ENGSCI 721

INVERSE PROBLEMS

Oliver Maclaren
oliver.maclaren@auckland.ac.nz

MODULE OVERVIEW

Inverse Problems (*Oliver Maclaren*) [~8 lectures/2-3 tutorials]

1. *Basic concepts* [3 lectures]
   Forward vs inverse problems. Well-posed vs ill-posed problems. Algebra of inverse problems (generalised inverses etc). Regularisation and trade-offs.
2. *More regularisation* [3 lectures]
   Higher-order Tikhonov regularisation, truncated singular value decompositions, iterative regularisation.

MODULE OVERVIEW

3. *Statistical view of inverse problems I* [2 lectures]
   Bayesians, Frequentists and all that. Basic frequentist analysis. Linearisation and covariance propagation.

LECTURE 5: REGULARISATION IN LINEAR PROBLEMS: SVD AND TSVD

Topics:

- Singular Value Decomposition
  - The ‘crown jewel’ of linear algebra!
  - Generalises eigenvalue analysis to general (non-square etc) matrices
- Truncated Singular Value Decomposition
  - As regularisation scheme
  - Relation to Tikhonov regularisation

EngSci 721 : Lecture 5.

Regularisation in linear problems :
The Singular Value Decomposition perspective

- SVD (type of matrix factorisation)
  - L extension of eigen analysis
  - L insight/calculation for inverses, resolution, effect of regulars.
  - L the 'crown jewel' of linear algebra
- Truncated SVD.
  - L as regularisation scheme.
  - L connection to Tikhonov.

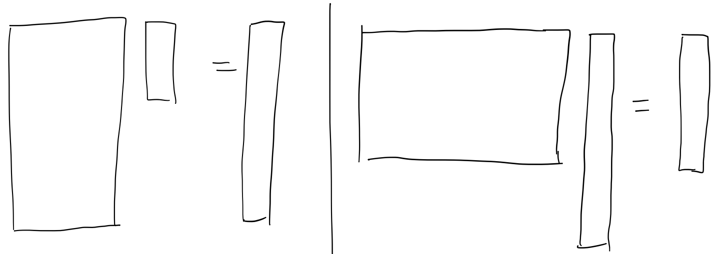
<u>Bonus</u> : rank factorisation & extension to nonlinear epi-mono factorisations

Eigenvalues

Recall that for a square matrix A, eigenvalues solve

$$\boxed{Ax = \lambda x}$$

However, we are interested in <u>non-square</u> matrices in inverse problems (& statistics etc)!



Tall/overdetermined systems
- 'classical statistics'

Wide/underdetermined systems
- 'inverse problems'
- 'nonparametric statistics
- 'machine learning' etc

→ eigenvalues don't make sense for non-square!

Eigenvalues?

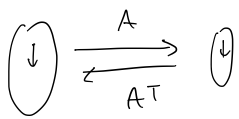
A	$m \times n$	
x	n	$Ax = y$
y	m	

$\underbrace{Ax}_{\in \mathbb{R}^m} = \underbrace{\lambda x}_{\in \mathbb{R}^n}$ doesn't make sense

} live in different spaces!

Solutions?

related {
- Consider different bases for each space
- Consider eigenvalues of square matrices like A^TA & AA^T:



A^TA: $\mathbb{R}^n \to \mathbb{R}^n$

AA^T: $\mathbb{R}^m \to \mathbb{R}^m$

Singular values & singular vectors I.

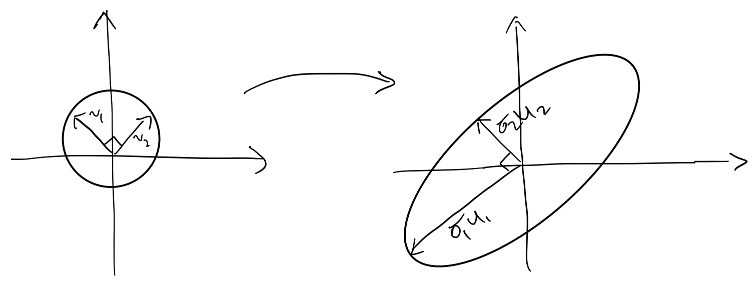
Instead of $Av = \lambda v$ } eigenvector v, eigenvalue λ

we consider $\boxed{Av = u\sigma}$

matrix, vector, vector, scalar

Solutions: $\{(u_i, \sigma_i, v_i)\}$

- left singular vector (data space)
- singular value
- right singular vector (model space)



model space $\mathbb{R}^n$
x

data space $\mathbb{R}^m$
y

Singular values & singular vectors II

In particular, for singular vectors/values

$$\boxed{Av_i = u_i \sigma_i}$$

we require $\{v_i\}$ & $\{u_i\}$ to both be

$\boxed{\text{orthonormal}}$ $\left\{ \begin{array}{l} \text{orthogonal} \\ \text{unit length} \end{array} \right\} \Rightarrow$ LI

sets of vectors.

