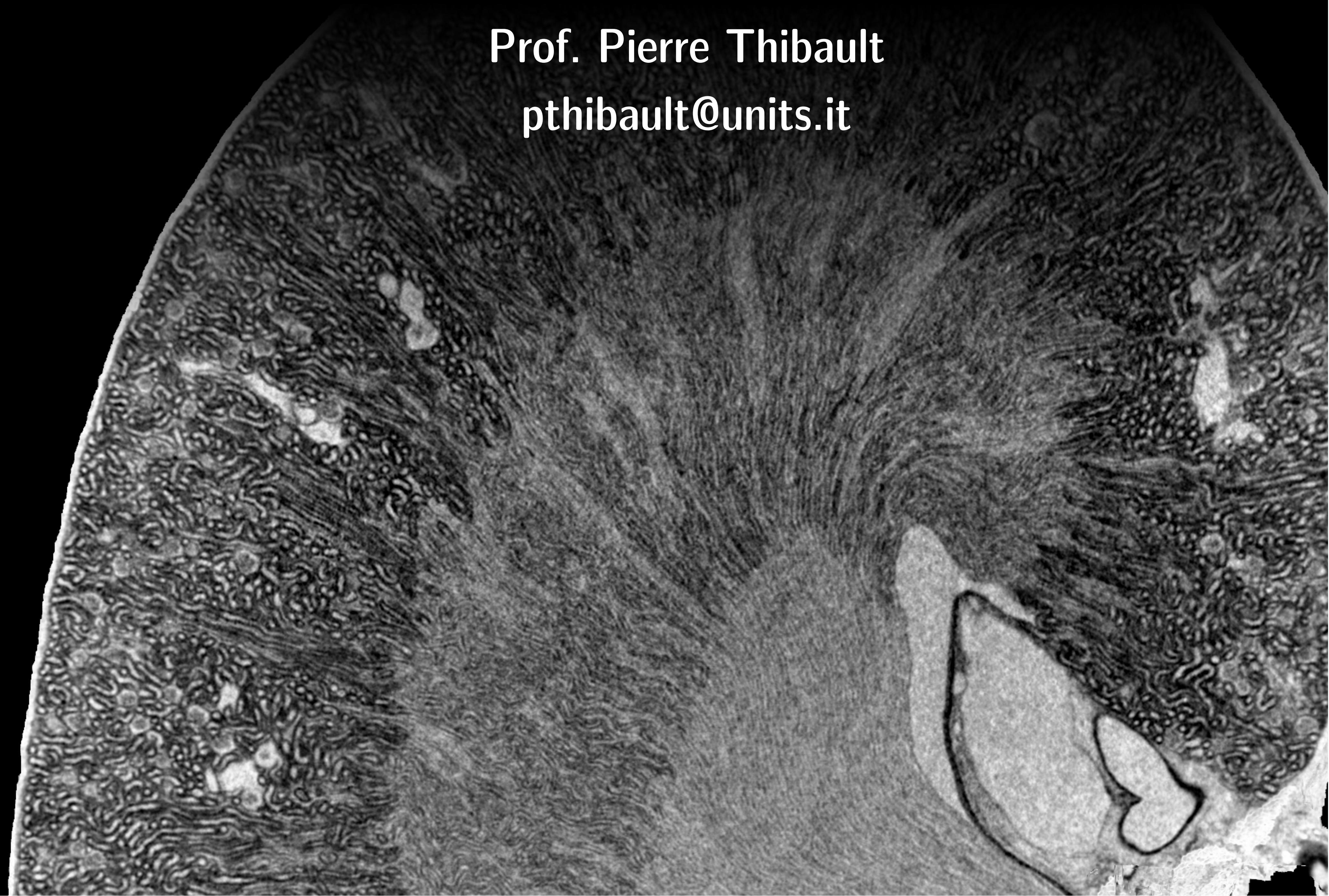
Image Processing for Physicists

Prof. Pierre Thibault

pthibault@units.it



Overview

- General remarks on optimization
- Least squares principle
  - Application examples
- Lagrange multipliers
  - Application examples

Image Processing Problems

- Image processing problems can be formulated as linear/nonlinear equations

  * data: y * model parameters β

  * variables: x (independent variables e.g. pixel coordinate)

- In many cases "true" solution does not exist (random noise!) or is hard to calculate

  model: $y = m(x; \beta)$

- Find "best-guess" approximation

  $\hat{\beta}$ "estimate" of β

- Need understanding of "approximation"

- Need understanding of "best" approximation

Estimation

- Estimator and Estimate

estimate $\hat{\beta}$

estimator : $t : \{y\} \longrightarrow \hat{\beta}$

- Cost function $f(y; x, \beta)$
  - Measures how well our estimate compares to the original

$$\min_{\beta} f = \hat{\beta}$$

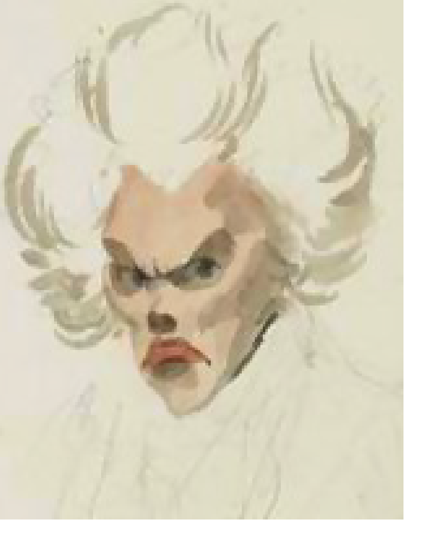
$\rightarrow$ Find Minima of cost function

