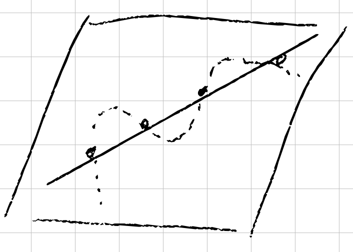
Lesson 1

- Aule e orari / Lingua di lezioni e slides
- Session: Pratiche - Portare Pc
- Dati scaricabili da sito - se non avete connessione posso usare una pennin

In science, one is frequently asked to infer or learn a model M from a given set of data points

$$\{x_1, \dots, x_n\} \subseteq \mathbb{R}^d$$



Usually there's more than one "good" model fitting the data, so what is usually sought is the "optimal" model in a set of low complexity ones. Here we focus on sets of low-dimension models, since they are fit for many purposes and obtain high-accuracy solutions to numerous problems.

Moreover, we will focus mainly on Geometric Models, leaving out the so-called Statistical ".

Statistical: x_i are usually drawn from a unknown probability distribution to be inferred. $x_i \sim \mu$

The most popular method is to maximise the likelihood over a space of parameter Θ as

$$\underset{\theta \in \Theta}{\arg\max} \prod_{i=1}^{n} \underbrace{p(x_i \mid \theta)}_{\substack{\text{Prob. of generating } x_i \\ \text{given the parameters } \theta}} =: \theta_{ML}$$

or, given a prior on Θ (i.e. how is likely for θ to happen), through Bayes rule you can look for the Maximum A Posteriori

$$\underset{\theta \in \Theta}{\arg\max} \prod_{i=1}^{N} p(x_i \mid \theta) \cdot p(\theta) =: \theta_{MAP}$$

⤳ Better with high-noise regimes

⤳ Use Geometric properties to choose the set Θ

Geometric: Exploit topological/geometrical constraints of the data, e.g. if they lay on a (affine) subspace or a submanifold that are low-dimensional. They are set to capture global algebraic/geometric/topological characteristics such as e.g. # of clusters, and to provide compact representations.

⤳ Better when the underlying geometric space is (locally) smooth.

⤳ Use statistical models to denoise the data

Depending on the problem at hand, we may want to extract different "features" and characteristics from the data. For clustering (catagorize) we only need a partition, for compression (analysis) we want some few key properties common to all data, for generating problem we need a probability to draw new data, for prediction (suggestions, ads) or completion (missing data) we may want different things etc.

Here we will talk mainly about linear models. If there's time, we'll go through some non-linear at the end. Most of the course will be dedicated to find low rank/dimension spaces where the data lies (approximately).

We also will be talking mostly about unstructured models. Some area, like control theory, communication protocol, model order reduction, have specific methods to deal with structured models that come from specific applications (Toeplitz / Hankel).

Singular Value Decomposition

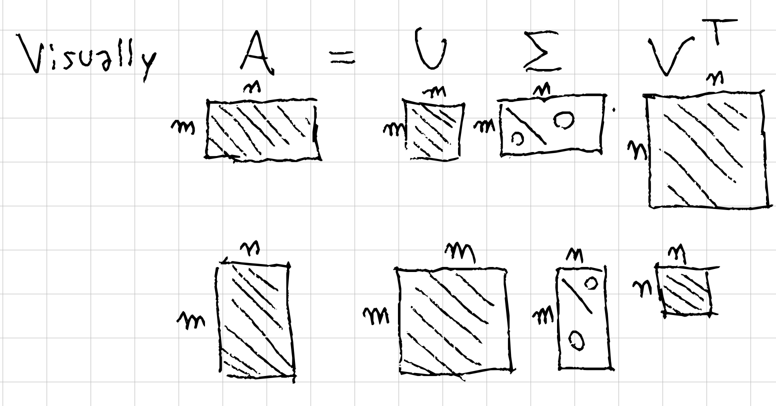
Theorem 1.1. Given $A \in \mathbb{R}^{m \times n}$ there exists U, Σ, V s.t.
$$A = U \Sigma V^T$$
and

- $U \in \mathbb{R}^{m \times m}$ orthogonal
- $V \in \mathbb{R}^{n \times n}$ "
- $\Sigma \in \mathbb{R}^{m \times n}$ and $\Sigma_{ij} = \begin{cases} 0 & i \neq j \\ \sigma_i & i = j \end{cases}$

where $\sigma_1 \geq \sigma_2 \geq .. \geq \sigma_{\min\{m,n\}} \geq 0$

Moreover, $\{\sigma_1, --, \sigma_{\min\{m,n\}}\}$ are unique and they are called **singular values** of A,

The columns of U are the **left singular vectors** of A
" " " V " " **right** " " of A



A can also be written as
$$A = \sum_{i=1}^{\min\{m,n\}} \sigma_i \cdot u_i \cdot v_i^T$$
where u_i, v_i are the left/right singular vectors

Notice that if Q is orthogonal, $U(\sigma I)V^T = (UQ)(\sigma I)(QV)^T$

Lemma 1.2 The SVD of A is unique up to orthogonal block matrices Q_U, Q_V, i.e.
$$A = U \Sigma V = U Q_U \cdot \Sigma \cdot Q_V^T V^T$$

where $Q_U = \begin{bmatrix} Q_1 & & & & \\ & Q_2 & & & \\ & & \ddots & & \\ & & & Q_k & \\ & & & & Q_0 \end{bmatrix}$ if there are k distinct non-zero s.v. in A, and $Q_i \in \mathbb{R}^{n_i \times n_i}$ orth. where n_i is the multipl. of the i-th distinct s.v.

$Q_V = \begin{bmatrix} Q_1 & & & & \\ & Q_2 & & & \\ & & \ddots & & \\ & & & Q_k & \\ & & & & \tilde{Q}_0 \end{bmatrix}$

Corollary 1.3 If $\sigma = \sigma_i$ is a simple s.v. of A, then the associated left/right sing. vectors are $\gamma u_i, \gamma \cdot v_i$ for a sign $\gamma = \pm 1$.

(Theorem of Eckart-Young)

Theorem 1.4 Given a matrix $A \in \mathbb{R}^{m \times n}$ and a number $k \leq \min\{m,n\}$, then
$$\min_{\substack{X \in \mathbb{R}^{m \times n} \\ rk(X) \leq k}} \|A - X\|_F^2 \quad / \quad \min_{\substack{X \in \mathbb{R}^{m \times n} \\ rk(X) \leq k}} \|A - X\|$$
are both solved by the matrix A_k obtained through SVD as $A = U\Sigma V^T \leadsto A_k = U \Sigma_k V^T$

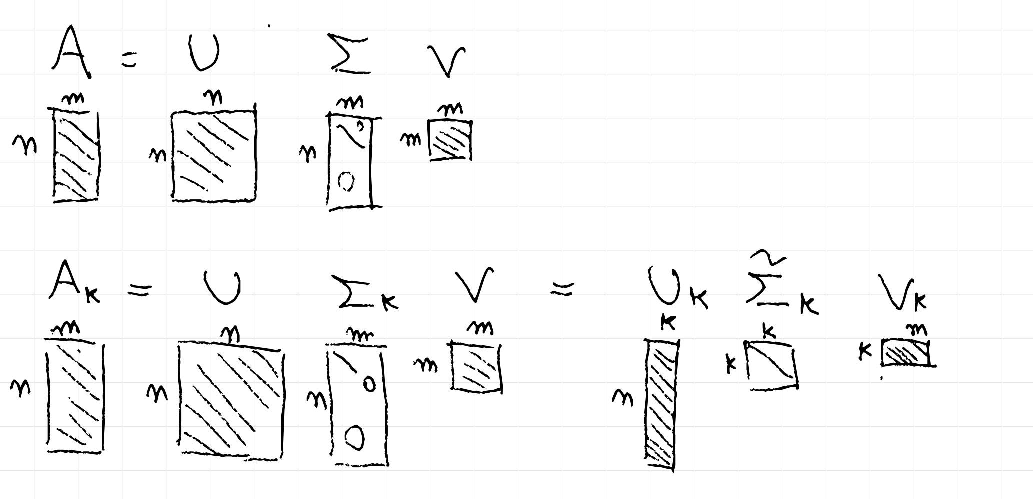
where $\Sigma_k = \text{diag}\{\sigma_1, \sigma_2, --, \sigma_k, 0, --, 0\}$.

