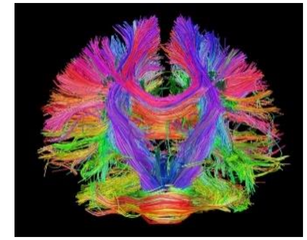



CS-570
Statistical Signal Processing

Lecture 9: Matrix Completion

Spring Semester 2019

Grigorios Tsagkatakis

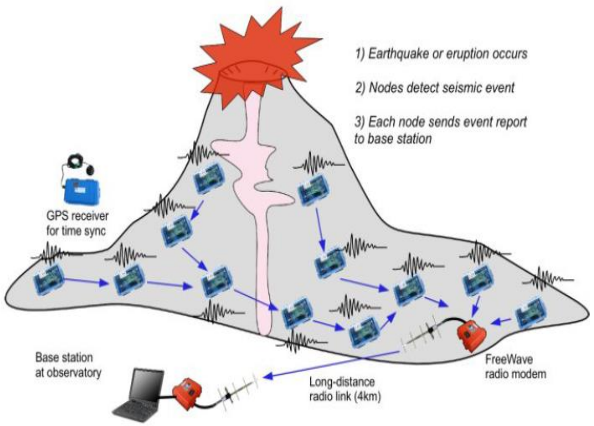
Incomplete Matrices

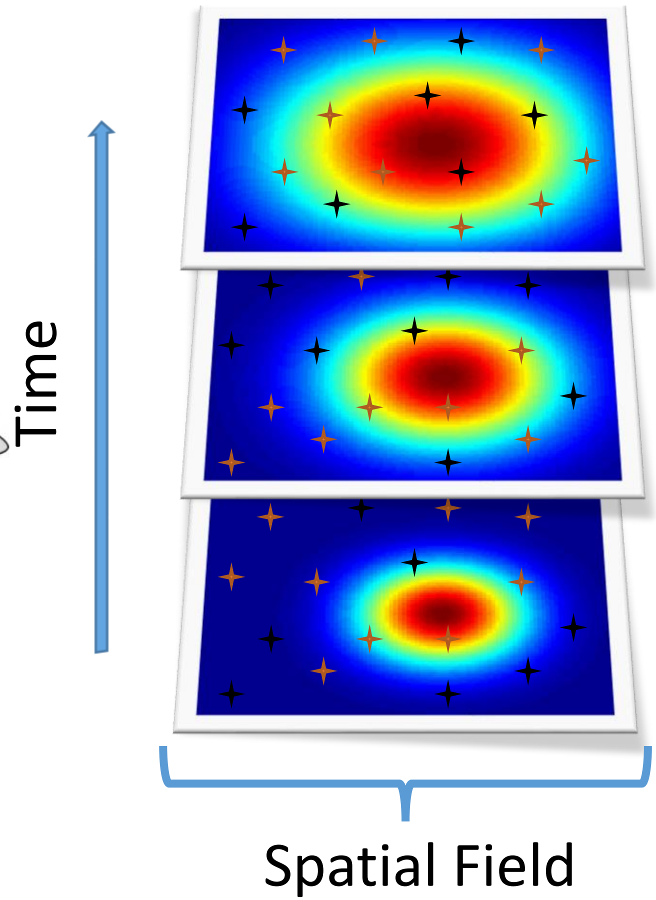
The 2009 Netflix Prize

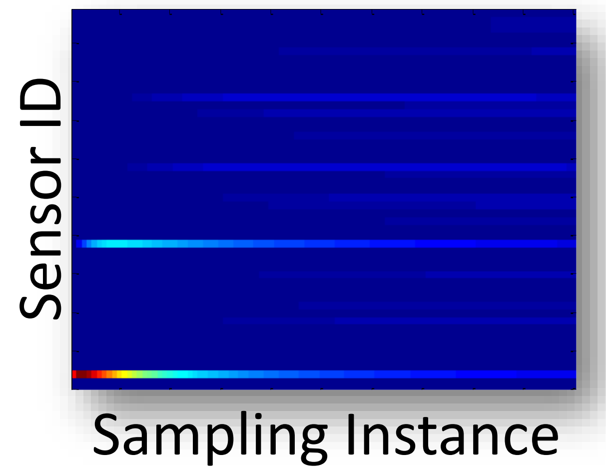
- Given user-movie rating, Guess missing entries
- 100M ratings, $1,000,000 prize
- Winner: BellKor's Pragmatic Chaos team (10% improvement)

	John	Anne	Scot	Mark	Alice
Chicago	2	5	?	?	?
Matrix	5	?	5	?	?
Star wars	?	?	5	?	1
Inception	?	3	?	2	?
Alien	4	1	?	?	?
Pulp Fiction	?	?	4	?	2