$\rightarrow$ Optimization theory

Least squares principle

1. Introduction. The method of least squares is the automobile of modern statistical analysis: despite its limitations, occasional accidents, and incidental pollution, it and its numerous variations, extensions, and related conveyances carry the bulk of statistical analyses, and are known and valued by nearly all. But there has been some dispute, historically, as to who was the Henry Ford of statistics. Adrien Marie Legendre published the method in 1805, an American, Robert Adrain, published the method in late 1808 or early 1809, and Carl Friedrich Gauss published the method in 1809.

Gauss



Legendre

Least squares principle

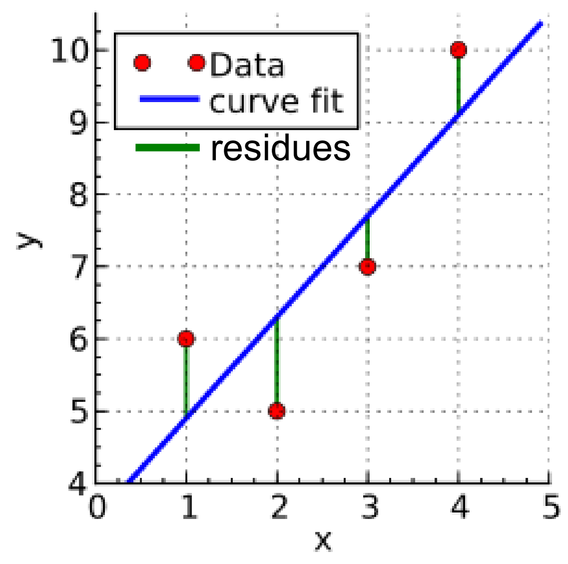
- Problem formulation

$$y_i = f(x_i; \beta)$$

$$r_i = y_i - f(x_i; \beta)$$

Cost function: $S = \sum_i r_i^2$ "sum of squares"

- Basic idea: minimize squared residues



$$\hat{\beta} = \min_{\beta} S$$

Optimization

- Find minimum/maximum of objective function (in our case: the cost function)

$$\min_x f(x) \qquad \left(\text{or} \quad \max_x f(x) \right)$$

- Inequality constraints

$$g(x) \leq 0$$

- Equality constraints

$$h(x) = 0$$

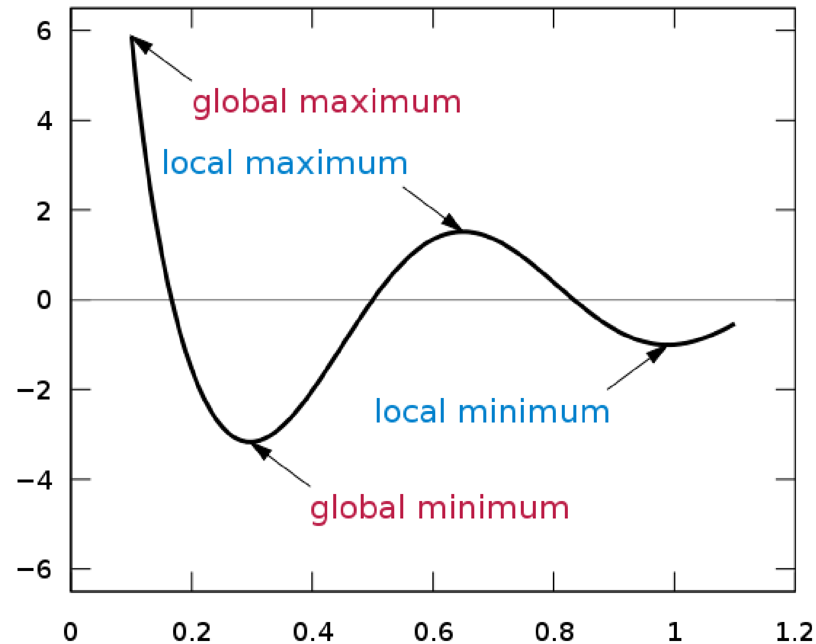
- Standard: minimization problem (negation of maximization problem)