A_k is called the **k-truncated SVD** of A.

The minimum is equal to
$$\|A - A_k\|_F^2 = \sigma_{k+1}^2 + .. + \sigma_{\min\{m,n\}}^2, \quad \|A - A_k\| = \sigma_{k+1}$$

Lemma 1.5 If $A \in \mathbb{R}^{n \times n}$ is positive semi-definite (PSD) then in the SVD $A = U\Sigma V^T$ one can take $U = V$ and find the eigendecomposition (EVD) $A = U \Sigma U^T$

Given $A = U\Sigma V^T \in \mathbb{R}^{n \times m}$, the k-truncated $A_k = U \Sigma_k V^T$ can be espressed with the **k-reduced** (or k-compact)
$$A_k = U_k \tilde{\Sigma}_k V_k^T, \quad U_k \in \mathbb{R}^{n \times k}, \tilde{\Sigma}_k \in \mathbb{R}^{k \times k}, V_k \in \mathbb{R}^{m \times k}$$
where $\Sigma_k = \begin{bmatrix} \tilde{\Sigma}_k & 0 \\ 0 & 0 \end{bmatrix}$ so $\tilde{\Sigma}_k$ is diagonal nonnegative and U_k, V_k are the first k columns of U, V.



Notice that A has nm entries, A_k is an approximation. of A, but A_k is computable by $U_k, \tilde{\Sigma}_k, V_k$ that have $(n + m + 1)k$ nonzero entries that is way less than $n \cdot m$ when $k \ll \min\{n,m\}$ ($k \leq \frac{2}{5} \min\{n,m\}$ is enough, in general the gain is $O(\frac{\min\{n,m\}}{k})$)

$\leadsto$ This can be seen as a <u>Compression Technique</u>.

We will see other more sophisticated methods, but most will follow the same paradigm: given the data X, decompose it as $X \sim A_1 \cdot A_2 \cdot A_3 \cdot .. \cdot A_k$ where
$$\sum_i nnz(A_i) \ll nnz(X)$$

$\leadsto$ In case of $\overset{n \times m}{X} \sim \overset{n \times k}{U} \cdot \overset{k \times m}{Y}$ where $k \ll \min\{n,m\}$

the columns of U can be interpreted as the 'key features' that compose the data $\{x_1, --, x_n\}$. In fact
$$x_i \sim U \cdot y_i = \sum_{j=1}^{k} u_j \cdot y_{ji}$$
meaning that x_i is a combination of $\{u_1, .., u_k\}$ with scalar coefficients $y_{ji} \in \mathbb{R}$.

There are several algorithm to compute the k-truncated SVD of A, like the Orthogonal Power Iteration (Golub, Loan '96) (Lanczos '50) or Power Factorization (PF, Hartley, Schaffalitzky '03). In general they are iterative methods converging with speed
$$O(\rho^{2 \cdot d}) \quad \text{where } \rho = \frac{\sigma_{k+1}}{\sigma_k}$$

$\leadsto$ Matlab uses a Lanczos Bidiagonalization Method (Larsen '98) [Gc]

Model Selection

In what we will discuss, we usually set a parameter k (rank or complexity) that defines the dimension of the reduced space where we suppose the data approximately lie. In decomposition/factorization it is the smallest size of the wanted matrices, e.g. $\overset{n\times m}{X} \sim \overset{n\times k}{U}\overset{k\times k}{\Sigma}\overset{k\times m}{V^T}$ the k-reduced SVD. But how to find the optimal k?

$\rightsquigarrow$ k must be small to ensure compression of the data and 'meaningfulness' of the decomposition.

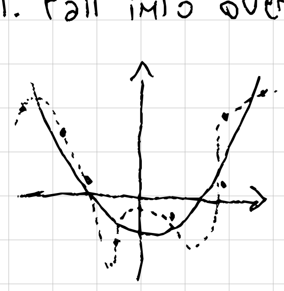
e.g. $X = I\cdot X$ gives us no information on X

$\rightsquigarrow$ k must not be too small if we do not want to lose accuracy and important info on X.

In particular if the error $\|X - X_k\|_F$ is too large, the data in X_k may lose important features present in X

$\rightsquigarrow$ k must not be too large to not. fall into overfitting

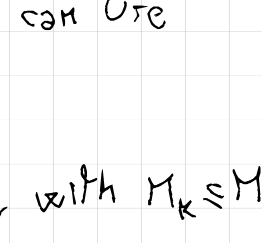
e.g. Given n real points, there exists always a $(n-1)$-degree polynomial interpolating them exactly, but it's more probable that the data is distributed according to a less complex polynomial with some perturbation.



For specific models we will discuss some ad hoc techniques to choose k, but in general we can use some good Empirical methods.

L-curve: Given $f(k) = \min_{\mathcal{M}\in M_k} err(X,\mathcal{M})$, with $M_k \subseteq M_{k+1}$

Then $f(k)$ is decreasing. Its plot in most applications presents an L-shape with a sharp change in derivative on one or few points, where the plot reaches the 'knee' of the L. k^* is usually chosen as the knee.



Information Criterion: Given $I(k)$ the number of parameters needed to express a generic model in M_k, k^* will minimize $f(k) + I(k)\cdot\alpha$ for some $\alpha > 0$. Since $I(k)$ increases in k, it balances the decreasing error $f(k)$.

Thresholding: Fix a tolerance for the error τ and take k^* as the minimum k s.t. $f(k) \leq \tau$.

Which one is better? None, it depends on the models and what info you need to extract from the data.

(For the exam: using stat. models, one can prove more. Given $\{x_1,..,x_n\} \subseteq \mathbb{R}^d$ s.t. $X_0 = [x_1,..,x_n]$ is low rank and $\omega > 0$, E random matrix with 0 mean, $1/\sqrt{n}$ var. and $X = X_0 + \omega E$, suppose $n \to \infty$, $d/n \to \beta$. Then the thresholding minimizing $\lim_{n\to\infty} \|X_\tau - X_0\|$ is

$$\tau^* = \omega\cdot\sqrt{2(\beta+1) + \frac{8\beta}{\beta+1+\sqrt{\beta^2+14\beta+1}}}$$

If $\beta = 1$, then $\tau^* = \frac{4}{\sqrt{3}}\omega$ [2e])

Suppose now we just want to find the numerical rank of a matrix A, i.e. the thresholding truncation for a tolerance τ, A classical algorithm is the QR with pivoting:

Fixing $l > 0$, the QR is an iterative procedure that gives

$$\overset{d\times n}{A} = \overset{d\times l}{Q}\overset{l\times n}{R} + E, \quad \|E\|_F \leq \tau, \quad Q \text{ with orth. columns}$$
$$R \text{ upper-triang. up to a perm. of columns}$$

This is called Rank Revealing QR because for l large enough, the number of steps k in the iterative algorithm corresponds to the numerical rank of A, and the cost is $O(lkn)$

From QR, a k-truncated SVD takes only $O(kdn)$. Krylov methods have the same complexity but are in general less robust.