eg $u_i^T u_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$ etc

& that they $\boxed{\text{span}}$ their respective spaces

while $\boxed{\sigma_i \geq 0}$ are non-negative.

In contrast to eigenvalues/vectors, we can always work with singular values/vectors

→ generalisation to nonsquare matrices

Singular value decomposition (SVD) I.

Suppose for the moment all rows are independent & consider all $\boxed{Av_i = u_i \sigma_i}$ solns ($\sigma_i \geq 0$):

$$\begin{array}{c} \\ m \end{array}\!\! \overset{n}{\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix}} \overset{n}{\begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{bmatrix}} = \overset{m}{\begin{bmatrix} | & | & & | \\ u_1 & u_2 & \cdots & u_m \\ | & | & & | \end{bmatrix}} \begin{bmatrix} \sigma_1 & & & 0 & \cdots & 0 \\ & \ddots & & \vdots & \ddots & \vdots \\ & & \sigma_m & 0 & \cdots & 0 \end{bmatrix}$$

$$m \times n \qquad n \times n \qquad m \times m \qquad m \times n$$

where $\{v_i\}$ & $\{u_i\}$ are orthonormal sets

ie $\boxed{AV = U\Sigma}$, V & U are orthogonal matrices, V $n \times n$, U $m \times m$

Note: n cols of A not LI, but n v vectors are (why?)

$\Rightarrow$ V & U are invertible, with inverses $V^{-1} = V^T$, $U^{-1} = U^T$

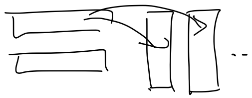
→

Side note: matrix multiplication

$\boxed{AB \text{ can be thought of in } \underline{\text{multiple}} \text{ ways}}$

here:
$A_{m\times n}$, $B_{n\times p}$
$A[:, i]$ = ith col of A
$A[i, :]$ = ith row of A
etc.

1. → usual $\boxed{\text{'rows of A times cols of B'}}$ (inner product entries)



$$\cdots = \begin{bmatrix} A_{[1,:]}B_{[:,1]} & \cdots & A_{[1,:]}B_{[:,p]} \\ \vdots & & \\ A_{[m,:]}B_{[:,1]} & \cdots & A_{[m,:]}B_{[:,p]} \end{bmatrix}$$

2. → generalised version of linear combo of A's cols from matrix times vector rule:

$$\begin{bmatrix} \boxed{1} & \boxed{2} & \cdots & \boxed{n} \end{bmatrix} \begin{bmatrix} \boxed{\quad 1 \quad} \\ \boxed{\quad 2 \quad} \\ \vdots \\ \boxed{\quad n \quad} \end{bmatrix}$$

$$\left(\text{cf: } \begin{bmatrix} \boxed{1} & \boxed{2} & \cdots \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ n \end{bmatrix} \right)$$

$$= \boxed{1}\,\boxed{\quad 1 \quad} + \boxed{2}\,\boxed{\quad 2 \quad} + \cdots$$

ie $\boxed{\text{'sum of outer products'}}$:

$$\boxed{AB = \sum_i A_{[:,i]}B_{[i,:]}} \quad \left(= \sum a_i b_i^T \right.$$

Note: common notation

$b_1^T = B_{[1,:]}$

$$\left. \text{if } A = \begin{bmatrix} a_1 \,|\, a_2 \cdots \end{bmatrix} \;\&\; B = \begin{bmatrix} b_1^T \\ b_2^T \\ \vdots \end{bmatrix} \right)$$

3. → matrix A times each col of B
ie $\boxed{\text{'multiple RHS'}}$ form

$$A \begin{bmatrix} \boxed{1} & \boxed{2} & \cdots & \boxed{\;} \end{bmatrix} = \begin{bmatrix} A\boxed{1} & A\boxed{2} & \cdots & A\boxed{p} \end{bmatrix}$$

$$= \begin{bmatrix} AB_{[:,1]} \;\vdots\; AB_{[:,2]} \;\vdots\; \cdots \;\vdots\; AB_{[:,p]} \end{bmatrix}$$