Global/Local Minima/Maxima

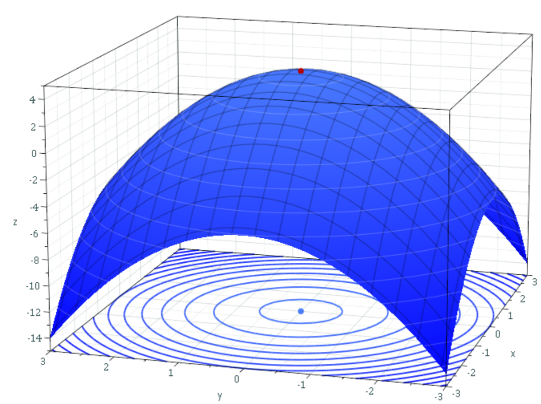
- Find extremal point of function

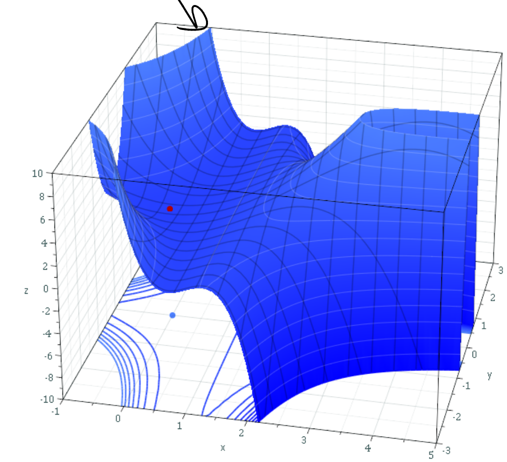
$$\frac{\partial S}{\partial \beta} = 0 \qquad \text{optimum}$$

$$\nabla_\beta S = 0$$



non convex





- Convex problems:

$\Downarrow$

→ local minimum is also global minimum

- All linear problems are convex!

Linear least squares

- Problem formulation

$$y_i = f(x_i; \vec{\beta})$$

$$= \sum_j \beta_j X_{ij}$$

$$\vec{y} = X \cdot \vec{\beta}$$

measurements

$$\begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_N \end{pmatrix} = \begin{pmatrix} M \times N & \\ & \\ X_{ij} & \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{pmatrix}$$

- Minimize cost function

$$\min_{\vec{\beta}} \sum_i \left| y_i - (X\beta)_i \right|^2$$

$$\sum_i \left| y_i - \sum_j x_{ij} \beta_j \right|^2$$

minimization reduces to solving linear problem

Example: Expectation value

- Given a set of random numbers, find an estimate for the expectation value of the underlying probability distribution

y_i : data $\qquad E(y) = \mu$ ("$= \beta$")

$$S(\mu) = \sum_i (y_i - \mu)^2 \longleftarrow \text{model}$$

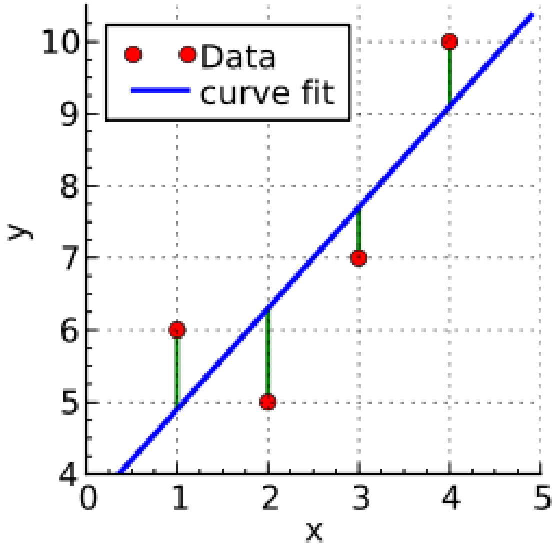
$$\frac{\partial S}{\partial \mu} = 2 \sum_i (\mu - y_i) = 0$$

$$N\mu - \sum_i y_i = 0 \implies \mu = \underbrace{\frac{1}{N} \sum_i y_i}_{\text{mean}}$$

mean value is a least square estimator for the expectation value

Example: Linear regression

- Given a set of measurements, find the parameters of a linear regression model



Example: Deconvolution

- Problem

known:

- measured image: g
- convolution kernel (PSF): h

original image: f

$$g = h * f \xrightarrow{\mathcal{F}} G = H \cdot F$$

$$F = G/H$$

$$f = \mathcal{F}^{-1}\left\{ G/H \right\}$$

becomes very large for high spatial frequencies

in reality: $G + N$

$$f = \mathcal{F}^{-1}\left\{ \frac{G}{H} + \frac{N}{H} \right\}$$

Original Blurred Wiener filtered

Example: Deconvolution

- Search for optimal filter W that minimizes least squares

model: $g = \underbrace{h * f + n}$

Cost function: $\epsilon\left(\sum_i |f_i - \hat{f}_i|^2\right)$

$\hat{f} = w * g$ optimal filter

Parsival theorem

$\sum_k |F_k - \hat{F}_k|^2 = \|F - \hat{F}\|^2$

$S = \epsilon\left(\sum_k |F_k - \hat{F}_k|^2\right)$

$= \epsilon\left(\|F - W \cdot G\|^2\right)$

$= \epsilon\left(\|F - W(H \cdot F + N)\|^2\right)$

$= \epsilon\left(\|F(1 - WH) - WN\|^2\right)$

$= \epsilon\left(\sum_k |F_k(1 - W_k H_k)|^2 + \sum_k W_k^2 N_k^2\right) - \underbrace{\epsilon\left(2\sum_k F_k(1 - W_k H_k) W_k N_k\right)}_{0}$

averages out

$\epsilon(N) = 0$

$\epsilon(N^2) \neq 0$

Example: Deconvolution

P_{sk} : Signal power spectrum $E(|F_k|^2)$

- Search for optimal filter W that minimizes least squares

Noise power spectrum $E(|N_k|^2)$

$$S = \sum_k P_{sk} \left| 1 - W_k H_k \right|^2 + \sum_k W_k^2 P_{Nk}$$

$$(1 - W_k H_k)(1 - W_k^* H_k^*)$$

$$\frac{\partial S}{\partial W_k^*} = P_{sk}(1 - W_k H_k)(-H_k^*) + P_{Nk} W_k = 0$$

$$-H_k^* P_{sk} + W_k |H_k|^2 P_{sk} + P_{Nk} W_k = 0$$

$$W_k \left(P_{Nk} + |H_k|^2 P_{sk} \right) = H_k^* P_{sk}$$

$$W_k = \frac{H_k^*}{|H_k|^2 + \frac{P_{Nk}}{P_{sk}}}$$

$\frac{1}{SNR}$

Wiener filter

General linear least squares

Solve $\vec{y} = X \cdot \vec{\beta}$ $\mathbb{C}$ ("$Ax = b$")

matrix "X" "β" "y" (data)

