ENGSCI 721

INVERSE PROBLEMS

Oliver Maclaren
oliver.maclaren@auckland.ac.nz

MODULE OVERVIEW

3. *Preview of the statistical view of inverse problems*
   [1 lectures]

   Bayesians, Frequentists and all that. Basic frequentist analysis.

MODULE OVERVIEW

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

1. *Basic concepts* [4 lectures]
   Forward vs inverse problems. Well-posed vs ill-posed problems. Algebra and calculus of inverse problems (matrix calculus, generalised inverses etc). Regularisation and trade-offs.
2. *More regularisation* [3 lectures]
   Higher-order Tikhonov regularisation, truncated singular value decompositions, iterative regularisation.

LECTURE 5: REGULARISATION - TIKHONOV AND BEYOND

Topics:

- Recap of Tikhonov
- Higher-order Tikhonov
- L_1 norm regularisation
  - Sparsity
  - Total variation
  - Robust regression

Eng Sci 721 : Lecture 5

Regularisation - Tikhonov & beyond

- Recap: basic Tikhonov
- Higher-order Tikhonov
- L_1 norm
  - Sparsity
  - Total variation
  - Robust regression

Recap: Basic Tikhonov

The standard Tikhonov approach to constructing regularised solutions to inverse problems like

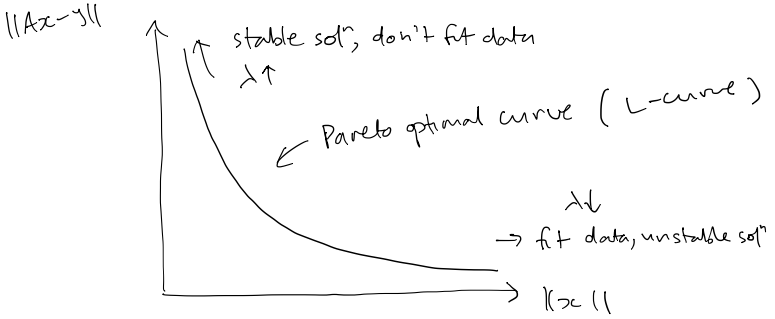
$$\boxed{\text{find } x \text{ satisfying } Ax = y}$$

where A tends to 'smooth' or 'reduce' x, i.e. $x \to y$, is to consider the modified problem:

$$\boxed{\min_{x(\lambda)} \|Ax - y\|^2 + \lambda \|x\|^2}$$

Note that the solution depends on the regularisation parameter λ, hence we write $x(\lambda)$

The parameter λ represents a trade-off between the importance of fitting the given data y & having a 'simple' or 'stable' solution



→ since the data y is not exact / exactly repeatable, we use λ to satisfy Hadamard's third condition of stability:

> 'small changes to given info should given small changes to solution'

→ Choosing λ is somewhat of an art & part of designing a regularisation procedure

↳ eg { L-curve corner ← we looked at.
discrepancy principle
cross-validation
etc.

When we use the squared L_2 norm for both, it is convenient to write this as the augmented least squares problem:

$$\min \|\tilde{r}\|^2 = \min \|\tilde{A}x - \tilde{y}\|^2$$

where $\tilde{A}x = \tilde{y}$ is $\begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix} x - \begin{bmatrix} y \\ 0 \end{bmatrix}$

→ We can then use any standard least-squares software to solve.

Eg • Matlab: lsqminnorm
• Python: np.lstsq (does min norm if over-det).

Can also use pinv since this gives least squares sol^n. } Slightly less eff alg. I think

(Note: Matlab \ gives different pseudoinverse!)

Linear vs nonlinear

The linear case has the explicit solution

$$x = (A^T A + \lambda I)^{-1} A^T y$$

ie

$$x = A_\lambda^{\#} y \quad \text{where} \quad A_\lambda^{\#} = (A^T A + \lambda I)^{-1} A^T$$

In the nonlinear case $F(x) = y$ we can still solve the 'variational form':

$$\min_{x(\lambda)} \|F(x) - y\|^2 + \lambda \|x\|^2$$

or

$$\min \|\tilde{r}\|^2 = \min \|\tilde{F}(x) - \tilde{y}\|^2$$

where is

$$\tilde{F}(x) = \tilde{y}$$

$$\begin{bmatrix} F(x) \\ \sqrt{\lambda}\, x \end{bmatrix} = \begin{bmatrix} y \\ 0 \end{bmatrix}$$

→ no explicit inverse, but...