Exercise:

verify $Av_i = u_i\sigma_i$ can be written $AV = U\Sigma$ as given on prev. page, when rank $= m$, $n > m$, by considering

$$\begin{array}{cc} & n \\ m & \begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} \end{array} \begin{array}{cc} n & \\ \begin{bmatrix} \vdots & \vdots & & \\ v_1 & v_2 & \cdots & v_n \\ \vdots & \vdots & & \end{bmatrix} & n \end{array}$$

$$\begin{array}{cc} & m \\ m & \begin{bmatrix} \vdots & & \vdots \\ u_1 & \cdots & u_m \\ \vdots & & \vdots \end{bmatrix} \end{array} \begin{array}{cc} & \begin{array}{cc} m & n-m \end{array} \\ m & \left[\begin{array}{ccc|c} \sigma_1 & & & \\ & \ddots & & 0 \\ & & \sigma_m & \end{array}\right] \end{array}$$

[Note: there may be multiple $\sigma = 0$ sol^{n}s!]

Singular value decomposition (SVD) II.

The SVD is then given by:

$$\boxed{A = U \Sigma V^T} \qquad (V^{-1} = V^T)$$

Every matrix has an SVD

→ if A is $m \times n$ with rank r

then U & V still $m \times m$ & $n \times n$ (span $\mathbb{R}^m$ & $\mathbb{R}^n$), while

$$\Sigma = \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix}$$ block matrix shape $m \times n$

& Σ_r is $r \times r$ diagonal matrix with positive entries, & ordered as $\sigma_1 \geq \sigma_2 \geq \dots \sigma_r \geq 0$

Compact/reduced form: (rank r)

$$\boxed{A = U_r \Sigma_r V_r^T}$$

U_r: first r col of U

V_r: first r col of V

ie $A = [U_r, U_0] \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} [V_r, V_0]^T$

Some Properties of SVD

- if a matrix has rank r then it has r non-zero singular values $\Big\}$ (as hinted prev. pages)
- $A = U \Sigma V^T = U_r \Sigma_r V_r^T$
- $A^T = V \Sigma^T U^T = V_r \Sigma_r U_r^T$ $\Big\}$ $A^T U = V \Sigma^T$; U basis for data, V for model space.
- $$\begin{aligned} A^T A &= V \Sigma^T U^T U \Sigma V^T \\ &= V \Sigma^2 V^T \\ &= V_r \Sigma_r^2 V_r^T \end{aligned}$$
- $$\begin{aligned} A A^T &= U \Sigma V^T V \Sigma^T U^T \\ &= U \Sigma \Sigma^T U^T \\ &= U_r \Sigma_r^2 U_r^T \end{aligned}$$