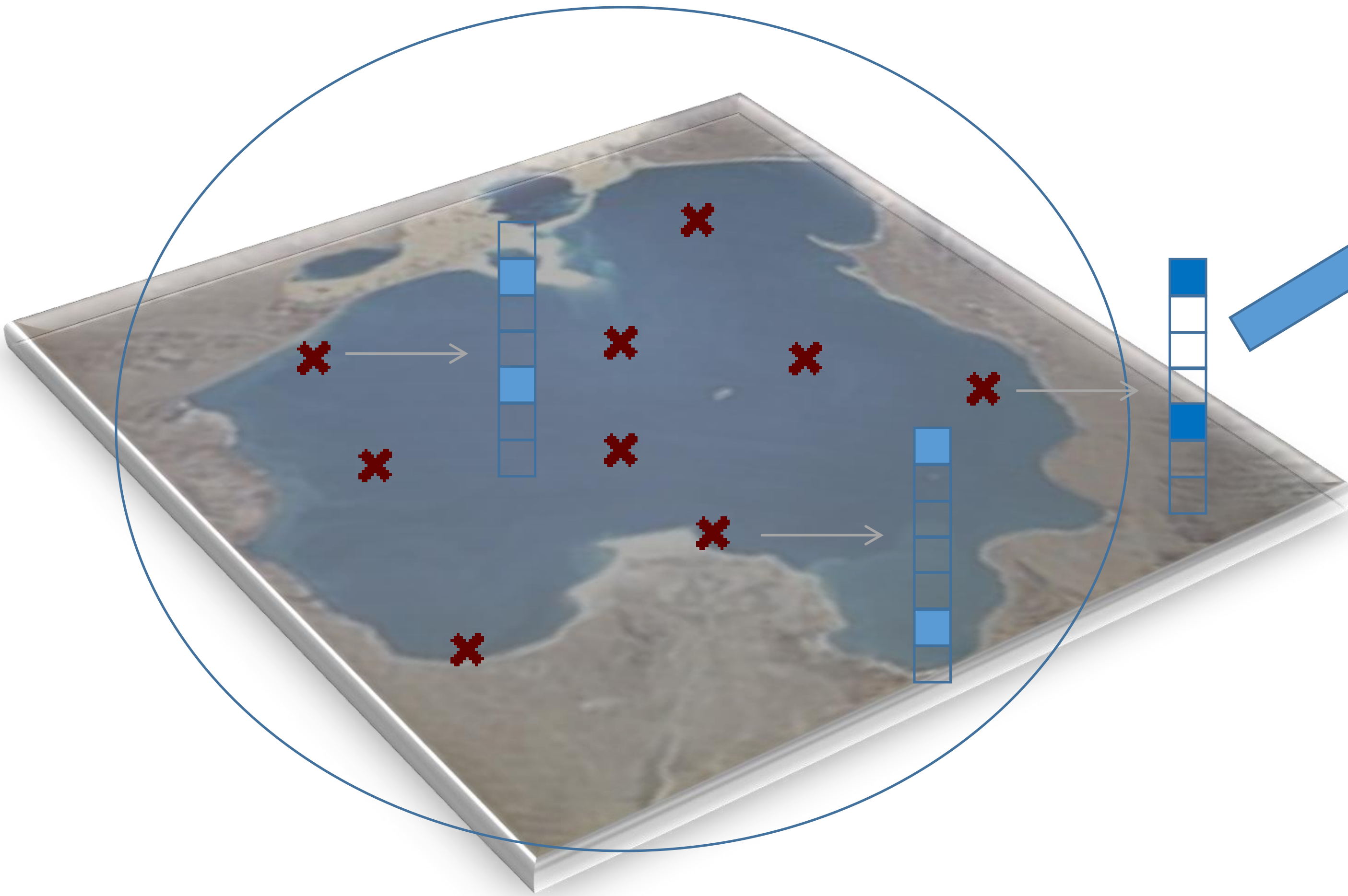
Multivariate observations

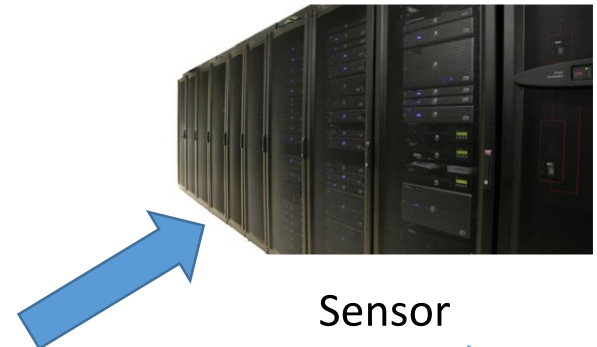


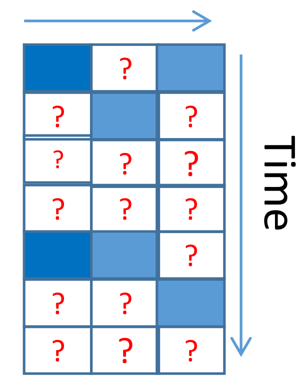




Sampling a WSN







Measurement Matrix

Matrix Rank

The **rank** of a matrix *M* is the size of the largest collection of linearly independent *columns of M* (the **column rank**) or the size of the largest collection of linearly independent *rows of M* (the **row rank**)

- Row Echelon Form

$$\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix} R_2 \rightarrow 2r_1 + r_2 \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 3 & 5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 3 & 5 & 0 \end{bmatrix} R_3 \rightarrow -3r_1 + r_3 \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & -1 & -3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & -1 & -3 \end{bmatrix} R_3 \rightarrow r_2 + r_3 \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 0 \end{bmatrix} R_1 \rightarrow -2r_2 + r_1 \begin{bmatrix} 1 & 0 & -5 \\ 0 & 1 & 3 \\ 0 & 0 & 0 \end{bmatrix}$$