Principal Compoment Analysis (PCA)

Given $\{x_1, \dots, x_n\} \subseteq \mathbb{R}^d$ and $k \leq d$, find $U \in \mathbb{R}^{d \times k}$, $\mu \in \mathbb{R}^d$ and $y_i \in \mathbb{R}^k$ that minimize

$$\sum_{i=1}^{n} \| x_i - \mu - U y_i \|^2$$

⤳ oldest and best known multivariate analysis technique

Theorem 5.1 A PCA of $\{x_1, \dots, x_n\} \subseteq \mathbb{R}^d$ is attained by $\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$, $\hat{X} = [\hat{x}_1, \dots, \hat{x}_n]$, $\hat{x}_i = x_i - \mu$,

$\hat{X} = \tilde{U} \tilde{\Sigma} \tilde{V}^T$ SVD, $\hat{X}_k = U \tilde{\Sigma}_k \hat{V}_k^T$ k-reduced SVD,

$y_i = (\tilde{\Sigma}_k V_k^+)_i$

Proof Let $Y = [y_1, \dots, y_n]$, $X = [x_1, \dots, x_n]$.

$$\sum_{i=1}^{n} \| x_i - \mu - U y_i \|^2 = \| X - \mu e^T - U Y \|_F^2$$

Fixing U, Y the derivative wrt μ gives

$$(A - \mu e^T) e = 0 \Rightarrow \mu = \tfrac{1}{n} A e, \quad A = X - UY$$

⤳ we need to minimize $\| (X - UY)(I - \frac{1}{n} e e^T) \|_F^2$

or $\| \hat{X} - U \hat{Y} \|_F^2$ where $\hat{X} = X (I - \frac{1}{n} e e^T)$ and $\hat{Y}$ is any $\hat{Y}$ s.t. $\hat{Y} e = 0$.

Notice that $rk(U\hat{Y}) \leq k$, so given the k-reduced SVD of $\hat{X} = U(\Sigma V^T)$, we have $\hat{Y} = \Sigma V^T$ since $\hat{X} e = 0 = U(\Sigma V^T e) = U \cdot \hat{Y} e \Rightarrow \hat{Y} e = 0$.

Eventually, $\mu = \frac{1}{n}(X - UY)e = \frac{1}{n}(X - U\hat{Y})e = \frac{1}{n}(X - \hat{X})e$

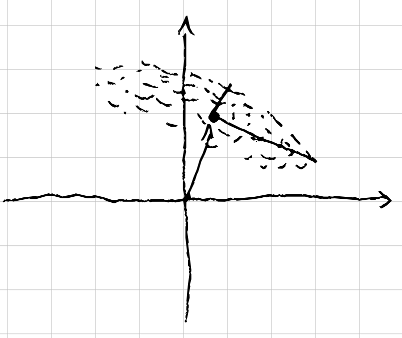
$= \frac{1}{n}(X - X + \frac{1}{n} X e e^T) e = \frac{1}{n} X e$ ▨

This in particular says that

- The optimal μ is the average $\mu = \frac{1}{n} X e$
- We are fitting the 0-mean data $\hat{X} = X - \mu e^T$ on a proper subspace whose basis is U

⤳ PCA is a k-$\overset{\text{low}}{\text{rank}}$ approximation of centered data

⤳ it coincides with the statistical model, whose aim is to find the directions with most 'variance'



⤳ the SVD is tuned to minimize $\|\cdot\|$ and $\|\cdot\|_F$, that fit the Gaussian noise. For other kind of noise, other norm may be better suited such as $\|\cdot\|_1$, $\|\cdot\|_\infty$, $\|\cdot\|_p$, etc. So the SVD may not be the answer to all problems. For example, uniform distribution noise is better removed using ℓ^∞ norm.

⤳ Once we have μ, U with U column orth, and a new data z, we can determine if z is an outlier or not. In fact, if z is an inlier, then $z \sim \mu + Uy$ for some y, or also said, the "projection of $z-\mu$ on $\mathrm{Span}(U)^\perp$" is close to 0, i.e. $\| U^T(z-\mu) \| \sim 0$.

If instead of U we have a low-rank L, we take the SVD $L = U \Sigma V^+$ truncated to k, and repeat the same.

This is similar to training a NN.

Incomplete PCA (Missing Entries)

Suppose the data matrix $X \in \mathbb{R}^{d \times n}$ has some missing entries, i.e. we have $W \in \{0,1\}^{d \times n}$ such that

$W_{i,j} = 1 \Rightarrow x_{i,j}$ is observed

$W_{i,j} = 0 \Rightarrow x_{i,j}$ is unobserved

Netflix 2016: They offered 1M dollars for improving by 10% the Reccomandation system ← won by Matrix fact. with low rank methods

→ Missing entries: each user has seen / rated few movies

[12e]

Given k, we want to find an affine subspace of dimension k S with translation μ and basis U such that there exists a completion of X with columns approximately inside S.

i.e.

$$\min_{\mu, U, Y} \| W \circ (X - \mu - UY) \|_F^2$$

$$= \min_{\mu, U, Y} \sum_{\substack{i,j: \\ w_{i,j}=1}} (x_{ij} - \mu_i - (UY)_{ij})^2$$

"Matrix Completion"

Forgetting μ for a moment, we need to find $\min_A \operatorname{rk}(A) : X = A$ on Ω

This problem, like many other involving $\operatorname{rk}(A)$, is NP-Hard, so we need some relaxation.

Nuclear Norm Given a matrix $A \in \mathbb{R}^{n \times m}$ with singular values $\sigma_1, \dots, \sigma_{\min\{n,m\}}$, then the Nuclear Norm of A (or 1-Schatten norm) is

$$\|A\|_* := \sum_{i=1}^{\min\{m,n\}} \sigma_i$$

Convex Envelope : Given a function $f : C \to \mathbb{R}$ where $C \subseteq \mathbb{R}^d$ is a convex set, the convex envelope of f over C is $\mathrm{Conv}_C(f) = g : C \to \mathbb{R}$ convex such that $\forall h : C \to \mathbb{R}$ convex such that $h \leq f$, we have $h \leq g \leq f$ or also

$$g(x) := \sup\{h(x) \mid h \leq f \text{ on } C,\ h \text{ convex on } C\}$$

Optimising quantities involving $rk(A)$ is usually NP-hard, so it is common to relax the rank with its convex envelope.

Theorem 2.1 The convex envelope of $\mathrm{rank}(A)$ in the domain $\{A : \|A\| \leq 1\}$ is $\|A\|_*$ (Fazel, '02)

proof for $C = \{A : \|A\|_* \leq 1\}$

First of all, notice that $\|A\|_* \leq 1 \Rightarrow \|A\|_* \leq \mathrm{rank}(A)$. Moreover, if $\mathrm{rank}(A) = 1$, $\sigma_1(A) = 1$, then $A \in C$. We can thus take $A \in C$ with SVD $A = \sum_i \sigma_i u_i v_i^T$ and notice that $\|A\|_* = \sum_i \sigma_i \leq 1$ and $u_i v_i^T \in C \ \forall i$ and $0 \in C$. If $f := \mathrm{Conv}_C(\mathrm{rank})$ then f is convex, so

$$\begin{aligned} f(A) &= f\Big(\sum_i \sigma_i (u_i v_i^T) + (1 - \|A\|_*) \cdot 0\Big) \\ &\leq \sum_i \sigma_i f(u_i v_i^T) + (1 - \|A\|_*) f(0) \\ &\leq \sum_i \sigma_i \mathrm{rank}(u_i v_i^T) + (1 - \|A\|_*) \mathrm{rank}(0) \\ &= \sum_i \sigma_i = \|A\|_* \end{aligned}$$

Since $\|\cdot\|_*$ is a norm, it is a convex function and thus $\|\cdot\|_*$ is the convex envelope of rank ▨

0-Norm : Given $x \in \mathbb{R}^d$, $\|x\|_0 := |\{i \mid x_i \neq 0\}|$

WARNING : This is <u>NOT</u> a norm since $\|\lambda x\|_0 = \|x\|_0 \ \forall \lambda \neq 0$

Corollary 2.2 $\mathrm{Conv}_C(\|\cdot\|_0) = \|\cdot\|_1$ for $C = \{\|x\|_\infty \leq 1\}$

[Not easy to prove]

Let's go back to $W \in \{0,1\}^{d \times n}$ being the matrix saying if an entry is observed or not.