$f : \mathbb{C} \to \mathbb{R}$

$f(z)$

$\frac{\partial f}{\partial z} = 0$ or $\frac{\partial f}{\partial z^*} = 0$

gives optimum

min $S(\vec{\beta})$

$$S = \|\vec{y} - X\vec{\beta}\|^2 = \sum_i \left| y_i - \sum_j X_{ij} \beta_j \right|^2$$

$$= \sum_i \left(\sum_j X_{ij} \beta_j - y_i \right) \left(\sum_j X_{ij}^* \beta_j^* - y_i^* \right)$$

$$\frac{\partial S}{\partial \beta_j^*} = \sum_i \left(\sum_j X_{ij} \beta_j - y_i \right) X_{ij}^* = 0$$

$$\sum_{ij} X_{ij} \beta_j X_{ij}^* = \sum_i X_{ij}^* y_i$$

$$\sum_{ij} (X^\dagger)_{ji} X_{ij} \beta_j = \sum_i (X^\dagger)_{ji} y_i$$

$$X^\dagger X \vec{\beta} = X^\dagger \vec{y}$$

$$\vec{\beta} = \underbrace{(X^\dagger X)^{-1} X^\dagger}_{} \vec{y}$$

Moore-Penrose pseudoinverse

"best" inverse in the least square sense

Least square optimization

General linear least squares

Example: fit a plane to a 2D image

$$y = X\beta$$

$$I(i,j) = A + Bi + Cj + Di^2 + Ej^2 + Fij$$

1D vector

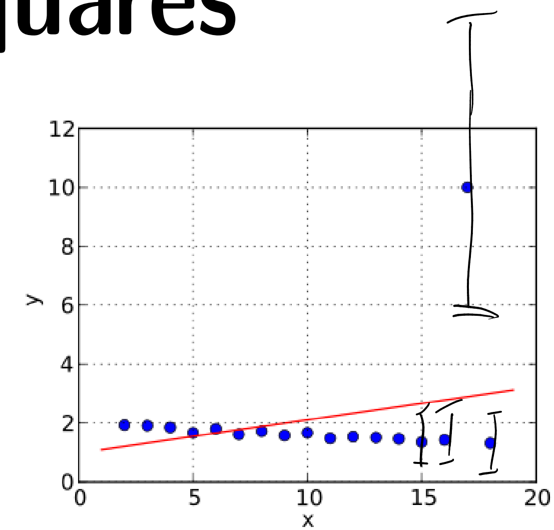
$$\begin{bmatrix} I(0,0) \\ I(1,0) \\ I(2,0) \\ \vdots \\ I(0,1) \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & \vdots & \\ 1 & \vdots & \\ 1 & & \\ 1 & 0 & 1 \\ \vdots & & \vdots \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 4 & 0 & 0 \\ 1 & \vdots & \vdots & \vdots & & \\ 1 & \vdots & \vdots & \vdots & 1 & \\ 1 & \vdots & 1 & \vdots & 4 & \\ 1 & \vdots & 1 & 1 & \vdots & \\ & & 1 & & 1 & \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \end{bmatrix}$$

Weighted least squares

- Problem: sensitivity to outliers

$$S = \sum_i w_i r_i^2 \qquad w_i = \frac{1}{\sigma_i^2}$$



- Solution: penalize problematic values using weights

$$\hat{\beta} = \min_{\beta} \left\| w^{\frac{1}{2}} (X\vec{\beta} - \vec{y}) \right\|^2$$

$w^{\frac{1}{2}}$ diagonal matrix

$$\begin{pmatrix} \frac{1}{\sigma_1} & & 0 \\ & \frac{1}{\sigma_2} & \\ 0 & & \ddots \end{pmatrix}$$

$$\hat{\vec{\beta}} = (X^T w X)^{-1} X^T w \vec{y}$$

Solving least squares problems

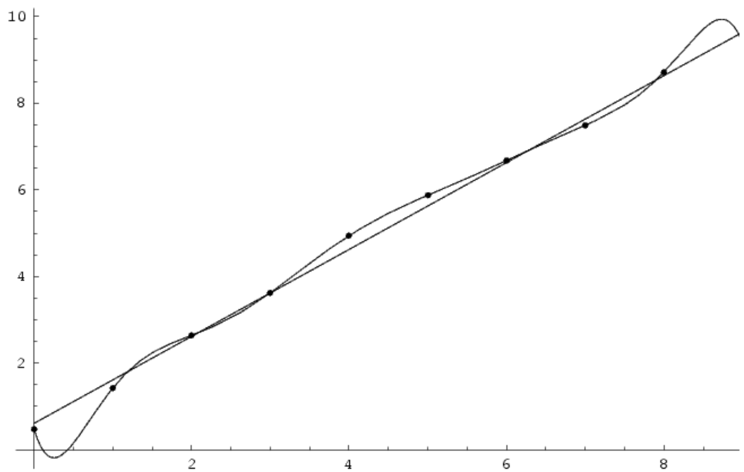
- Many approaches to solution exist
  - Pseudo inverse $(X^T X)^{-1} X^T$
  - Singular value decomposition (SVD)
  - QR decomposition
  - Iterative methods
  - ...