Rank=2

A matrix is in **row echelon form** if

(i) all nonzero rows are above any rows of all zeroes

(ii) The leading coefficient of a nonzero row is always strictly to the right of the leading coefficient of the row above it

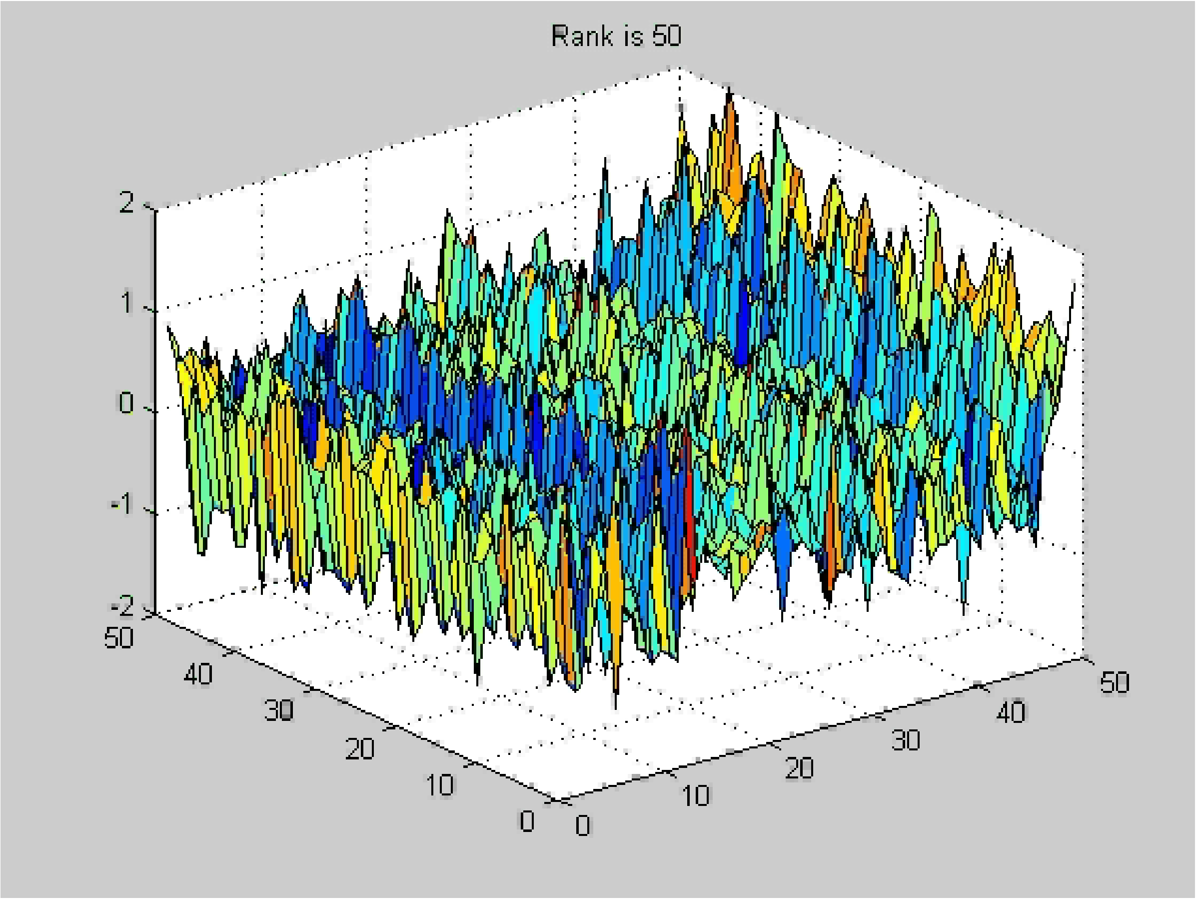
Matrix Rank

- The rank of an $m \times n$ matrix is a nonnegative integer and cannot be greater than either m or n. That is, $\text{rank}(M) \leq \min(m, n)$.
- A matrix that has a rank as large as possible is said to have **full rank**; otherwise, the matrix is **rank deficient**.

$$\text{rank}(AB) \leq \min(\text{rank } A, \text{rank } B).$$

$$\text{rank}(A^T A) = \text{rank}(A A^T) = \text{rank}(A) = \text{rank}(A^T)$$

Matrix Rank



Singular Value Decomposition (SVD)

Given any $m \times n$ matrix **M**, algorithm to find matrices **U**, **Σ**, and **V** such that $\mathbf{M} = \mathbf{U}\,\boldsymbol{\Sigma}\,\mathbf{V}^{T}$

- **U**: left singular vectors (orthonormal)
- **Σ**: diagonal containing singular values
- **V**: right singular vectors (orthonormal)

$$M = U \Sigma V^{T}$$

$m \times m$ | $m \times n$ | V is $n \times n$

$$\begin{pmatrix} \\ M \\ \\ \end{pmatrix} = \begin{pmatrix} \\ \mathbf{U} \\ \\ \end{pmatrix} \begin{pmatrix} s_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & s_n \end{pmatrix} \begin{pmatrix} \\ \mathbf{V} \\ \\ \end{pmatrix}^{\mathrm{T}}$$