→ solve via nonlinear least-squares, or any minimisation alg (see later)

Generalisations

The variational form is easiest to generalise

→ nonlinear problems ✓

→ different norms for data/model space ✓

Just gives an objective function to minimise via any minimisation alg.

→ scipy.minimize library (Python)

→ fminsearch (Matlab; see also opt. toolbox)

- First, however, we look at forms that still fit in least squares framework
- Then forms that fit into other efficient frameworks
  - convex opt. (eg Boyd & Vandenberghe)
- Finally (later lectures), generic problems & alternative approaches (iterative reg.)

Higher-order Tikhonov

The standard model norm that we've seen measures 'size' via

$$\|x\| \quad \text{or} \quad \|x\|^2$$

→ we can generalise this by considering

$$\|Dx\| \quad \text{or} \quad \|Dx\|^2$$

for some operator

(or $\|Dx - x_0\|$ esp. for nonlinear prob)

→ Typically D represents a first or second derivative operator, (or even a differential eqⁿ !)

↳ can consider diff. operators as 'roughening' operators (cf. integration as smoothing)

↳ we prefer smoother ie less variable solutions, hence penalise roughness.

General Tikhonov: (can be linear or nonlinear)

$$\min \|Ax - y\|_2^2 + \lambda \|Dx\|_2^2$$

where,

(for discrete, linear diff. operators in 1D → can generalise to higher D):

'zeroth order'

$$D_0 = I = \begin{bmatrix} 1 & & \\ & 1 & \\ & & \ddots \\ & & & 1 \end{bmatrix}$$

'first order'

$$D_1 = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{bmatrix}$$

Second order

$$D_2 = \begin{bmatrix} 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & & 1 & -2 & 1 \end{bmatrix}$$

Note:

$$D_1 x = \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$= \begin{bmatrix} x_2 - x_1 \\ x_3 - x_2 \\ \vdots \\ x_n - x_{n-1} \end{bmatrix}$$

ie first order forward finite differences

$$D_2 x = \begin{bmatrix} 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$= \begin{bmatrix} x_1 - 2x_2 + x_3 \\ \vdots \\ x_{n-2} - 2x_{n-1} + x_n \end{bmatrix}$$

ie second-order central finite differences

Also:

$$D_1^2 = D_1 \cdot D_1 = \begin{bmatrix} -1 & 1 & \\ & -1 & 1 \\ & & -1 \end{bmatrix} \begin{bmatrix} -1 & 1 & \\ & -1 & 1 \\ & & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -2 & 1 \\ & 1 & -2 \\ & & 1 \end{bmatrix}$$

$$= D_2$$

---

or 'wrapping' (ie 'circular differences')

$$D_{1c}^2 = D_{1c} D_{1c} = \begin{bmatrix} -1 & 1 & \\ & -1 & 1 \\ 1 & & -1 \end{bmatrix} \begin{bmatrix} -1 & 1 & \\ & -1 & 1 \\ 1 & & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -2 & 1 \\ 1 & 1 & -2 \\ -2 & 1 & 1 \end{bmatrix}$$

$$= D_{2c}$$

↑ 'circular'

Least squares form

Just like before, when working with the L_2-norm we can re-write

$$\min_{x(\lambda)} \|Ax - y\|_2^2 + \lambda \|Dx\|_2^2$$

as

$$\min_{x(\lambda)} \left\| \begin{bmatrix} A \\ \sqrt{\lambda} D \end{bmatrix} x - \begin{bmatrix} y \\ 0 \end{bmatrix} \right\|_2^2$$

ie

$$\min_{x(\lambda)} \|\tilde{A}x - \tilde{y}\|_2^2$$

where $\tilde{A} = \begin{bmatrix} A \\ \sqrt{\lambda} D \end{bmatrix}$

& use standard least-squares software (linear or nonlinear)

Explicit solution (linear only)