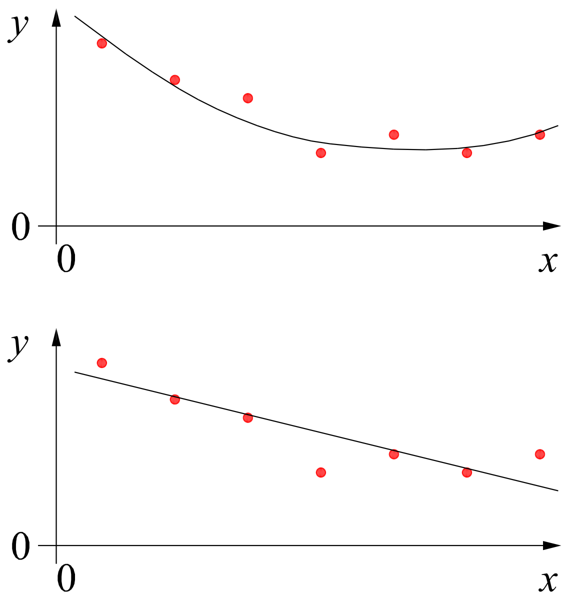
$X^{+} = (X^T)^*$

- Choice depends on
  - Robustness
  - Speed
  - Memory consumption
  - ...

Overfitting & ill-defined problems

- Guess can only be as good as the underlying model
- Too complicated models can lead to too complicated solutions





- Simultaneous optimization of model and its parameters
- Need *regularization* ← constraints on parameter space

Lagrange multipliers

- Optimization under equality constraints

$$\max_{x,y} f$$

$$\downarrow$$

$$\nabla f = 0$$

replaced with

$$\nabla f = -\lambda \nabla g \quad , \quad g(x,y) = c$$

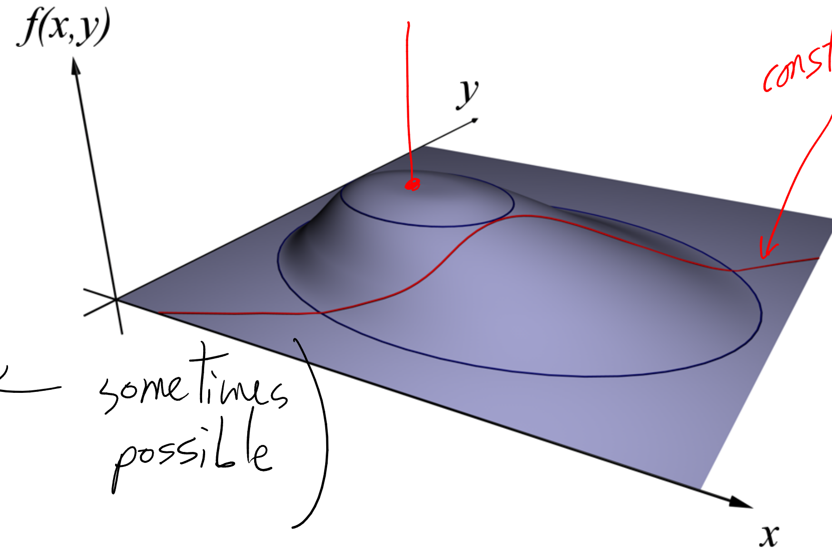
$g(x,y)=c$

(e.g. $y = mx+b$

$f(x, mx+b)$ ← sometimes possible)

(x,y) maximizing f

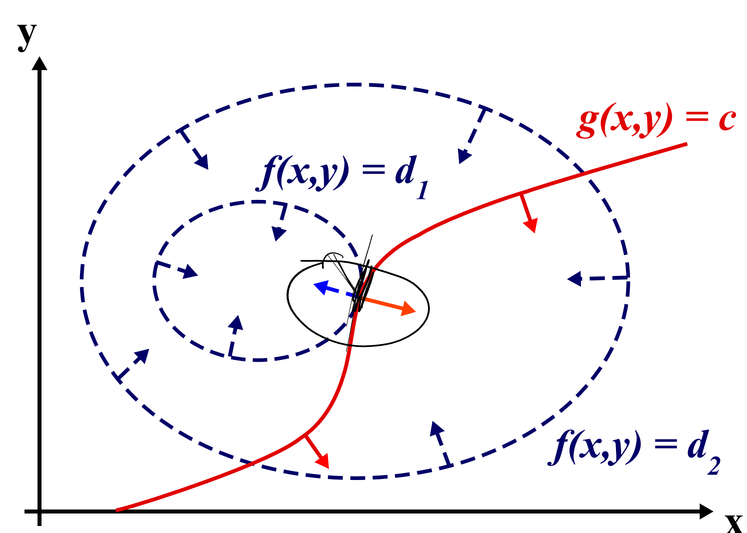
constraint



$$\boxed{\mathcal{L} = f + \lambda\left(g(x,y) - c\right)}$$

$$\nabla \mathcal{L} = \nabla f + \lambda \nabla g = 0$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = g(x,y) - c = 0$$