Singular Value Decomposition (SVD)

Properties

- The s_i are called the singular values of **M**
- If **M** is singular, some of the s_i will be 0
- In general *rank*(**M**) = number of nonzero s_i
- SVD is mostly unique (up to permutation of SV)

Low rank approximation

Matrix norms

- Frobenius norm can be computed from SVD $\|M\|_{\mathrm{F}} = \sum_i \sum_j m_{ij}^2$

- Changes to a matrix $\leftrightarrow$ changes to singular values $\|M\|_{\mathrm{F}} = \sum_i s_i^2$

Low rank approximation

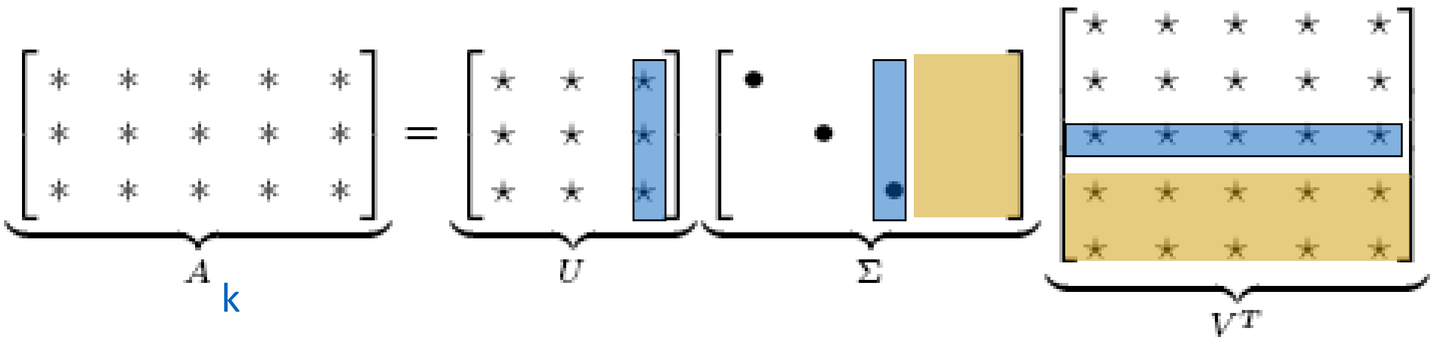
Approximation problem: Find $\boldsymbol{M_k}$ of rank $\boldsymbol{k}$ such that

$$M_k = \min_{X:rank(X)=k} \|M - X\|_F$$

Singular Value Decomposition (SVD)

- Solution via SVD $M_k = U \, \text{diag}(\sigma_1, \ldots, \sigma_k, \underbrace{0, \ldots, 0}) V^T$

set smallest r-k singular values to zero



$$M_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$$

column notation: sum of rank 1 matrices

Approximation error

- How good (bad) is this approximation?
- It's the best possible, measured by the Frobenius norm of the error:

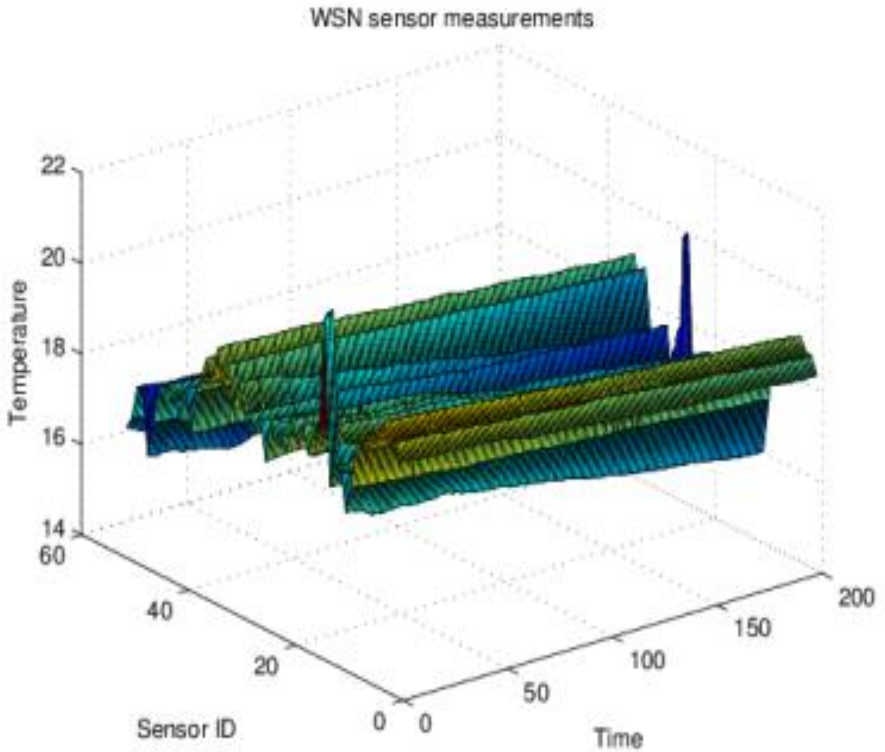
$$\min_{X:rank(X)=k} \|M - X\|_F = \|M - M_k\|_F = \sigma_{k+1}$$