In the linear case we have the explicit solution

$$x = (A^T A + \lambda D^T D)^{-1} A^T y$$

ie

$$x = A^{\#}_{D,\lambda} y$$ (or just $x = A^{\#} y$ if lazy)

where $$A^{\#}_{D,\lambda} = (A^T A + \lambda D^T D)^{-1} A^T$$

[Note that we technically have to assume that $A^T A + \lambda D^T D$ is invertible

-> obvious for $D = I$ (see before)

-> true for $D = D_1$ or D_2 too

condition: $N(A) \cap N(D) = \{0\}$

ie null spaces have only trivial soln in common]

Examples

- Simple smoothing

$$x_{noisy} = x_{smooth} + \epsilon = I x_{smooth} + \epsilon$$

$$= A x_{smooth} + \epsilon \,, \quad \epsilon \text{ unknown}$$

$$\Rightarrow \boxed{A = I}$$ ('deterministic' part; ϵ dealt with by using least squares sol')

Note: we don't explicitly model ϵ, but relates to choice of 'fit'

Here: $\| A x_{smooth} - x_{noisy} \|_2^2$

→ least squares fit (don't expect exact fit when noisy).

→ but without reg., will fit exactly

Regularised:

$$\boxed{x_{smooth} = \left(I + \lambda D^T D \right)^{-1} I \, x_{noisy}}$$

solves: $\| x_{noisy} - x_{smooth} \|_2^2 + \lambda \| D x_{smooth} \|_2^2$

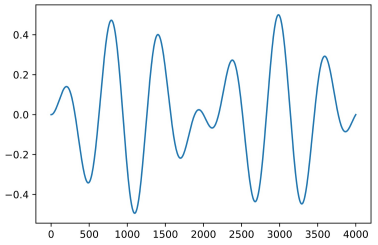
fit data — smooth data.

Example

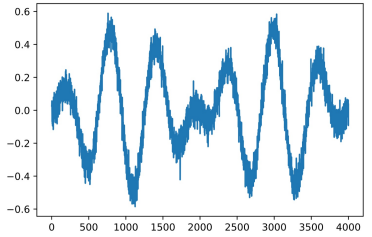
[see Boyd & Vandenberghe 6.3.3 (attached) for original]

My attempt

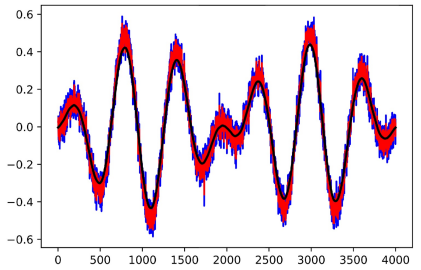
True signal



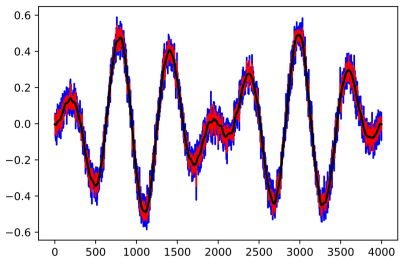
Noisy signal



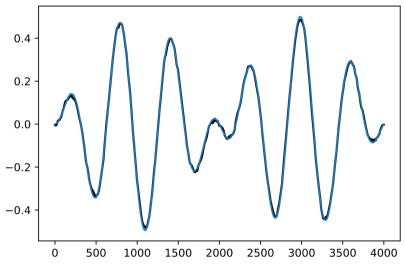
Recovered (D_1 reg.) diff. λ



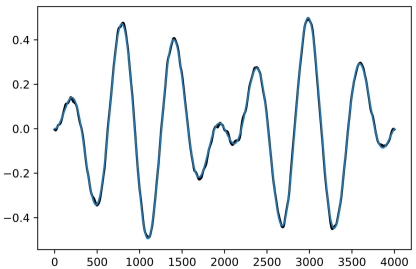
Recovered (D_2 reg.) diff. λ



'Best D_1' & True



'Best D_2' & True



Other norms etc

'Tikhonov' regularisation generally refers to the L_2 norm for both 'data space' & 'model space'

→ Useful computationally (eg re-frame as standard least squares, differentiable...)