Eigen for model/data

$$\mathbb{R}^n \underset{A^T}{\overset{A}{\rightleftarrows}} \mathbb{R}^m$$

$A^T A$ $n \times n$	$A A^T$
V $n \times n$ basis v_i	U $m \times m$ basis u_i

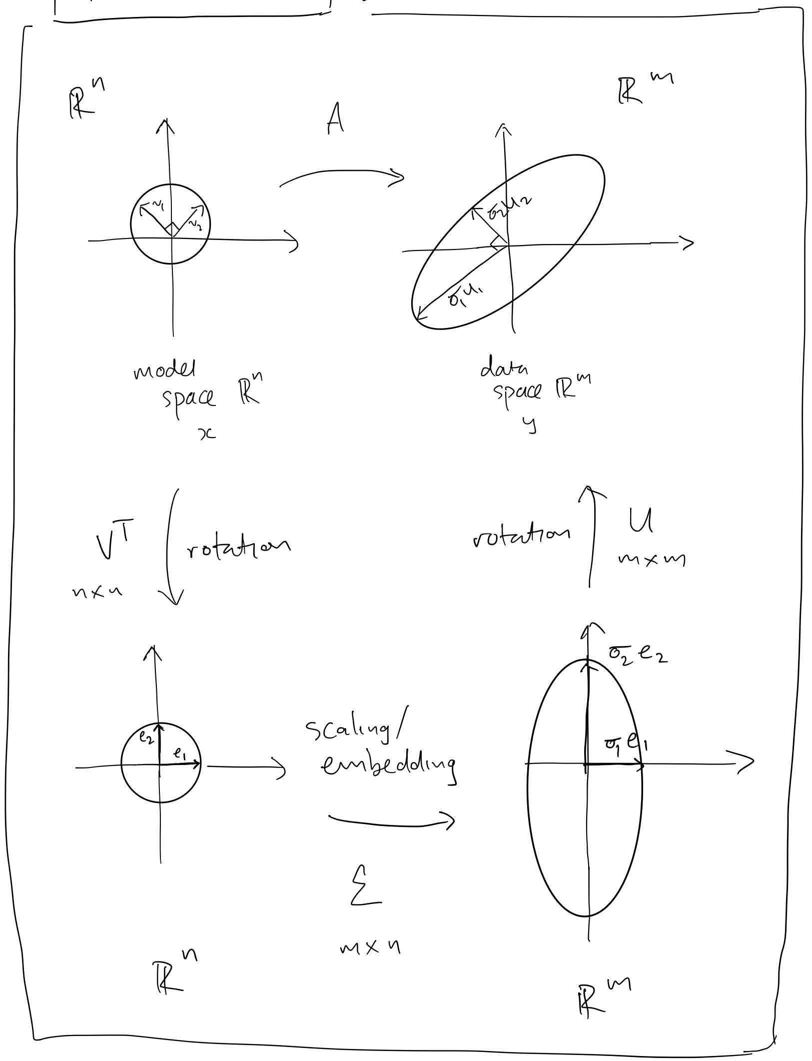
→ σ_i^2 are the non-zero eigenvalues of $A^T A$ & of $A A^T$

→ associated (non-zero σ_i) u_i & v_i are eigenvectors of $A A^T$ & $A^T A$ respectively

---

SVD : Interpretation

$\boxed{A = U \Sigma V^T}$:



SVD : Big picture

Key advantage : $\boxed{\text{explict calculation}}$

- inverses (left, right, pseudo)
- model / data resolution operators
- stability / instability depending on singular values
- stablised approximations via truncation
- effect of Tikhonov (etc) regularisation on singular values

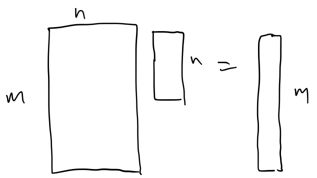
Disadvantage : though some intutions transfer to nonlinear, essentially a linear concept.
(But see rank factorisation)

SVD & Inverses (Left / Retraction)

Recall:

A left inverse satisfies $\boxed{LA = I}$ $\left.\right\}$ A $m \times n$, L $n \times m$, I $n \times n$

& a left inverse exists when # rows $\geq$ # cols of A & the cols are LI



U $m \times m$

V $n \times n$

Given $\boxed{A = U_n \Sigma_n V_n^T}$, $\boxed{\text{rank} = n}$

Consider:

$$\boxed{L = \underbrace{\underbrace{V_n}_{n \times n} \underbrace{\Sigma_n^{-1}}_{n \times n} \underbrace{U_n^T}_{n \times m}}_{n \times m}}, \quad \Sigma_n^{-1} = \begin{bmatrix} \frac{1}{\sigma_1} & & \\ & \ddots & \\ & & \frac{1}{\sigma_n} \end{bmatrix}$$

$$LA = V_n \Sigma_n^{-1} U_n^T U_n \Sigma_n V_n^T$$

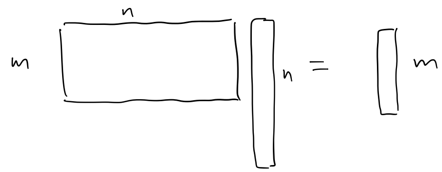
$$= I \qquad (n \times n)$$

SVD & Inverses (Right / Section)

Recall:

A right inverse satisfies $\boxed{AR = I}$ $\left.\right\}$ A $m \times n$, R $n \times m$, I $m \times m$