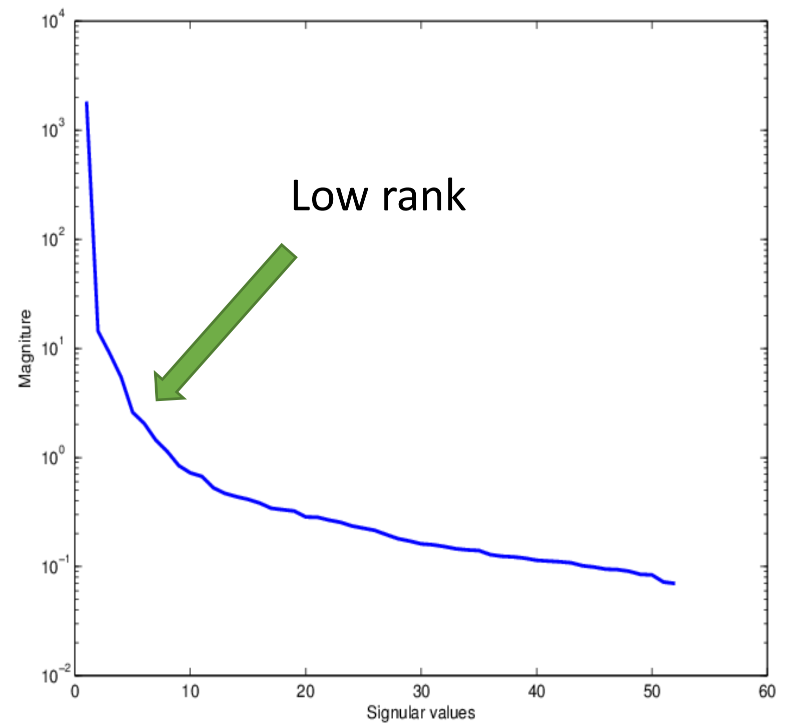
where the σ_i are ordered such that $\sigma_i \geq \sigma_{i+1}$.

Suggests why Frobenius error drops as *k* increased.

Data model

- ✦Data modeling
  - ✦ Spatio-temporal correlations <-> Low rank measurement matrix