$\rightsquigarrow$ If W is too sparse, the solution is far from unique, and the best k may be lower than wanted

$\rightsquigarrow$ If the 'underlying' true X is too sparse, even few unobserved entries may modify the solution greatly

e.e.g. $X = e_1 e_1^T = \begin{bmatrix} 1 & \\ & \end{bmatrix}$. If $w_{1,1} = 0$, the optimal solution will be the zero matrix.

$\rightsquigarrow$ If the pattern of W is too 'structured', the solution is also likely to change

e.g. $W = \begin{bmatrix} 0 & - & 0 \\ & 1 & \end{bmatrix} \Rightarrow$ we lose all info on the first component and μ_1, $U_{1,:}$ will never be able to approximate the true X.

To solve this problems we will

- bound the number of 0 in W and in X
- suppose that the pattern of 0 in W is random

Incoherence wrt Sparse Matrices

A matrix is incoherent when it is "uniformally dense". More specifically, we say that a matrix $X \in \mathbb{R}^{d \times n}$ of rank k and k-reduced SVD $X = U \Sigma V^T$ is **ν-incoherent** if

$$\max_i \|U_{i,:}\|^2 \leq \nu \sqrt{\frac{k}{d}} \qquad \max_j \|V_{j,:}\|^2 \leq \nu \sqrt{\frac{k}{d}}$$

$$\|UV^T\|_\infty \leq \nu \sqrt{\frac{k}{dn}}$$

The idea is that each element of U, V is bounded in abs. value by 1, and when this is reached it is the only nonzero on the associated row and column, so the matrix is already quite sparse. On the contrary, a column of U with all elements roughly of the same magnitude, will have entries with abs. value $\sim \sqrt{\frac{1}{d}}$, so the rows of a balanced full matrix will have norm $\sqrt{k/d} < 1$. In this sense, a bigger ν leads to a potentially sparser matrix. (This has sense on a statistical ground)

Theorem 3.1 Suppose $X \in \mathbb{R}^{d \times n}$ ($n \geq d$) with rank k and ν-incoherent. Suppose we sample at random M entries from X with $M \geq C \cdot \nu^4 \cdot k\, n \log(n)^2$.
Then with probability $\geq 1 - 1/n^3$ X is the unique solution of

$$\min_A \|A\|_* \quad \text{s.t.} \quad A = X \text{ on the observed } M \text{ entries}$$

Notice that this is a convex problem, and it is a relaxation of the NP-hard problem

$$\min_A \operatorname{rank}(A) \text{ s.t. } \quad '' \quad '' \quad '' \qquad '' \quad ''$$

A similar result also holds for Compressive Sensing of low-rank matrices, i.e. $\min_A \|A\|_* : \mathcal{P}(A) = \mathcal{P}(X)$ as long as the linear operator $\mathcal{P}$ is "incoherent" with A

(For the exam: This can be found in [3e])

There are specific algorithms to solve IPCA (for the exam: Proximal Gradient (Cai, 2008) Power Factorization/Lanczos Method (Hartley, Schaffalitzky '03) Partition Alt. Min. (Jain, '12)

Thus, given the model

$$\min_{\mu,U,Y} \| W \circ (X - \mu e^t - UY) \|_F^2$$

we formulate an Alternating Method to solve it:

Given $f : \Omega := \Omega_1 \times \dots \times \Omega_m \to \mathbb{R}$ in $C^1(\Omega)$, $\sum n_i = n$, where $\Omega_i \subseteq \mathbb{R}^{n_i}$ are closed and convex, we want to minimize f over its domain Ω. If $x \in \Omega$, then $x = (x_i)_{i=1,\dots,m}$ where $x_i \in \Omega_i \subseteq \mathbb{R}^{n_i}$. A general framework of alternating method is

Alternating Method

```
Initialize $x^{(0)}$ randomly in $\Omega$
Repeat until convergence
    For every $i = 1, \dots, m$
        $x_i^{(k+1)} = \arg\min_x f(x_1^{(k+1)}, \dots, x_{i-1}^{(k+1)}, x, x_{i+1}^{(k)}, \dots, x_m^{(k)})$
```

Theorem 6.1 [4] Suppose all the min problem solved in an Alt. Met. have an unique minimum. Then every limit point of $x^{(k)}$ is a local minimizer of $f(x)$ over Ω. If $m=2$ the unicity is not required.

In particular, the hypothesis is fulfilled whenever f is strongly convex in every x_i.

AM for IPCA

Let $X \in \mathbb{R}^{d \times n}$ and $W \in \{0,1\}^{d \times n}$. The IPCA is

$$\min_{\mu,U,Y} \| W \circ (X - \mu e^T - UY) \|_F^2 = \min_{\mu,U,Y} \sum_{\substack{i,j: \\ w_{ij}=1}} (x_{ij} - \mu_i - (UY)_{ij})^2$$

where $U \in \mathbb{R}^{d \times k}$, $Y \in \mathbb{R}^{k \times n}$, $\mu \in \mathbb{R}^d$. One can see this as a function in $d + k(d+n)$ variables (or less depending on W). Notice that we do not know all x_{ij} but only the observed ones ($w_{i,j} = 1$) so we cannot use a truncated SVD. But we can optimize in an alternating fashion fixing everything but one among μ, u_i^T (rows of U), y_j (columns of Y). In particular, taking the derivatives we find that

$$\partial_{\mu_i} = -2 \sum_j w_{ij} (x_{ij} - \mu_i - (UY)_{ij}) = 0 \Rightarrow \mu_i = \frac{\sum_j w_{ij} (X - UY)_{ij}}{\sum_j w_{ij}}$$

$$\partial_{u_i} = -2 \sum_j w_{ij} (x_{ij} - \mu_i - u_i^T y_j) \cdot (-y_j) = 0$$

$$\Rightarrow \left(\sum_j w_{ij} \, y_j y_j^T \right) u_i = \sum_j w_{ij} (x_{ij} - \mu_i) y_j$$

$$\Rightarrow u_i = \left(\sum_j w_{ij} y_j y_j^T \right)^{-1} \sum_j w_{ij} (x_{ij} - \mu_i) y_j$$

$$\partial_{y_j} = -2 \sum_i w_{ij} (x_{ij} - \mu_i - y_j^T u_i)(-u_i) = 0$$

$$\Rightarrow y_j = \left(\sum_i w_{ij} u_i u_i^T \right)^{-1} \sum_i w_{ij} (x_{ij} - \mu_i) \cdot u_i$$

A problem to address is the loss of unique decomposition $X \sim \mu e^T + UY$. In fact, for example

$$\mu e^T + UY = (\mu + Uv) e^T + U(Y - ve^T) = \mu + URR^{-1}Y$$

A common choice is $U^T U = I$ that can be achieved through a reduced QR $U = QR = ▯▫$ (that theoretically is a reduced SVD, but algorithmically it's better to compute) and $Ye = 0$ that can be achieved with $Y \mapsto Y - Yee^T/n$ and it has sense statistically since μ is usually the mean of X and UY is zero-mean.