& a right inverse exists when # cols $\geq$ # rows of A & the rows are LI



U $m \times m$

V $n \times n$

Given $\boxed{A = U_m \Sigma_m V_m^T}$, $\boxed{\text{rank} = m}$

Consider:

$$\boxed{R = \underbrace{\underbrace{V_m}_{n \times m} \underbrace{\Sigma_m^{-1}}_{m \times m} \underbrace{U_m^T}_{m \times m}}_{n \times m}}, \quad \Sigma_m^{-1} = \begin{bmatrix} \frac{1}{\sigma_1} & & \\ & \ddots & \\ & & \frac{1}{\sigma_m} \end{bmatrix}$$

$$AR = U_m \Sigma_m V_m^T V_m \Sigma_m^{-1} U_m^T$$

$$= I \qquad (m \times m)$$

SVD & The Generalised (Pseudo) inverse

In general, given

$\boxed{A = U_r \Sigma_r V_r^T}$, $\boxed{\text{rank } r}$

we have the generalised (pseudo) inverse:

$\boxed{A^\dagger = V_r \Sigma_r^{-1} U_r^T}$

explicit formula
... & ...
recall V, U, σ related to A^TA & AA^T eigen.

→ The generalised inverse is usually computed via SVD

→ We have seen that the generalised inverse needs regularisation

↳ New idea: truncate SVD for $p < r$

But first ..... recall

Model resolution, data resolution operators:

$\boxed{R_D = A A^\dagger}$ } how much data is 'shrunk' or smeared

$\boxed{R_M = A^\dagger A}$ } how much model is 'shrunk' or smeared

see below

Not I in gen. but something 'similar'
→ Note $I^2 = I$ ('idempotent')

Projection operators P characterised by

$\boxed{P^2 = P}$ ('idempotent')

→ one application of P gives 'maximum' effect

1. Suppose $A^\dagger A = I$ but $AA^\dagger \neq I$ (left inverse only)

$\Rightarrow R_D R_D = A A^\dagger A A^\dagger = A A^\dagger = R_D$

$\Rightarrow R_D$ is a projection on data space

2. Suppose $AA^\dagger = I$ but $A^\dagger A \neq I$ (right inverse only)

$R_M R_M = A^\dagger A A^\dagger A = A^\dagger A = R_M$

$\Rightarrow R_M$ is a projection on model space.

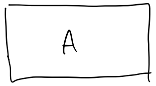
SVD & Resolution: Explicit calculation.

Now: $\boxed{R_D = U_r U_r^T}$ $m \times m \begin{cases} U_r & m \times r \\ U_r^T & r \times m \end{cases}$ (r $m$vectors)

$\boxed{R_M = V_r V_r^T}$ $n \times n \begin{cases} V_r & n \times r \\ V_r^T & r \times n \end{cases}$ (r $n$vectors)

- If rank $r = m < n$

  $\Rightarrow R_D = I_m$



(recover data exactly)

(models are 'reduced')

  But $R_M \neq I_n$

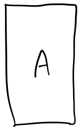
  Though $R_M^2 = V_r \underline{V_r^T V_r} V_r^T = V_r V_r^T = R_M$

  $\Rightarrow$ R_M is model projection operator

- If rank $r = n < m$

  $R_D \neq I_m$

  $R_M = I_n$



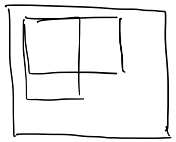
(data are 'reduced')

(models recovered exactly.)

  though $R_D = U_r \underline{U_r^T U_r} U_r^T = U_r U_r^T = R_D$

  $\Rightarrow$ R_D is data projection

- If rank $r < m$ & n



  $R_D \neq I_m$

  $R_M \neq I_n$

  $\Rightarrow$ Both are projection operators.

  ($U_r^T U_r = I$, $V_r^T V_r = I$ still)

Exercise (Tut/Assignment):

Explore model/data resolution operators for typical inverse problem examples seen so far

---

→

So.... regularisation!

→ singular values may be positive but effectively zero (machine tol. etc)

⇒ cause: $\boxed{\text{effective rank } p < \text{rank } r}$

→ small singular values cause instability

Key: inverse leads to

$\boxed{\text{dividing by small } \sigma_i \text{ values}}$

SVD as basis expansion:

$$A^\dagger = V_r \Sigma_r^{-1} U_r^T = \sum_i^r \left(v_i \frac{1}{\sigma_i} u_i^T \right)$$

& $x^\dagger = A^\dagger y$

$$= \sum_i^r \left(v_i \frac{1}{\sigma_i} u_i^T y \right)$$

$$= \mathrm{Sum}\left[\underbrace{\left(\frac{u_i^T y}{\sigma_i} \right)}_{\text{coeff.}} \underbrace{v_i}_{\text{basis vector in model space}} \right]$$