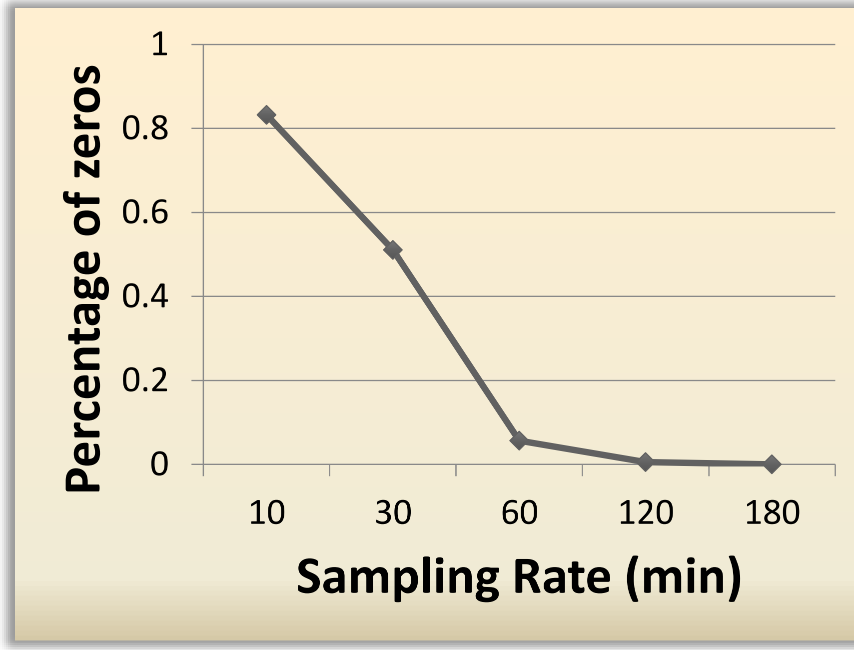
The case of missing values

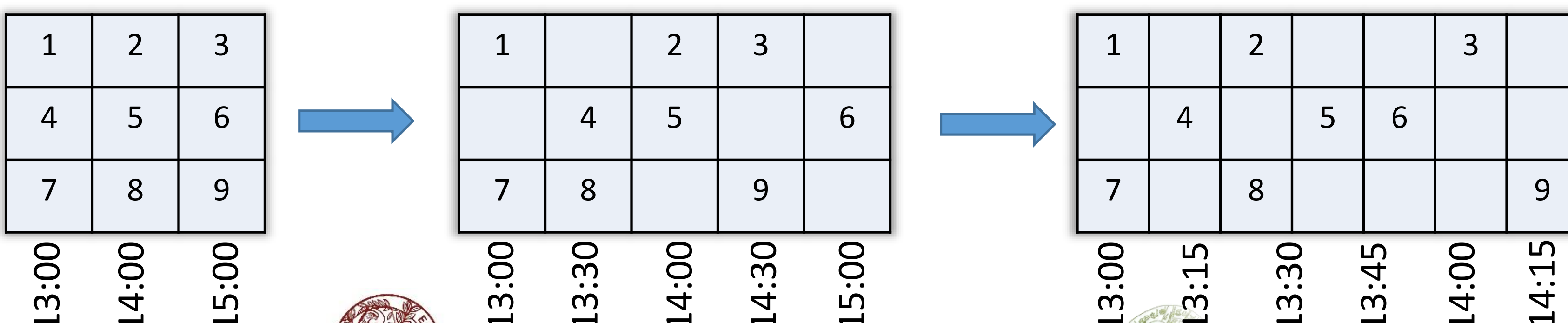
Power consumption

Packet losses

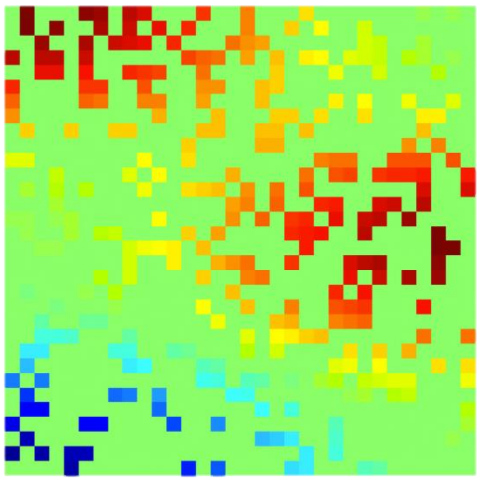
Temporal sampling

- Sampling rate
- De-synchronization
- Temporal resolution

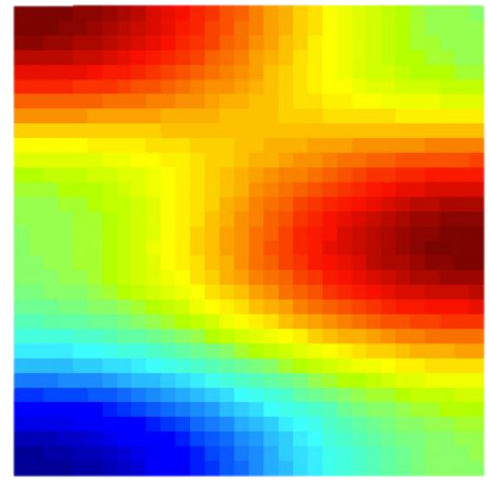




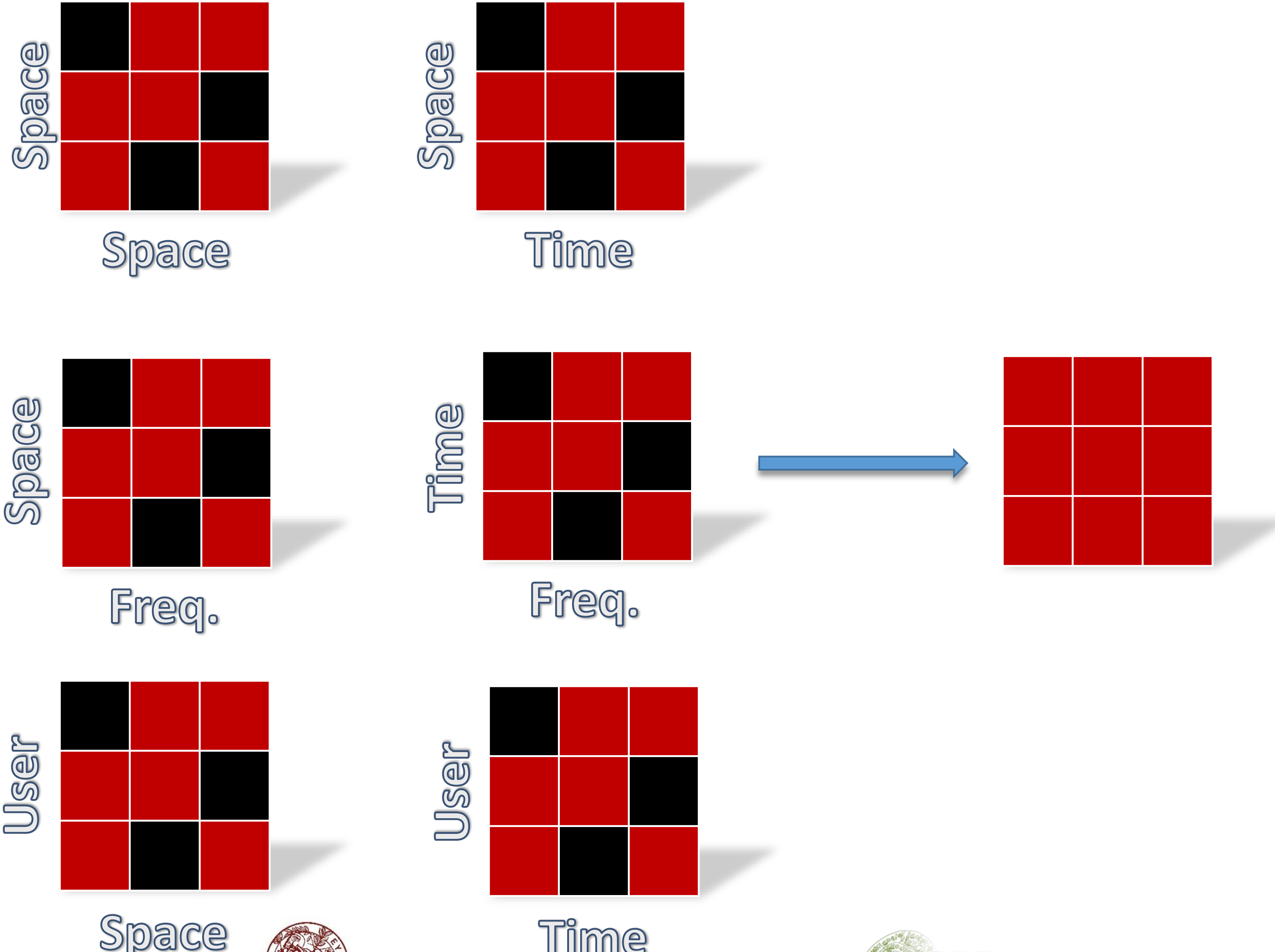
Matrix completion



low rank matrix with missing entries



low rank matrix


Space
Space
Space
Time
Space
Freq.
Time
Freq.
User
Space
User
Time

Matrix Completion (MC)

Let $\mathbf{M} = [\mathbf{M_0}, ..., \mathbf{M_1}] \in \mathbf{R^{n \times s}}$ be a measurement matrix consisting of *s* measurements from *n* different sources.

Recovery of **M** is possible from *k<<ns* random entries if matrix **M** is *low rank* and $k \geq Cn^{6/5} r log(n)$

To recover the unknown matrix, solve:

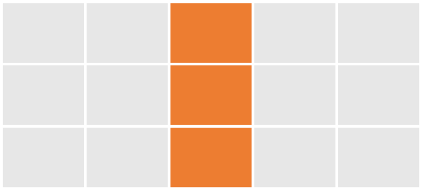
$$\min\{\ rank(\mathbf{X})\ \ :\ \ \mathcal{A}(\mathbf{X}) = \mathcal{A}(\mathbf{M})\}$$

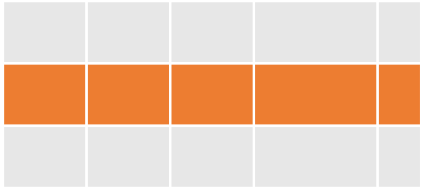
Rank constraint makes problem NP-hard....

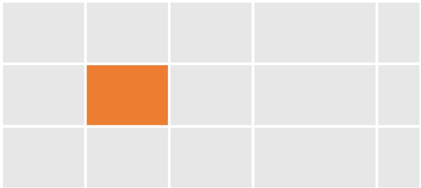
Sampling operator

Sampling operator $\mathcal{A}_{ij}(\mathbf{M}) = \begin{cases} M_{ij}, & \text{if } ij \in S \\ 0, & \text{otherwise} \end{cases}$

- Not all low-rank matrices can be recovered from partial measurements!
  - … a matrix containing zeroes everywhere except the top-right corner.
  - This matrix is low rank, but it ***cannot*** be recovered from knowledge of only a fraction of its entries!

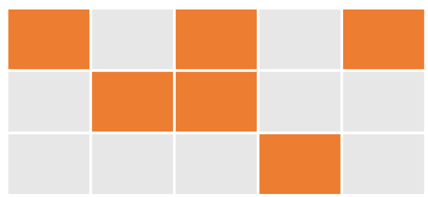






Matrix Coherence