Tikhonov Regularization

$$S = \| \vec{y} - X\vec{\beta} \|^2$$

$$\mathcal{L} = \| \vec{y} - X\vec{\beta} \|^2 + \lambda \| \beta \|^2$$

avoids "explosion" of parameters. Makes a numerical problem well-conditioned

$$\nabla \mathcal{L} = 0 \longrightarrow \vec{\beta} = \left(X^{\dagger} X + \underbrace{\lambda \mathbb{1}} \right)^{-1} X^{\dagger} \vec{y}$$

e.g. Lagrange multiplier:

A, B: two images $\in \mathbb{C}$ that differ only by $e^{i\varphi}$

Problem: find φ.

$$S = \| A - e^{i\varphi} B \|^2 \leftarrow \text{possible.}$$

$$S = \| A - zB \|^2 + \lambda \left(|z|^2 - 1 \right)$$

$$\frac{\partial S}{\partial z^*} = \sum_i \left(z B_i - A_i \right) B_i^* + \lambda z = 0$$

$$z \left(\lambda + \sum_i |B_i|^2 \right) = \sum_i A_i B_i^*$$

$$\longrightarrow z = \frac{\sum_i A_i B_i^*}{\lambda + \sum_i |B_i|^2}$$

$$\rightarrow \varphi = \arg \left(\sum_i A_i B_i^* \right)$$

Nonlinear least squares

- If possible: linearize

- Linearization not possible? $\rightarrow$ iterative solution, brute force search, etc...

Example: Image registration

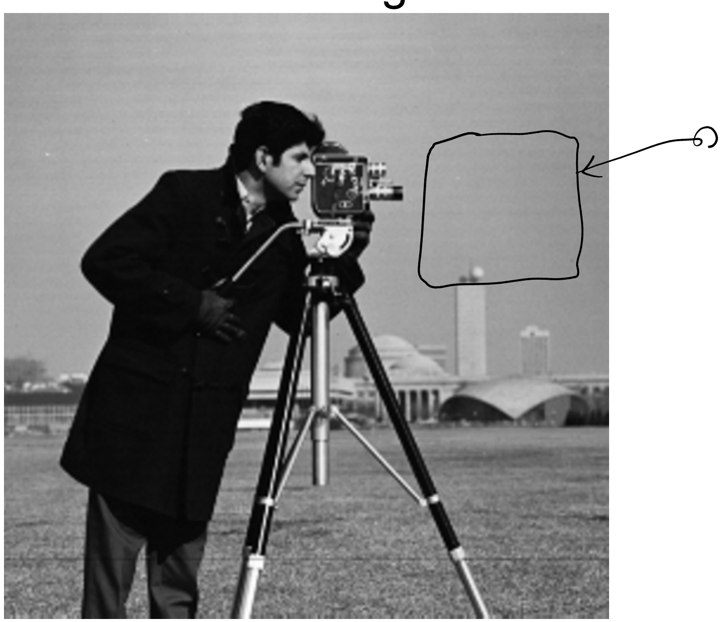
- Problem formulation: estimate the parameters of a transform s.t. the difference between original and distorted image is minimal

$$\underset{\vec{r}_0}{\text{minimize}} \; \| B(\vec{r}) - T(\vec{r} - \vec{r}_0) \|^2$$

$$S(\vec{r}_0) = \sum_{\vec{r}} m(\vec{r} - \vec{r}_0) \left(B(\vec{r}) - T(\vec{r} - \vec{r}_0) \right)^2$$

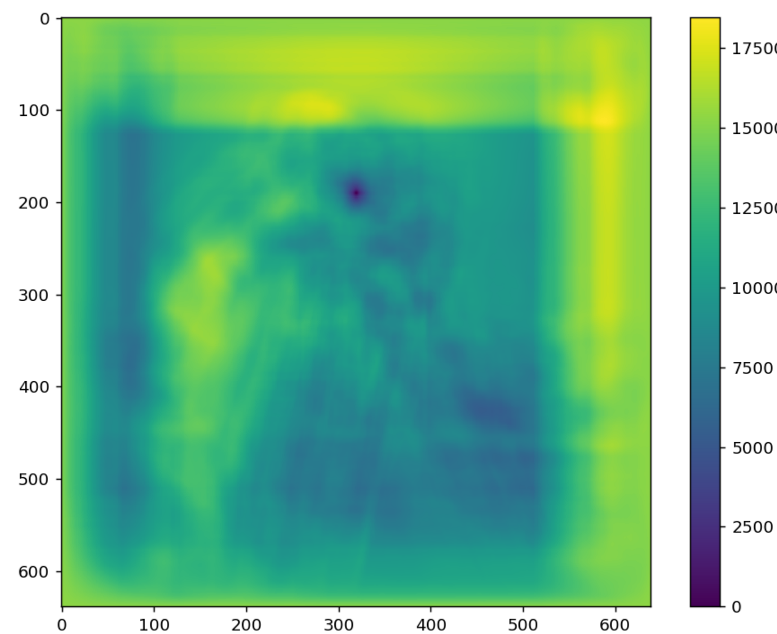
$$= \sum_{\vec{r}} m(\vec{r} - \vec{r}_0) B^2(\vec{r}) + \sum_{r} m(\vec{r} - \vec{r}_0) T^2(r - \vec{r}_0) - 2 \sum_{\vec{r}} B(\vec{r}) \, m(r - \vec{r}_0) \, T(\vec{r} - \vec{r}_0)$$