Large for $\sigma_i \to 0$

Stability

consider $x^\dagger = A^\dagger y$

& $x^{\dagger\prime} = A^\dagger y'$

for small data perturbation

$$\| y - y' \|_2 < \delta$$

Then $x^\dagger - x^{\dagger\prime} = A^\dagger (y - y')$

& $\| x^\dagger - x^{\dagger\prime} \|_2 \leq \| A^\dagger \|_2 \| y - y' \|_2$

$$\boxed{\text{where } \| A \|_2 := \max_{\|x\|_2 = 1} \| A x \|_2 = \sigma_1 = \text{largest singular value}}$$

Leads to (with other details...)

$$\boxed{\frac{\| x^\dagger - x^{\dagger\prime} \|_2}{\| x^\dagger \|_2} \leq \frac{\sigma_1}{\sigma_r} \frac{\| y - y' \|_2}{\| y \|_2}}$$

σ_1: largest singular value

σ_r: smallest singular value.

Stability : Key point

Stability (continuity modulus) of A^+ governed by

$$\boxed{\text{cond}(A^+) = \frac{\sigma_1}{\sigma_r}}$$

(condition number)

Key trade-off :

truncate singular value expansion

↳ more stable (less 'variance')

↳ biased (model resolution less like identity

→ favour particular models)

(stats : Bias - Variance tradeoff)

Spectrum

Plot of singular values in decreasing order :

Key : $\boxed{\text{Ill-posed}}$

→ no clear gap

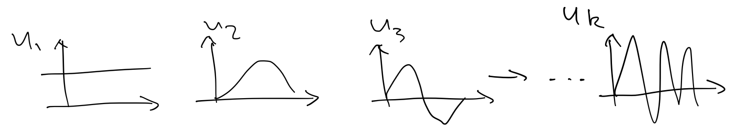
→ decreases to zero

} $\boxed{\text{rank is hard to define}}$

[cf : rank deficient : clear gap. A^+ OK then ?]

Also : singular vectors 'oscillate' more (sign changes in elements) for smaller values

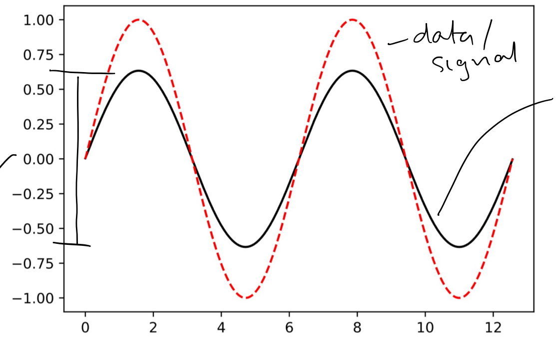
↳ Like Fourier bases (see ex.)



Example

Return to deconvolution example from L1.

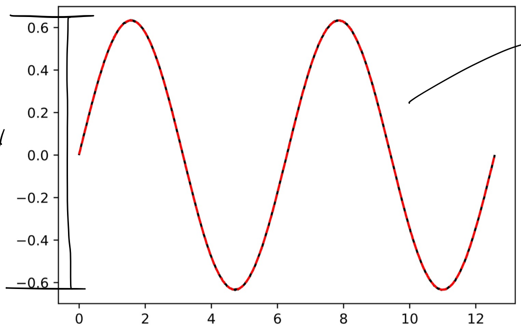
(convolution $\simeq$ window averaging
deconvolution $\simeq$ ..undoing ↑ !)



- data / signal

smoothed ('convolved') signal

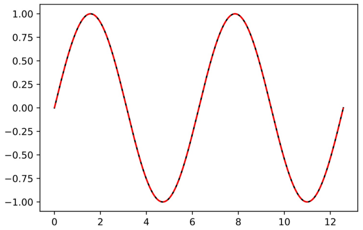
$Ax + \epsilon$



smoothed signal + tiny noise

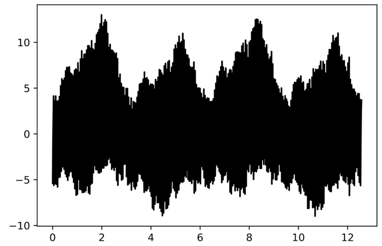
Example

Deconvolution no noise



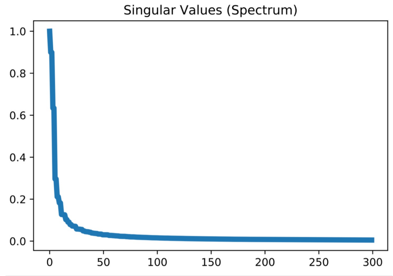
yay!