The coherence of subspace ***U*** of $\Re^n$ and having dimension *r* with respect to the canonical basis $\{\mathbf{e}_i\}$ is defined as: $\mu(U) = \frac{n}{r} \max_{1 \le i \le n} \|Ue_i\|^2$



$\mu(U) = O(1)$

– sampled from the uniform distribution with $r > \log n$

Formal definition of key assumptions

- Consider an underlying matrix **M** of size n_1 by n_2. Let the SVD of **M** be given as follows:

$$M = \sum_{k=1}^{r} \sigma_k u_k v_k^T$$

- We make the following assumptions about **M**: $\sum_{k=1}^{r} u_k v_k^T$

(A0) $\mu_1 \sqrt{r/(n_1 n_2)}, \mu_1 > 0$

(A1) The maximum entry in the n_1 by n_2 matrix is upper bounded by

$$\exists \mu_0 \text{ such that } \max(\mu(U), \mu(V)) \leq \mu_0$$

FORTH Institute of Computer Science

What do these assumptions mean

(A0) means that the singular vectors of the matrix are sufficiently **incoherent** with the canonical basis.

(A1) means that the singular vectors of the matrix are **not spiky**

- canonical basis vectors are spiky signals – the spike has magnitude 1 and the rest of the signal is 0;
- a vector of n elements with all values equal to 1/square-root(n) is not spiky.

What is the trace-norm of a matrix?

- The nuclear / trace norm of a matrix is the **sum of its singular values.**

$$\|\mathbf{M}\|_* = \sum_{i=1}^{k} \sigma_i$$

- It is a **softened version of the rank** of a matrix
- Similar to the $L_0 \rightarrow L_1$-norm of a vector
- Minimization of the trace-norm is a **convex optimization problem** and can be solved efficiently.
- This is similar to the L_1-norm optimization (in compressive sensing) being efficiently solvable.