Power Iteration [7e]

$Y(X - \mu e^T)^T$

```
Initialize $Y, U$
Repeat until convergence
  Power_Iteration_step
    $\mu_i \leftarrow \frac{\sum_j w_{ij}(x_{ij} - u_i^T y_j)}{\sum_j w_{ij}}$    $\forall i$
    $u_i \leftarrow \left( \sum_j w_{ij} y_j y_j^T \right)^{-1} \sum_j w_{ij} (x_{ij} - \mu_i) y_j$    $\forall i$
    $U \xleftarrow{Q} \begin{bmatrix} u_1^{(k+1)} \\ \vdots \\ u_d^{(k+1)} \end{bmatrix} = QR$    reduced QR
    $y_j \leftarrow \left( \sum_i w_{ij} u_i^T u_i \right)^{-1} \sum_i w_{ij} (x_{ij} - \mu_i) \cdot u_i^T$    $\forall j$
Return $\mu + \frac{1}{n} UYe$, $U$, $Y(I - ee^T/n)$
```

This is fairly expensive $O(ndk)$, but not too much. The inv. are on $d \times d$ matr. Notice moreover that you need k.

→ We are going now to tackle a more general problem: Sparse/Robust PCA.

Lesson 2

Robust PCA

Suppose the data matrix $X \in \mathbb{R}^{d \times m}$ has corrupted entries, i.e. there exists a matrix $E \in \mathbb{R}^{d \times m}$, usually sparse, and a $\overset{\text{(affine)}}{\text{low}}$ rank $L \in \mathbb{R}^{d \times m}$ such that

$$X \sim L + E$$

and we want to retrieve L.

$\rightsquigarrow$ This is harder than Incomplete PCA because we don't know the sparsity pattern of E. Moreover, if we take in IPCA W and set

$$x_{i,j} = \lambda \gg \|X_{\text{obs.}}\|_\infty \quad \forall i,j : w_{i,j} = 0$$

Then solving RPCA on X gives us the solution of IPCA.

It is not easy to formulate a model, since we want to minimize $\|X - L - E\|$, $\dim(L)$, $\|E\|_0$.

- A model is a generalization of IPCA fixing k

$$\min_{\mu, U, Y} \|W \circ (X - \mu - UY)\|_F^2 \quad : W = f(\mu, U, Y)$$

This is the weight approach and ideally we want $W_{i,j} \sim 1$ when $(X - \mu - UY)_{ij} \sim 0$ and $w_{i,j} = 0$ otherwise. One way to define it is

$$E = X - \mu - Uy$$
$$w_{i,j}^2 = \rho(E_{i,j}) / E_{i,j}^2$$

M-Estimators
"
Maximal-Likelihood
-type

where $\rho(x)$ is a 'loss function' s.t. $w_{i,j}$ is decr. in $|x|$ and ≥ 0

In this case, the problem is rewritten as

$$\min_{\mu, U, Y} \sum_{i,j} \rho(x_{i,j} - \mu_i - (UY)_{ij})$$

$\rightsquigarrow$ $\rho(x) = x^2$ is the standard PCA

$\rightsquigarrow$ other loss funct. are $|x|$, $x_0^2 \log\left(1 + \frac{x^2}{x_0^2}\right)$, etc.

When W is fixed, this is an IPCA so a classical algorithm is to alternatively solve the IPCA and then set $w_{i,j} = \sqrt{\rho(E_{i,j}) / E_{i,j}^2}$.

Robust PCA with Power Iteration

Here we have X close to $\mu e^\top + UY$ up to some entries that may be very large, so we want to find μ, U, Y and W where

$$W_{ij} \sim 0 \quad \text{iff} \quad (X - \mu e^\top - UY)_{ij} = E_{ij} \text{ is large}$$

The idea is to define $W_{ij} = \frac{\varepsilon_0^2}{E_{ij}^2 + \varepsilon_0^2}$ and minimise $\sum_{i,j} W_{ij} E_{ij}^2$ as in IPCA over all μ, U, Y and then reupdate W.

Iteratively Reweighted Least Squares (IRLS)

```
Initialize $\mu, U, Y$ as classic PCA of $X$, $\varepsilon_0 > 0$
Repeat until convergence
    $E = X - \mu e^\top - UY$,   $w_{i,j} = \frac{\varepsilon_0^2}{E_{ij}^2 + \varepsilon_0^2} \quad \forall i,j$
    $(\mu, U, Y) = \text{Power\_Iteration\_Step}(\mu, U, Y, W)$
Return $\mu + \frac{1}{n} UYe$, $U$, $Y(I - ee^\top/n)$, $E = X - \mu e^\top - UY$
```

Notice that the first step is the same as saying $w_{i,j} = 1 \ \forall i j$.

- A different model is fixing $\varepsilon > 0$ and solving

$$\min_{\mu, L, E} \ \text{rank}(L) + \lambda \|E\|_0 \quad : \|X - \mu - L - E\| \leq \varepsilon$$

where, in statistical terms, ε is a bound on the noise variance. Let's now discuss the exact case and $\mu = 0$

As usual, this is not convex, so we relax it into the PCP

Principal Component Pursuit

$$\min_{\mu, L, E} \ \|L\|_* + \lambda \|E\|_1 \ {}^{\swarrow \text{vector norm } 1} \quad ; \quad X = L + E$$

[Le] **Theorem 3.2** Given $X = L_0 + E_0$ with L_0 ν-incoherent and s.t. $\text{supp}(E_0)$ is unif. distr. among all patterns with fixed $m = \|E_0\|_0$. If there exist costants ρ_d, ρ_s such that

$$\text{rk}(L_0) \leq \frac{\rho_d \min\{d, n\}}{\nu \log(\max\{d, n\})^2}, \quad m \leq \rho_s \cdot nd$$

Then there exists a constant c such that the solution to PCP

$$\min_{L, E} \ \|L\|_* + \|E\|_0 \frac{1}{\sqrt{\max\{n, d\}}} \quad ; \quad X = L + E$$

is the exact (L_0, E_0) with prob. $\geq 1 - c \max\{n, d\}^{-10}$.

Warning: Even if $\|\cdot\|_*$, $\|\cdot\|_{2,1}$, $\|\cdot\|_1$ are norms and thus convex they are NOT differentiable, so gradient methods are not guaranteed to converge. One can turn to semidefinite programming, but those are typically very expensive.

The ADMM uses subgradient, but usually there's more than one subg. that get the derivate to zero, so it suffers from this ambiguity.
This is usually why one turns to simpler AM.

When dealing with models using $\|\cdot\|_*$, $\|\cdot\|_1$ we may want to use gradient descend methods, but those norms are NON-diff., so we need to resort to subdifferentials.

<u>Subdifferential</u> : $v \in \partial_{sub} f(x_0)$, f convex, if

$$f(x) \geq f(x_0) + v^T(x - x_0) \quad \forall x$$

⇝ For the minimum we have $0 \in \partial_{sub} f(x_0)$, and it's an iff.

Lemma 2.3 $v \in \partial_{sub} \|x\|_1 \iff v_i = \begin{cases} \operatorname{sgn}(x_i) & x_i \neq 0 \\ |\alpha_i| \leq 1 & x_i = 0 \end{cases}$

Proof Suppose $v \in \partial_{sub} \|x\|_1$. If $x_i \neq 0$, then for $|\gamma| < |x_i|$ we have $\|x + \gamma e_i\|_1 = \|x\|_1 + \operatorname{sgn}(x_i)\gamma \geq \|x\|_1 + \gamma v^T e_i$, so

$$\gamma\,[\operatorname{sgn}(x_i) - v_i] \geq 0 \quad \forall \gamma \in [-|x_i|, |x_i|] \Rightarrow v_i = \operatorname{sgn}(x_i)$$

If $x_i = 0$, then $\|x + \gamma e_i\|_1 = \|x\|_1 + |\gamma| \geq \|x\|_1 + \gamma v^T e_i \quad \forall \gamma$

so if $\gamma = 1$, $1 \geq v^T e_i$ and if $\gamma = -1$, $v^T e_i \geq -1 \Rightarrow |v_i| \leq 1$.