Deconvolution with noise

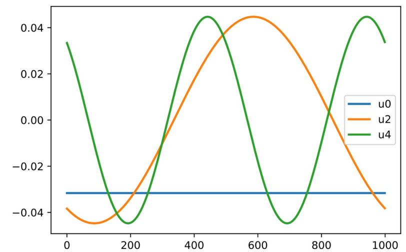


noo!

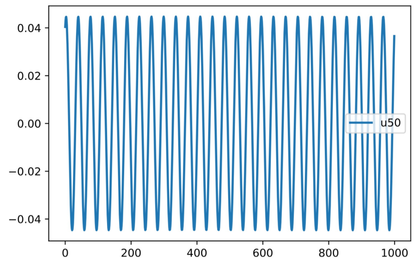
SVD: spectrum



u vectors (v similarly)



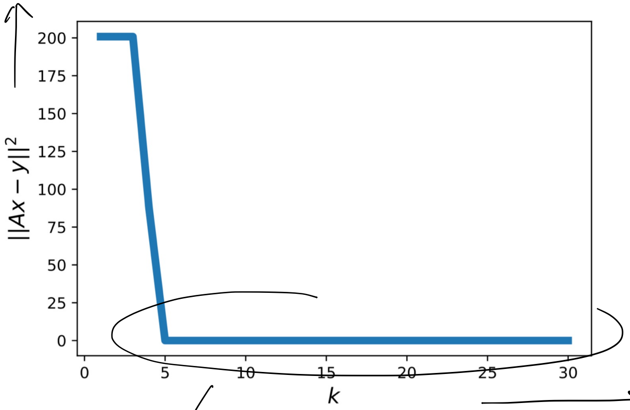
u vector for small σ_i:



Think: Fourier components.

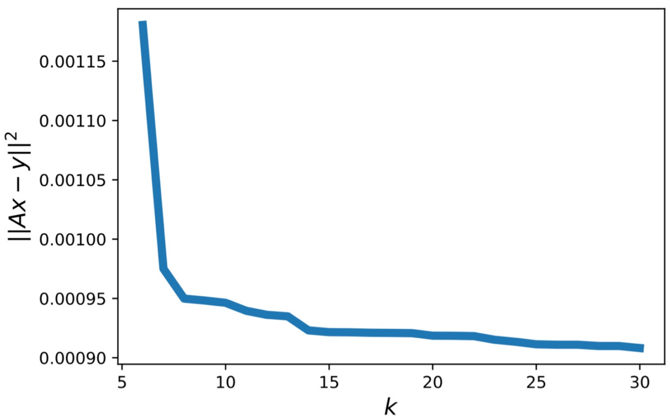
Pareto (trade-off) curve:

Fit



(number of retained singular values)

zoom

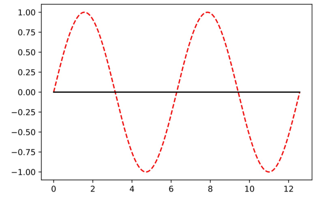


Solutions as depending on k (number retained singular values)

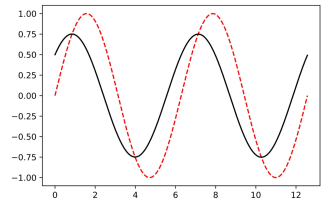
— recovered

- - - true

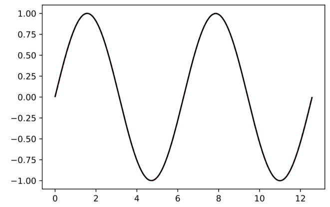
Stable/ Biased



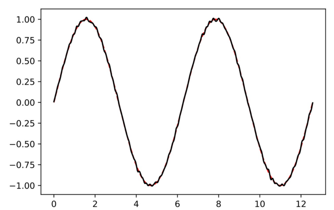
$k=1$



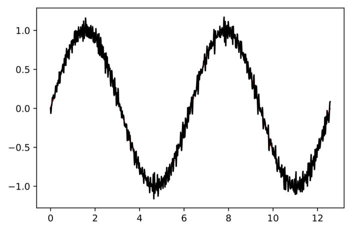
$k=4$



$k=5$ ← sweet spot



$k=100$



$k=300$

Unstable/ Unbiased

(this is a very smooth problem -- typically much worse, faster)

Choosing truncation ?