Matrix Completion (MC)

Relaxation

$$\min\{\ \|\mathbf{M}\|_* \ :\ \ \mathcal{A}(\mathbf{X}) = \mathcal{A}(\mathbf{M})\}$$

Performance

$$\left\|M - M^*\right\|_F^2 \le 4\sqrt{\frac{(2+p)\min(n_1, n_2)}{p}}\delta + 2\delta,$$

$$\text{where } p = \text{fraction of known entries} = \frac{m}{n_1 n_2} = \frac{|\Omega|}{n_1 n_2}$$

Noisy case

$$\min\{\ \|\mathbf{M}\|_* \ :\ \ \|\mathcal{A}(\mathbf{X}) - \mathcal{A}(\mathbf{M})\|_F^2 \le \epsilon\}$$

Recovery guarantees

Theorem 1.3 *Let* $\boldsymbol{M}$ *be an* $n_1 \times n_2$ *matrix of rank* r *obeying* **A0** *and* **A1** *and put* $n = \max(n_1, n_2)$. *Suppose we observe* m *entries of* $\boldsymbol{M}$ *with locations sampled uniformly at random. Then there exist constants* C, c *such that if*

$$m \geq C \max(\mu_1^2, \mu_0^{1/2}\mu_1, \mu_0 n^{1/4})\, nr(\beta \log n) \tag{1.9}$$

for some $\beta > 2$, *then the minimizer to the problem* (1.5) *is unique and equal to* $\boldsymbol{M}$ *with probability at least* $1 - cn^{-\beta}$. *For* $r \leq \mu_0^{-1} n^{1/5}$ *this estimate can be improved to*

$$m \geq C\, \mu_0\, n^{6/5} r(\beta \log n) \tag{1.10}$$

with the same probability of success.

the trace-norm minimizer

Candes EJ, Recht B. Exact matrix completion via convex optimization. Found. of Computational mathematics. 2009, 9(6):717-772.
Candes EJ, Tao T. The power of convex relaxation: Near-optimal matrix completion. Information Theory, IEEE Transactions on. 2010, 56(5):2053-2080.

Matrix Completion solvers

- Objective $\text{minimize}_{\mathbf{X}} \ \|\mathcal{A}(\mathbf{X}) - \mathbf{y}\|_2 + \lambda\|\mathbf{X}\|_*$

- Iterative Hard Thresholding

$$\boldsymbol{Y}_{k+1} = \boldsymbol{X}_k - \gamma_k \mathcal{A}^*(\mathcal{A}(\boldsymbol{X}_k) - \boldsymbol{y}))$$

$$\boldsymbol{X}_{k+1} = \text{ProjectRank}_R(\boldsymbol{Y}_{k+1}).$$

← **SVD**

Matrix Completion Solvers

- Reformulate $\text{minimize}_{\mathbf{X}} \; \|\mathcal{A}(\mathbf{X}) - \mathcal{A}(\mathbf{M})\|_2 + \lambda \|\mathbf{X}\|_*$

$$\text{minimize}_{\mathbf{X}} \; \|\mathcal{A}(\mathbf{X}) - \mathbf{y}\|_2 + \lambda \|\mathbf{X}\|_*$$

- Proximal gradient approach

$$\hat{\mathbf{X}} = \text{prox}_{\gamma}(\hat{\mathbf{X}} - \gamma \mathcal{A}^*(\mathcal{A}(\hat{\mathbf{X}}) - \mathbf{y}))$$

$$\text{prox}_{\gamma}(\hat{\mathbf{Z}}) = \arg\min_{\mathbf{X}} \|\mathbf{X} - \mathbf{Z}\|_F^2 + \lambda \|\mathbf{X}\|_*$$

Matrix Completion solvers

- Matrix Completion via ALM
  - Objective

$$\text{minimize}_{\mathbf{X}} \quad \|\mathbf{X}\|_*$$
$$\text{subject to} \quad \mathcal{A}(\mathbf{X}) = \mathcal{A}(\mathbf{M})$$

  - Reformulation

$$\text{minimize}_{\mathbf{X},\mathbf{E}} \quad \|\mathbf{X}\|_*$$
$$\text{subject to} \quad \mathbf{X} + \mathbf{E} = \mathbf{M}$$
$$\mathcal{A}(\mathbf{E}) = 0$$

CS and MC

	Sparse recovery	***Rank minimization***
Unknown	Vector *x*	Matrix *A*
Observations	*y = Ax*	*y = L[A]* (linear map)
Combinatorial objective	$\#\{\mathbf{x}_i \neq 0\} = \|\mathbf{x}\|_0$	$\mathrm{rank}(A) = \#\{\sigma_i(A) \neq 0\} = \|\sigma(A)\|_0$
Convex relaxation	$\|\mathbf{x}\|_1 = \sum_i \|\mathbf{x}_i\|$	$\|A\|_* = \sum_i \sigma_i(A)$
Algorithmic tools	Linear programming	Semidefinite programming

Yi Ma et al, "Matrix Extensions to Sparse Recovery", CVPR2009