→ But other norms can be useful to emphasise different data &/or model features

mix & match data & model norms

In particular, the L_1 norm is closely related to eg

L_1 regularisation
- Sparsity / compressed sensing
- Total variation reconstruction

L_1 objective / loss function
- Robust regression

(The L_0 & L_∞ norms also come up fairly often)

L_1 norm & Lp norms

The L_1 norm has the form

$$\|x\|_1 = |x_1| + |x_2| + \cdots |x_n|$$

Both the L_1 & L_2 norms are special cases of L_p norms:

$$\|x\|_p = \left(|x_1|^p + |x_2|^p + \cdots |x_n|^p\right)^{1/p}$$

for $p \geq 1$.

The L_∞ norm is the limit as $p \to \infty$ and given by

$$\|x\|_\infty = \max\left\{|x_1|, |x_2|, \cdots |x_n|\right\}$$

The L_0 'norm' is not technically a norm but means 'number of non-zero' entries in x

ie $\|x\|_0 = \text{card}(x)$

where 'card' means 'cardinality'.

L_1 & L_0 'norms': Sparsity.

The L_0 'norm' measures 'sparsity'

→ small L_0 'norm' means few non-zero entries } 'sparse'

→ can get 'simple' solutions in sense of 'only a few non zero components'

→ Unfortunately is basically just enumeration & very difficult to work with L_0 directly

A famous & somewhat surprising result is that the L_1 norm, when used as regularisation also tends to produce sparse solutions, ie solutions with exactly zero elements

Furthermore: (linear) L_1 problems lead to

convex optimisation problems

that can still be solved relatively efficiently

(eg Linear / Quadratic Programming etc)

Equivalence of norms?

In contrast to L_0/L_1, L_2 produces solutions with very small but non-zero elements

But aren't all norms in $\mathbb{R}^n$ 'equivalent'? ie

$$\exists\, \alpha, \beta \text{ s.t. } \alpha \|x\|_a \leq \|x\|_b \leq \beta \|x\|_a$$

for any two norms $\|\cdot\|_a$ & $\|\cdot\|_b$

→ value of norm can be approximated, but the geometry & solutions favoured by different norms are different

↳ also, α, β matter! (& as $n \to \infty$ approx gets worse)

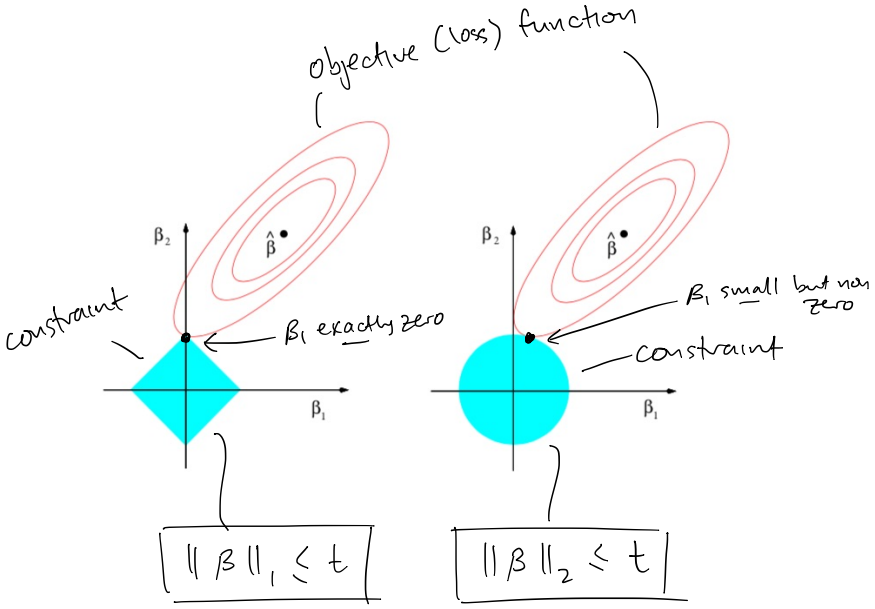
→ quantitatively similar evaluation of given solution but problems lead to qualitatively solutions as 'best'

L_2 : - lots of small but non-zero values
- related to 'average'

L_0/L_1 : - lots of exactly zero values
- related to 'median'
- more robust/less sensitive to large individuals than L_2.