To prove that it is sufficient, notice that

$$\|x\|_1 + v^T y = \sum_i |x_i| + v_i \cdot y_i = \sum_{x_i = 0} v_i \cdot y_i + \sum_{x_i \neq 0} \operatorname{sgn}(x_i)(x_i + y_i)$$

$$\leq \sum_i |x_i + y_i| = \|x + y\|_1 \qquad ▨$$

Theorem 2.4 Given $X_0 = U_0 \Sigma_0 V_0^T$ SVD, then

$$S \in \partial_{sub} \|X_0\|_* \iff S = U_0 \left[\begin{array}{c|c} \begin{smallmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{smallmatrix} & \\ \hline & W \end{array}\right] V_0^T$$

where W corresponds to the zero sv block and $\|W\| \leq 1$

Proof ($\Rightarrow$) Given a s.v. $\sigma_i \neq 0$ of X_0, let $|\gamma| < \sigma_i$. Then

$$\|X_0 + U_0(\Sigma_0 + \gamma E_{ii})V_0^T\|_* = \|X_0\|_* + \gamma \geq \|X_0\|_* + \langle S, V_0 E_{ii} V_0^T\rangle \gamma$$

$$\Rightarrow \gamma\,[1 - \langle U_0^T S V_0, E_{ii}\rangle] = \gamma\,(1 - (U_0^T S V_0)_{ii}) \geq 0 \quad \forall |\gamma| < \sigma_i$$

Since γ can be pos. and neg., then $1 = (U_0^T S V_0)_{ii}$. Take now $\sigma_i \neq 0$ and σ_j that may be also zero. Then

$$\|X_0 + U_0(\Sigma_0 + \gamma E_{ij})V_0^T\|_* = \|X_0\|_* + \underbrace{\left\|\begin{bmatrix} \sigma_i & \\ \gamma & \sigma_j \end{bmatrix}\right\|_* - \left\|\begin{bmatrix} \sigma_i & \\ & \sigma_j \end{bmatrix}\right\|_*}$$

$$\geq \|X_0\|_* + \gamma \langle S, U_0 E_{ij} V_0^T S\rangle \Rightarrow M_{ij}(\gamma) \geq \gamma\,(U_0^T S V_0)_{ij} \quad \forall \gamma$$

but $M_{ij}(\gamma) \leq \left(\left\|\begin{bmatrix} 1 & \\ \gamma & 1 \end{bmatrix}\right\|_* - \|I\|_* \right) \left\|\begin{bmatrix} \sigma_i & \\ & \sigma_j \end{bmatrix}\right\|_* = \underset{\gamma \to 0}{O}(\gamma^2)$

$\Rightarrow (U_0^T S V_0)_{ij}\,\gamma = \underset{\gamma \to 0}{O}(\gamma^2)$ so taking γ pos. and neg. we find

$(U_0^T S V_0)_{ij} = 0$. Now call $\tilde{P} = \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & P \end{array}\right]$ and $U_0^T S V_0 = \left[\begin{array}{c|c} * & * \\ \hline * & W \end{array}\right]$ so that

$$\|X_0 + U_0(\Sigma_0 + \tilde{P})V_0^T\|_* = \|X_0\|_* + \|P\|_* \geq \|X_0\|_* + \langle U_0^T S V_0, \tilde{P}\rangle$$

$\Rightarrow \|P\|_* \geq \langle W, P\rangle \quad \forall P$. Let $W = U \Sigma V^T$ be its SVD and suppose $P = U D V^T$ where $D = \operatorname{diag}(d)$ and d is not necess. nonnegative

$$\|D\|_* = \|d\|_1 \geq \langle \Sigma_W, D\rangle = d^T \sigma_W \;\; \forall d \Rightarrow \sigma_W \in \partial_{sub} \|0\|_1$$

$\Rightarrow (\sigma_W)_i \leq 1 \;\; \forall i \Rightarrow \|W\| \leq 1$.

($\Leftarrow$) $\|X_0\|_* + \langle S, X - X_0\rangle = \|\Sigma_0\|_* + \left\langle \left[\begin{array}{c|c} \begin{smallmatrix} 1 & \\ & \ddots \end{smallmatrix} & \\ \hline & W \end{array}\right], U_0^T X V_0 - \Sigma_0 \right\rangle$

$= \left\langle \left[\begin{array}{c|c} \begin{smallmatrix} 1 & \\ & \ddots \end{smallmatrix} & \\ \hline & W \end{array}\right], U_0^T X V_0 \right\rangle = \langle \Sigma_S, \tilde{X}\rangle$ where $\|\Sigma_S\| \leq 1$

notice that $\langle \Sigma_S, \tilde{X}\rangle = \sigma_S^T \operatorname{Diag}(\tilde{X}) \leq e^T |\operatorname{Diag}(\tilde{X})|$, but to change sign to the diagonal of $\tilde{X}$ it is enough to multiply by a diagonal sign matrix that is unitary so it is equal to $e^T \operatorname{Diag}(\hat{X})$ where $\hat{X}$ has the same sv of X.

$$e^T \operatorname{Diag}(U \Sigma_X V^T) = \sum_{ij} u_{ij} \sigma_j^X v_{ij} = \sum_j \sigma_j^X \sum_i u_{ij} v_{ij} = \sum_j \sigma_j^X \cdot (U^T V)_{jj}$$

$$\leq \sum_j \sigma_j^X = \|X\|_*$$ because $U^T V$ is unitary, so $e_j^T (U^T V) e_j \leq \|U^T V\| = 1$

▨

Theorem 2.5 $\partial_{sub} \|v\| = v / \|v\|$ if $v \neq 0$ and $\partial_{sub} \|0\| = \{x : \|x\| \leq 1\}$

Proof $\nabla_x \|x\| = \nabla_x \sqrt{\sum x_i^2} = \dfrac{2x}{2\sqrt{\sum x_i^2}} = \dfrac{x}{\|x\|}$ if $\|x\| \neq 0$. Instead

$v \in \partial_{sub} \|0\| \iff \|x\| \geq \langle x, v\rangle \;\; \forall x \iff \|v\| \leq 1$ ▨

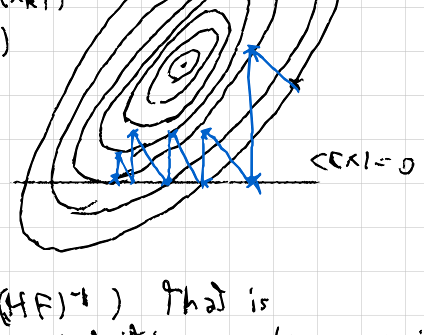
Lagrange and Penalization [1,2]

Given the optimization problem ($\phi : \mathbb{R}^n \to \mathbb{R}$, $c : \mathbb{R}^n \to \mathbb{R}^m$)

$$\min_x \phi(x) \quad \text{s.t.} \quad c(x) = 0 \qquad (1)$$

there are many techniques to approach it.