Pareto :

Smallest number of singular values giving adequate fit, beyond which 'flattens'

Picard condition

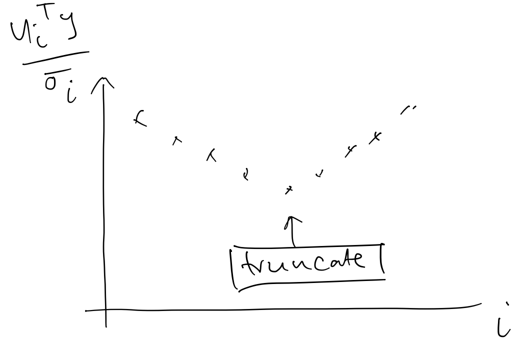
Consider

$$x = \mathrm{Sum}\left[\underbrace{\left(\frac{u_i^T y}{\sigma_i}\right)}_{\text{coeff.}} \underbrace{v_i}_{\text{model basis}}\right]$$

Plot $\frac{u_i^T y}{\sigma_i}$ vs i [exercise : do for convolution!]

Expect : $u_i^T y$ decay faster initially, then start to increase

→ truncate here



Note : relative decay is what matters, not abs. magnitude

Tikhonov & SVD

Finally, lets return to Tikhonov regularisation & see if SVD can help understand.

Zeroth order : Normal eq^{n}s

instead of λ to simplify

$$\boxed{(A^T A + \alpha^2 I)x = A^T y}$$

where now

$$A = U \Sigma V^T = U_r \Sigma_r V_r^T$$

$$A^T = V \Sigma^T U^T = V_r \Sigma_r U_r^T$$

$$A^T A = V \Sigma^T U^T U \Sigma V^T$$
$$= V \Sigma^2 V^T$$
$$= V_r \Sigma_r^2 V_r^T$$

→

... Tikhonov & SVD ...

can show

$$x = \text{sum}\left[\underbrace{\left(\frac{u_i^T y}{\sigma_i}\right)}_{\text{coeff.}} \underbrace{v_i}_{\text{model basis}}\right]$$

becomes

$$\boxed{x_\alpha = \sum_{i}^{r}\left[\left(f_i \cdot \frac{u_i^T y}{\sigma_i}\right) v_i\right]}$$

where

$$\boxed{f_i = \frac{\sigma_i^2}{\sigma_i^2 + \alpha^2}}$$

are the $\boxed{\text{filter factors}}$

Note:

$$\left\{\begin{array}{l} \alpha = 0 \Rightarrow f_i = 1 \\ \sigma_i << \alpha \Rightarrow f_i \to \left(\frac{\sigma_i}{\alpha}\right)^2 \to 0 \\ \sigma_i >> \alpha \Rightarrow f_i \to 1 \end{array}\right\}$$

⇒ $\boxed{\text{Tikhonov regularisation implements (continuous version of) truncated SVD!}}$

Bonus (not examinable)

Rank factorisation

→ see Piziak & Odell (1999) on Canvas

Given $\boxed{m \times n}$ matrix A with $\boxed{\text{rank } r > 0}$

⇒ can write $\boxed{A = FG}$

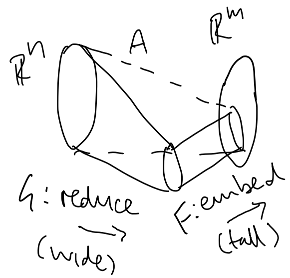
where $\boxed{\begin{array}{ll} G & r \times n \\ F & m \times r \end{array}}$

so:

$$A \sim \begin{array}{c|c} & r \\ \hline m & \begin{bmatrix} \text{tall} \end{bmatrix} \end{array} \begin{array}{c|c} & n \\ \hline r & \begin{bmatrix} \text{wide} \end{bmatrix} \end{array}$$

} note similarity to reduced SVD

ie tall × wide: $\boxed{\text{apply wide then tall}}$



} idea generalises to $\boxed{\text{nonlinear}}$

$\boxed{\text{epi-mono factorisation}}$ ie $\boxed{\text{onto-1-1 factorisation}}$

$$* \boxed{f = m \circ e} *$$

$\underset{\substack{\text{'tall'} \\ (1\text{-}1)}}{m}$, $\underset{\substack{\text{'wide'} \\ (\text{onto})}}{e}$ } nonlinear versions