L_1 vs L_2 regularisation

$\min \quad \|A\beta - y\|_2^2$	$\min \quad \|A\beta - y\|_2^2$
$\text{s.t.} \quad \|\beta\|_1 \leq t$	$\text{s.t.} \quad \|\beta\|_2 \leq t$



[From 'Statistical learning with sparsity' by Hastie et al.]

Applications of L_1: Sparse solutions

('LASSO' regression)

$$\min \|x\|_1 \quad \text{s.t.} \quad \|Ax - y\|_2 \leq \delta$$

same as before, but emph. model norm.

or variational form:

$$\min \|Ax - y\|_2^2 + \lambda \|x\|_1$$

key form.

→ Tikhonov-like, but L_1 on model/parameters x instead of L_2

→ Non-differentiable at $x = 0$

→ But: convex optimisation when A is linear

↳ use eg linear/quad. programming, iterative reweighted LS etc

↳ many existing libraries for LASSO!

→ Can also use differentiable approx. to L_1

→ or just derivative-free optimisation

etc.

Examples

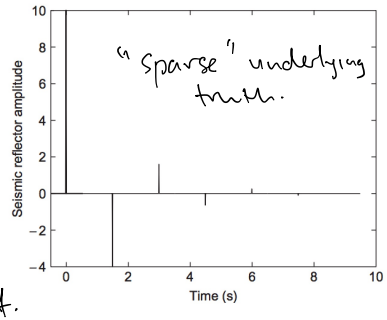
- LASSO regression – Google!
  - Hastie et al
  - sklearn (python machine learning)

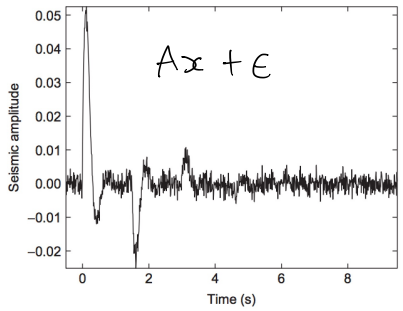
- Aster et al (Inverse Problems)

  Example 7.2 (Full detail attached →)

Deconvolution seismic sensing

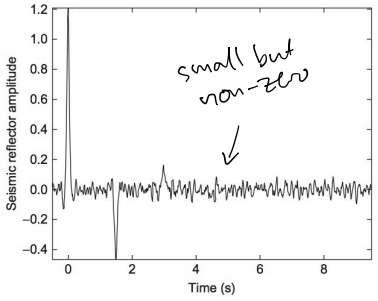
True signal → Observations



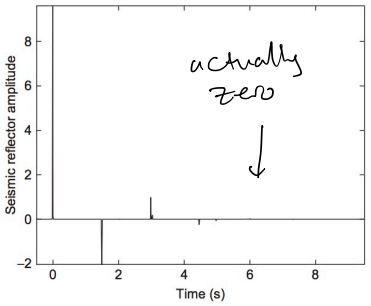


invert.

L_2 regularisation recovered signal:



L_1 regularisation recovered signal:



Applications of L_1: Total variation regularisation

$$\min \|D_1 x\|_1$$

$$\text{s.t. } \|Ax - y\|_2 \leq \delta$$

or variational form:

$$\boxed{\min \|Ax - y\|_2^2 + \lambda \|D_1 x\|_1}$$

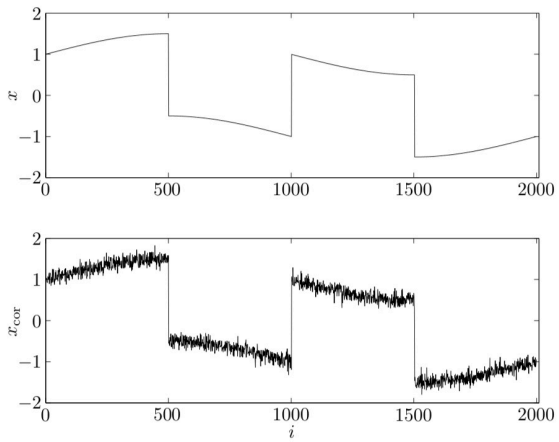
→ Tikhonov-like, but L_1 on model 'roughness' $D_1 x$ instead of L_2