- When we have a projection $P : \mathbb{R}^n \to \mathbb{R}^n$ on the set $\{x : c(x) = 0\}$, we can use a (sub)-gradient method $\tilde{x}_{k+1} = x_k - \alpha(\nabla_x \phi(x_k))$ and after each step project $x_{k+1} = P(\hat{x}_{k+1})$, where α is taken as the best possible, i.e. $\partial/\partial\gamma\, \phi(x_k + \gamma \nabla_x \phi(x_k)) = 0$ equivalent to $\nabla_x \phi(x_k) \perp \nabla_x \phi(x_k + \alpha \nabla_x \phi(x_k))$ (DPG : deepest projected gradient)
  In general, it can be proved that the error goes as $O(k^{-2})$ for a general class.



- $\tilde{x}_{k+1} = x_k - D\, \nabla_x \phi(x_k)$ where D psd is a matrix gives us Newton method ($D = (HF)^{-1}$) that is faster generally, like $O(\rho^k)$, $0 < \rho < 1$, but it is space/time expensive
  store of H ↗ ↖ inversion

- Penalization : $\min_x \phi(x) + \frac{1}{2}\rho \|c(x)\|^2$, where the aim is not to stay exactly on the space $c(x)=0$, but find a close solution. When ρ is very large, the problem finds a solution that is really close to $\min \phi(x)$. If x_ρ is the solution for a certain ρ, one can prove that for $c(x)$, $\phi(x)$ cont. functions, every converg. subs. of x_ρ tends to a min of $\phi(x)$. The problem is that the problem is highly unstable for big ρ, because the algorithms tend to ignore $\phi(x)$ if it is $\ll \rho$.

with $\phi(x)$ differentiable $\phi : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^m$
Then we can introduce the Lagrangian function

$$\mathcal{L}(x,\lambda) = \phi(x) + \lambda^T \cdot c(x)$$

Here we can recall the common multiplier method, that shows that x^* is a strict local min. of (1) if (x^*, λ^*) solves

$$\begin{cases} \nabla_x \phi(x) + \lambda^T \cdot \nabla_x c(x) = 0 & = \nabla_x \mathcal{L}(x,\lambda) \\ c(x) = 0 & = \nabla_\lambda \mathcal{L}(x,\lambda) \\ z^T H_x \mathcal{L}(x,\lambda) \cdot z > 0 & \forall z : z^T \nabla_x c(x) = 0,\ z \neq 0 \end{cases}$$

Lemma 4.1 If $z^T H z > 0$ $\forall z : z^T J = 0$, then $\exists \rho \geq 0$ such that $H + \rho J J^T$ is psd

Proof $\theta(v) = -\frac{v^T H v}{v^T J J^T v}$ on $\|v\| = 1$ is a function that goes to $-\infty$ near the kernel of JJ^T and it is continuous otherwise, so it has a max equal to ρ, since $\rho \geq -\frac{v^T H v}{v^T J J^T v} \Rightarrow v^T H v + \rho v^T J J^T v \geq 0$ $\forall v$ ∎

If now we consider the augmented Lagrangian

$$\mathcal{L}_\rho(x,\lambda) = \phi(x) + \lambda^T c(x) + \frac{\rho}{2}\|c(x)\|^2$$

and (x^*, λ^*) the solution above, we find

$$\nabla_x \mathcal{L}_\rho(x^*,\lambda^*) = \frac{\rho}{2} \nabla_x \|c(x^*)\|^2 = \rho\, \overset{\overset{0}{\parallel}}{c(x^*)}^T \nabla_x c(x^*) = 0$$

$$H_x \mathcal{L}_\rho(x^*,\lambda^*) = H_x \mathcal{L}(x^*,\lambda^*) + \rho\, \nabla_x c(x^*)^T \nabla_x c(x^*) \succcurlyeq 0$$

$\Rightarrow$ If λ is close to λ^*, we just need to minimize $\mathcal{L}_\rho(x,\lambda)$ over x. To update λ, notice that we want $\nabla_x \mathcal{L}(x,\lambda) = 0$, so given x_ρ the minimum of $\mathcal{L}_\rho(\cdot,\lambda)$ we have

$$\begin{aligned} 0 = \nabla_x \mathcal{L}_\rho(x_\rho,\lambda) &= \nabla_x \phi(x_\rho) + \lambda^T \nabla_x c(x_\rho) + \rho\, c(x_\rho)^T \nabla_x c(x_\rho) \\ &= \nabla_x \phi(x_\rho) + (\lambda + \rho c(x_\rho))^T \nabla_x c(x_\rho) \\ &= \nabla_x \mathcal{L}(x_\rho, \lambda + \rho c(x_\rho)) \end{aligned}$$

⤳ the update of λ is $\lambda \leftsquigarrow \lambda + \rho c(x_\rho)$

The advantage is that there is no need for ρ to go to ∞, so we have a more stable method.

Theorem 4.2 Suppose (x^*, λ^*) satisfies the condition in $\mathcal{L}(x,\lambda)$ to be a local strict min, and let ρ be big enough as before. Then $\exists \delta, \varepsilon, M$ t.c.

$$\|\lambda_k - \lambda^*\| \leq \rho_k \cdot \delta, \quad \rho_k \geq \rho$$

$\Rightarrow$ $\min_x \mathcal{L}_{\rho_k}(x,\lambda_k)$ has a unique minimizer x_k with

$$\|x_k - x^*\| \leq M \|\lambda_k - \lambda^*\| / \rho_k$$

and moreover if $\lambda_{k+1} = \lambda_k + \rho_k c(x_k)$ then

$$\|\lambda_{k+1} - \lambda^*\| \leq M \|\lambda^k - \lambda^*\| / \rho_k \qquad [3, \text{Prop. } 4.2.3]$$

⤳ This shows a convergence $O(\varepsilon^n)$

PCP : Principal Component Pursuit

$$\min_{\mu,L,E} \|L\|_* + \lambda \|E\|_1^{\swarrow\text{vector norm 1}} \quad : \quad X = L+E$$

Let us write its augmented Lagrangian:

$$\mathcal{L}_\mu(L,E,\Lambda) = \|L\|_* + \lambda\|E\|_1 + \langle \Lambda, X-L-E\rangle + \frac{\mu}{2}\|X-L-E\|_F^2$$

ADMM Algorithm

Alternating Direction Method of Multipliers

```
Initialize $E_0 = \Lambda_0 = 0$, $\mu = \frac{nd}{4\|X\|_1}$, $\lambda = \frac{1}{\sqrt{n}}$   $(n \geq d)$
Repeat until convergence
    $L_{k+1} = \arg\min_L \mathcal{L}_\mu(L, E_k, \Lambda_k)$
    $E_{k+1} = \arg\min_E \mathcal{L}_\mu(L_{k+1}, E, \Lambda_k)$
    $\Lambda_{k+1} = \Lambda_k + \mu(X - L_{k+1} - E_{k+1})$
```

Notice that $\mathcal{L}_\mu$ is convex in L, E. We can compute the argmin for L, E explicitly through the subgradient. In fact

$$\arg\min_L \mathcal{L}_\mu(L,E,\Lambda) = \arg\min_L \|L\|_* - \langle \Lambda, L\rangle + \frac{\mu}{2}\|X-L-E\|_F^2$$

given $L = U\Sigma V^T$ the SVD, let $USV^T \in \partial_{sub}\|L\|_*$ and

$$\partial_{sub} = USV^T - \Lambda + \mu L + \mu(E-X) = 0$$

$$\Rightarrow U(S+\mu\Sigma)V^T = \Lambda + \mu(X-E)$$

so $\Lambda + \mu(X-E) = U\tilde{\Sigma}V^T$ gives us U, V and $S + \mu\Sigma = \tilde{\Sigma}$, so

$$\omega_i = \begin{cases} 0 & \tilde{\omega}_i \leq 1 \\ \frac{\tilde{\omega}_i - 1}{\mu} & \tilde{\omega}_i > 1 \end{cases} \rightsquigarrow \omega_i = 1/\mu \max\{0, \tilde{\omega}_i - 1\}$$