Base image



Template

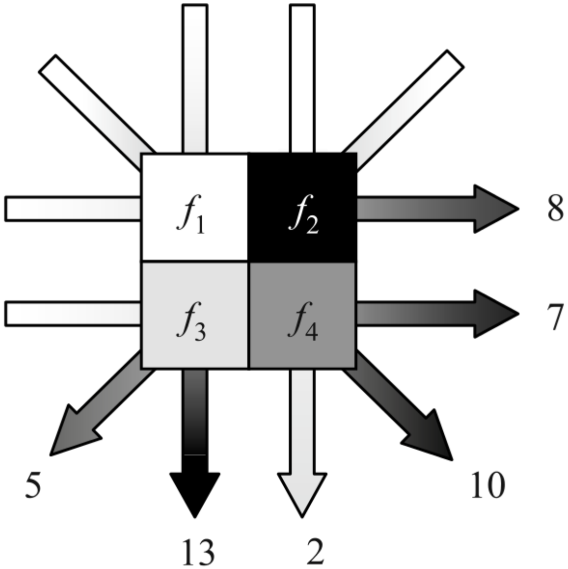
Distance map



Iterative solutions

- Move towards optimum in steps
  - Gradient descent
  - Newtons method
  - Gauss-Newton algorithm
  - Conjugate gradients
  - ...
- Projection onto constraint sets

Tomography revisited

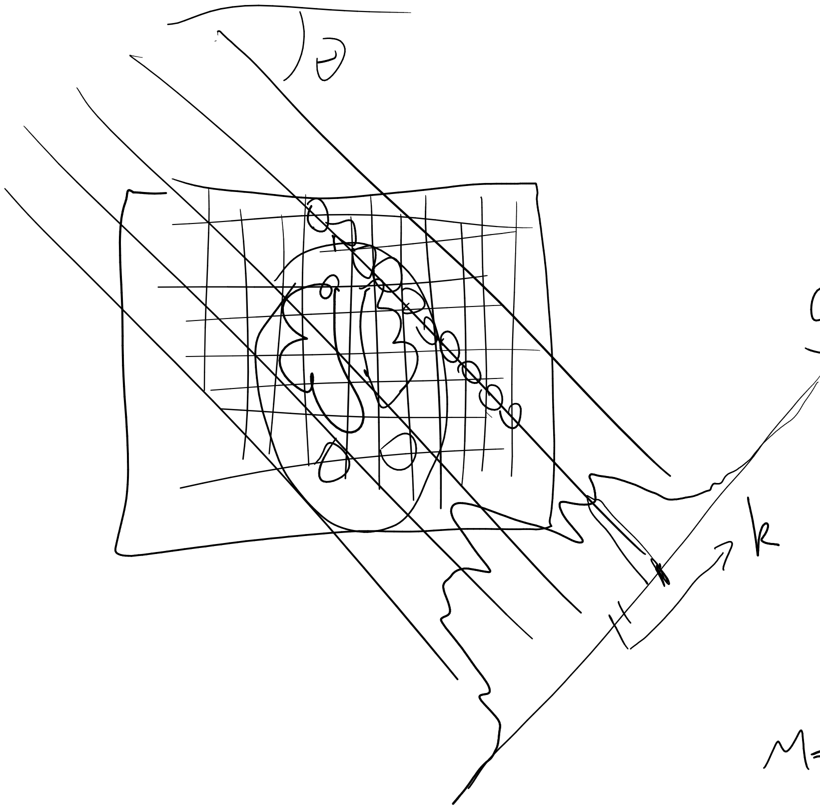


Radon transform

$$S = M \cdot T$$

sinogram — "system matrix" — tomogram

M is sparse (lots and lots of 0's)
but its (pseudo) inverse is not sparse
and it is <u>big</u>



