrix decomposition into low rank and sparse components, and $\mathbf{W}$ is a noise matrix.

Additive Matrix Decomposition

- Estimate the pair $(\mathbf{L}^*, \mathbf{S}^*)$ as:

$$\min_{\mathbf{L},\mathbf{S}\in\mathbb{R}^{m\times n}} \left\{\frac{1}{2}||\mathbf{Z}-(\mathbf{L}+\mathbf{S})||_F^2+\lambda_1\Phi_1(\mathbf{L})+\lambda_2\Phi_2(\mathbf{S})\right\} \tag{22}$$

  where Φ_1 and Φ_2 are penalty functions each designed to enforce type of generalized sparsity.
- In the case of low rank and sparse matrices, the penalty functions are: $\Phi_1(\mathbf{L}) = ||\mathbf{L}||_*$ and $\Phi_2(\mathbf{S}) = ||\mathbf{S}||_1$

Factor Analysis With Sparse Noise

- Widely used form of linear dimensionality reduction that generalizes PCA.
- Generative model: Generate random vectors $y_i \in \mathbb{R}^p$ using the noisy subspace model:

$$y_i = \mu + \boldsymbol{\Gamma} u_i + w_i, \text{for } i = 1, 2, \cdots, N. \tag{23}$$

- $\mu \in \mathbb{R}^p$ is a mean vector, $\boldsymbol{\Gamma} \in \mathbb{R}^{p \times r}$ is a loading matrix, and the random vectors $u_i \sim N(0, \mathbf{I}_{r \times r})$ and $w_i \sim N(0, \mathbf{S}^*)$ are independent.
- Given N samples, the goal is to estimate the column of the loading matrix $\boldsymbol{\Gamma}$, or equivalently, the rank r matrix $\mathbf{L}^* = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^T \in \mathbb{R}^{p \times p}$ that spans the column space of $\boldsymbol{\Gamma}$.

Factor Analysis With Sparse Noise

- The covariance matrix of y_i has the form $\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^T + \mathbf{S}^*$
- When $\mathbf{S}^*$ is sparse, the problem of estimating $\mathbf{L}^* = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^T$ can be understood as an instance of our general problem $p = N$.
- Let our observation matrix $\mathbf{Z} \in \mathbb{R}^{p\times p}$ be the sample covariance matrix $\frac{1}{N}\sum_{i=1}^{N} y_i y_i^T$.
- Thus, $\mathbf{Z} = \mathbf{L}^* + \mathbf{S}^* + \mathbf{W}$, where $\mathbf{L}^* = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^T$ is of rank r and $\mathbf{W} := \frac{1}{N}\sum_{i=1}^{N} y_i y_i^T - \{\mathbf{L}^* + \mathbf{S}^*\}$

Robust PCA

Standard PCA:

- Find SVD of $\mathbf{Z} \in \mathbb{R}^{N \times p}$, where row i represents the i^{th} sample of a p-dimensional data vector.
- Rank-r SVD is obtained by minimizing the squared Frobenius norm $||\mathbf{Z} - \mathbf{L}||_F^2$ subject to a rank constraint on $\mathbf{L}$.
- If some entries of $\mathbf{Z}$ are corrupted, its solution is very sensitive to noise.

Robust PCA

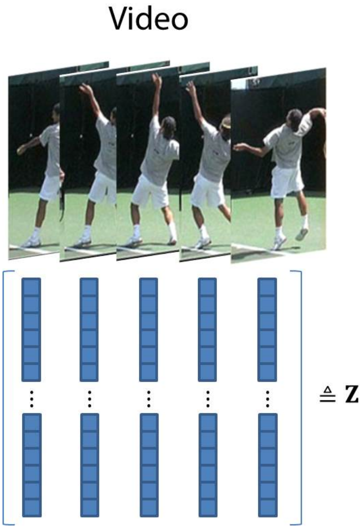
- Additive decompositions provide one way in which to introduce robustness to PCA.
- Instead of approximating $\mathbf{Z}$ with a low-rank matrix, approximate it with the sum $\mathbf{L}+\mathbf{S}$ of a low-rank matrix with a sparse component.
- In the case of element-wise corruption, the component $\mathbf{S}$ would be modeled as a row-sparse matrix. Given some target rank r and sparsity k, the direct approach solves the optimization problem.

$$\min_{\text{rank}(\mathbf{L})\leq r, \text{card}(\mathbf{S})\leq k} \frac{1}{2}||\mathbf{Z}-(\mathbf{L}+\mathbf{S})||_F^2 \tag{24}$$

- Criterion is non-convex, due to both the rank and cardinality constraints. A natural convex relaxation is provided $\Phi_1(\mathbf{L}) = ||\mathbf{L}||_*$ and $\Phi_2(\mathbf{S}) = \sum_{i,j}|s_{i,j}|$ for element wise sparsity.

Robust PCA: Video Surveillance

Columns of $\mathbf{Z}$ are frames from a video.



Robust PCA: Video Surveillance