"soft Thresholding"

$$\rightsquigarrow \underline{\arg\min_L \mathcal{L}_\mu(L,E,\Lambda) = D_{1/\mu}\left(1/\mu\,\Lambda + X - E\right)}$$

$$\arg\min_E \mathcal{L}_\mu(L,E,\Lambda) = \arg\min_E \lambda\|E\|_1 - \langle \Lambda, E\rangle + \frac{\mu}{2}\|X-L-E\|_F^2$$

$$\partial_{sub} = \lambda\,\mathrm{sgn}(E) + \lambda W - \Lambda + \mu E + \mu(L-X) = 0$$

$$\Rightarrow \lambda\,\mathrm{sgn}(E) + \mu E + \lambda W = \Lambda + \mu(X-L) = A$$

$$\Rightarrow E_{i,j} = \begin{cases} 0 & |A_{ij}| \leq \lambda \\ A_{ij} - \lambda/\mu & A_{ij} > \lambda \\ A_{ij} + \lambda/\mu & A_{ij} < -\lambda \end{cases} = \mathrm{sgn}(A_{ij}) \max\left\{0, \frac{|A_{ij}| - \lambda}{\mu}\right\}$$

$$\rightsquigarrow \underline{\arg\min_E \mathcal{L}_\mu(L,E,\Lambda) = S_{\lambda/\mu}\left(1/\mu\,\Lambda + X - L\right)}$$

ADMM Algorithm

```
Initialize $E_0 = \Lambda_0 = 0$, $\mu = \frac{nd}{4\|X\|_1}$, $\lambda = 1/\sqrt{m}$   $(m \geq d)$
Repeat until convergence
    $L_{k+1} = D_{1/\mu}(1/\mu\,\Lambda_k + X - E_k)$
    $E_{k+1} = S_{\lambda/\mu}(1/\mu\,\Lambda_k + X - L_{k+1})$
    $\Lambda_{k+1} = \Lambda_k + \mu(X - L_{k+1} - E_{k+1})$
```

Notice that the rk of L is not an input of the system.

Robust PCA to Outliers

When some of the data $\{x_i\}$ may be corrupted or badly sampled we say that we are in presence of Outliers.

A way to deal with it is with the weight matrix W as in the previous case, where now $W = e w^T$, i.e. there's a weight associated with every x_i and they are computed as

$$w_i = \rho(\varepsilon_i)/\varepsilon_i^2 \quad , \quad \varepsilon_i = \| x_i - \mu - U y_i \|$$

A similar algorithm can be adopted for dealing with outliers:

Iteratively Reweighted Least Squares with Outliers

```
Initialize $\mu, U, Y$ as classic PCA of $X$, $\varepsilon_0 > 0$
Repeat until convergence
    $E = X - \mu e^T - UY$, $\quad W_{ij} = \frac{\varepsilon_0^2}{\|E_j\|^2 + \varepsilon_0^2} \quad \forall i,j$
    $(\mu, U, Y) = $ Power_Iteration_Step $(\mu, U, Y, W)$

Return $\mu + \frac{1}{n} UYe$, $U$, $Y(I - ee^T/n)$, $E = X - \mu e^T - UY$
```

Another way is to solve the problem

$$\min_{L,E} \operatorname{rank}(L) + \lambda \|E\|_{2,0} \quad : X = L + E$$

where the idea is to find a low rank L and a column sparse E, in fact $\|E\|_{2,0}$ is the number of non zero columns, or equivalently the 0-norm of $(\|E_1\|, \dots, \|E_m\|)$. As usual, this is NP-hard, so we relax to the convex hulls

$$\min_{L,E} \|L\|_* + \lambda \|E\|_{2,1} \quad ; \quad X = L + E$$

where $\|E\|_{2,1} = \sum_{i=1}^{n} \|E_i\|$. This is called Outlier Pursuit Program.

See Th. 7.1 for when the OPP gives the correct underlying $X = L + E$. See ADMM for a way to solve this problem.

$\rightsquigarrow$ Same warning for ADMM and non-diff. holds here.

Incoherence wrt Column Sparse Matrices

A rank k matrix $L \in \mathbb{R}^{d \times n}$ with reduced SVD $L = U \Sigma V^T$ and $(1-\gamma)n$ nonzero columns is said ν-incoherent wrt... if

$$\max_j \|v_j\|^2 \leq \frac{\nu k}{(1-\gamma) n} \qquad v_j \text{ rows of } V$$

Theorem 7.1 Let $X = L_0 + E_0 \in \mathbb{R}^{d \times n}$ with L_0 ν-incoherent wrt.. and at least $(1-\gamma)n$ columns non zero, and with E_0 supported on at least γn columns. If

$$\operatorname{rk}(L_0) \leq \left(\frac{3}{n}\right)^2 \frac{1-\gamma}{\nu \cdot \gamma}$$

Then the solution of the Outlier Pursuit Program with $\lambda = \frac{3}{7\sqrt{\gamma n}}$

$$\min_{L,E} \|L\|_* + \lambda \|E\|_{2,1} \quad : \quad X = L + E$$

identifies the column space of L_0 and the outlier index in E_0. [8]

In case of Outliers, the OPP model is

$$\min_{L,E,\Lambda} \mathcal{L}_\mu(L,E,\Lambda) = \min_{L,E,\Lambda} \|L\|_* + \lambda \|E\|_{2,1} + \langle \Lambda, X - L - E \rangle + \frac{\mu}{2} \|X - L - E\|_F^2$$

so we can apply an ADMM to minimize it. The updates of L and Λ are the same as above. The only difference is the update of E

$$\operatorname*{argmin}_E \lambda \|E\|_{2,1} - \langle \Lambda, E \rangle + \frac{\mu}{2} \|X - L - E\|_F^2$$

$$\underset{\partial_{sub}}{\rightsquigarrow} \lambda B - \Lambda + \mu E + \mu(L - X) = 0 \quad , \quad \begin{matrix} E_j = 0 \Rightarrow \|B_j\| \leq 1 \\ E_j \neq 0 \Rightarrow B_j = E_j / \|E_j\| \end{matrix}$$

$$\frac{\lambda}{\mu} B + E = \frac{1}{\mu} \Lambda + X - L = A$$

so

$$\left. \begin{matrix} \|A_j\| > \lambda/\mu \Rightarrow E_j = \frac{A_j}{\|A_j\|} (\|A_j\| - \lambda/\mu) \\ \|A_j\| \leq \lambda/\mu \Rightarrow E_j = 0 \end{matrix} \right\} E = T_{\lambda/\mu}(A)$$

ADMM Algorithm for outliers

```
Initialize $E_0 = \Lambda_0 = 0$, $\mu = \frac{nd}{4\|X\|_{2,1}}$
Repeat until convergence
    $L_{k+1} = D_{1/\mu}(\frac{1}{\mu}\Lambda_k + X - E_k)$
    $E_{k+1} = T_{\lambda/\mu}(\frac{1}{\mu}\Lambda_k + X - L_{k+1})$
    $\Lambda_{k+1} = \Lambda_k + \mu(X - L_{k+1} - E_{k+1})$
```

Notice that once we have L, one can determine if a new data y is an outlier: the idea is that an inlier is $y \sim L \cdot v$ with $\|v\| \leq 1$. Given $L = U \Sigma V$ with $U \Sigma$ square, then a big $\|\Sigma^{-1} U^T y\|$ $\gg 1$ is an indicator of outliers.

$\rightsquigarrow$ This is akin to a training of a NN