S^θ

Iterative methods break up the problem

For one angle θ

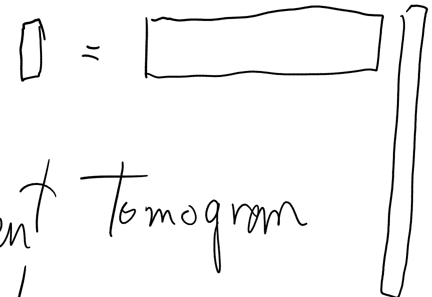
$$S^\theta = A^\theta T$$

$$S^\theta_k = \sum_j A^\theta_{kj} T_j$$

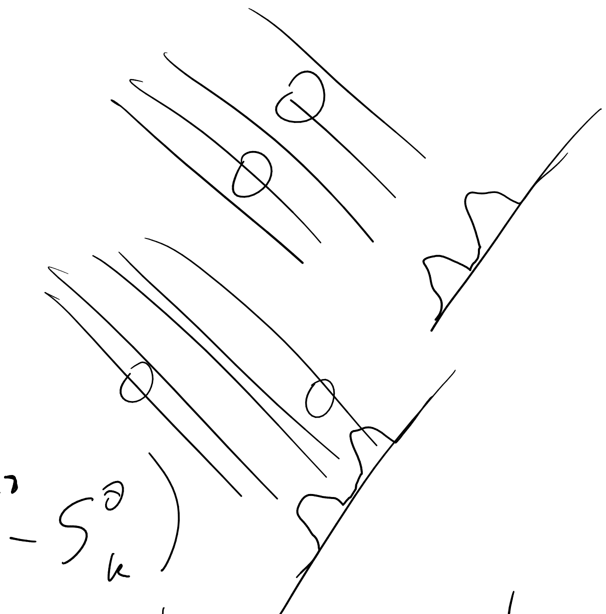
$$M = \begin{pmatrix} A^{\theta_1} & 0 & 0 \\ & A^{\theta_2} & 0 \\ 0 & 0 & A^{\theta_3} \\ & & \ddots \end{pmatrix}$$

Algebraic reconstruction techniques

$$S_k^\theta = \sum_j A^\theta_{kj} T_j \quad (*)$$



under-determined problem



Problem formulation: given current tomogram estimate T, what is the new tomogram that satisfies (*) with minimal change to T.

new tomogram

$$D = \sum_j \left(T_j - T_j'\right)^2 + \underbrace{\sum_k \lambda_k \left(\sum_j A^\theta_{kj} T_j' - S_k^\theta\right)}_{\text{Lagrange multipliers imposing constraint } (*)}$$

$$\frac{\partial D}{\partial T_j'} = 2\left(T_j' - T_j\right) + \sum_k \lambda_k A^\theta_{kj}$$

$$\boxed{T_j' = T_j - \frac{1}{2} \sum_k \lambda_k A^\theta_{kj}}$$

$$T' = T - \frac{1}{2} A^{\theta T} \lambda$$

Least square optimization

Algebraic reconstruction techniques

To find λ_k impose constraint on the solution (general feature when working with Lagrange multipliers)

$$S_m^\theta = \sum_j A_{mj}^\theta \left(T_j - \frac{1}{2} \sum_k A_{kj}^\theta \lambda_k \right)$$

$$= \sum_j A_{mj}^\theta T_j - \frac{1}{2} \sum_{jk} A_{mj} A_{kj} \lambda_k$$

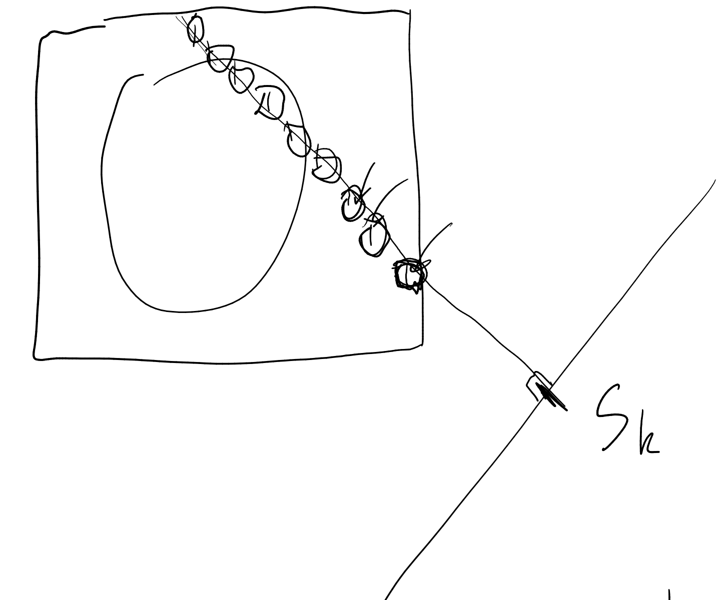
$$= (A T)_m - \frac{1}{2} (A A^T \lambda)_m \longrightarrow \frac{1}{2} A A^T \lambda = P - S$$

(projection = P_m)

$$\boxed{\lambda = 2 (A A^T)^{-1} (P - S)}$$

will be 0 if constraint is already satisfied!

Algebraic reconstruction techniques



$$S_k = \sum_j A_{kj} T_j$$

1 if on the k^{th} ray
0 otherwise

$$\sum_k (A^T)_{jk} S_k \leftarrow \text{"spread" } S_k \text{ over } k^{th} \text{ ray within } T$$

A^T = backprojection

$(AA^T) \simeq$ diagonal matrix
with entries = # voxel on
a given ray

"SART"

$$T' = T + A^T (AA^T)^{-1} (S - P)$$

$$T_j' \simeq T_j + \sum_k A_{kj} \frac{1}{N_k} (S_k - P_k)$$

number of voxel on k^{th} ray.

Algebraic reconstruction techniques

Recipe for SART:

1. pick θ (loop through all angles)
2. compute $P = A^{\theta} T$
3. compute $\Delta S = S - P$ "sinogram error"
4. multiply with $\frac{1}{N_k}$ "ray length" $^{-1}$
5. Back project
6. Add to tomogram estimate

Summary

- Approximate solutions can be found using estimation
- Approximation quality can be quantified by cost function
- Optimum solution is found by minimizing the cost function
- Least square estimator minimizes squared residues
- Lagrange multipliers can be used to implement additional constraints
- Iterative schemes allow solution of hard problems