→ piecewise constant (for D_1) but allows for small number of discontinuous 'jumps'

↳ good for recovering objects with 'sharp' features like edges in images

→ Solving: same basic idea as before (convex / LP / QP / IRLS etc)

Example: Boyd & V. 6.3.3



← True signal

← corrupted/noisy signal

Figure 6.11 A signal $x \in \mathbf{R}^{2000}$, and the corrupted signal $x_{\text{cor}} \in \mathbf{R}^{2000}$. The noise is rapidly varying, and the signal is mostly smooth, with a few rapid variations.

$\|Dx\|_2^2$ regularised sol^{n}s

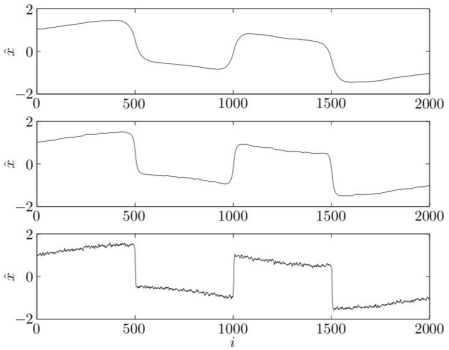


Figure 6.12 Three quadratically smoothed signals $\hat{x}$. The top one corresponds to $\|\hat{x} - x_{\text{cor}}\|_2 = 10$, the middle one to $\|\hat{x} - x_{\text{cor}}\|_2 = 7$, and the bottom one to $\|\hat{x} - x_{\text{cor}}\|_2 = 4$. The top one greatly reduces the noise, but also excessively smooths out the rapid variations in the signal. The bottom smoothed signal does not give enough noise reduction, and still smooths out the rapid variations in the original signal. The middle smoothed signal gives the best compromise, but still smooths out the rapid variations.

$\|Dx\|_1$ regularised sol^{n}s

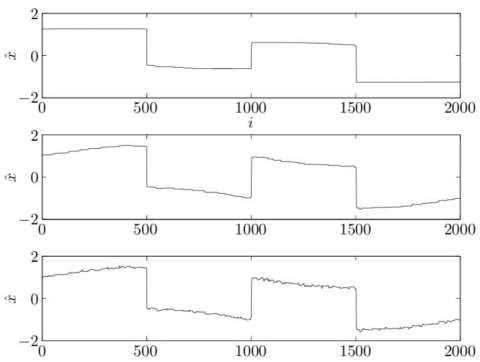


Figure 6.14 Three reconstructed signals $\hat{x}$, using total variation reconstruction. The top one corresponds to $\|D\hat{x}\|_1 = 5$, the middle one to $\|D\hat{x}\|_1 = 8$, and the bottom one to $\|D\hat{x}\|_1 = 10$. The bottom one does not give quite enough noise reduction, while the top one eliminates some of the slowly varying parts of the signal. Note that in total variation reconstruction, unlike quadratic smoothing, the sharp changes in the signal are preserved.

Applications of L_1: Robust regression

- Here we consider replacing the L_2 norm by the L_1 norm in the data fit term eg

$$\min \|Ax - y\|_1 + \lambda \|x\|_2^2$$

[Variational/Tikhonov form]

LAD: Least Absolute Deviation regression

- It turns out that, while minimising wrt the L_2 norm leads to the average as the best data fit, minimising wrt the L_1 norm leads to the median as the best data fit

→ This is much more robust to outliers (extreme observations)

→ Not as easy to work with (non-diff) as before, though still convex for linear

↳ Can eg formulate as a linear programming problem (see appendix)

↳ Again, many packages exist.

First: simple example of 'robustness'

$y = (2, 3, 5, 7, 8)$

$ave(y) = 5$

$med(y) = 5$

$y' = (2, 3, 5, 7, 80)$ ← outlier (here: data error?)

$ave(y') = 19.4 >> ave(y)$

$med(y') = 5 = med(y)$

Huber et al 'Robust Statistics':

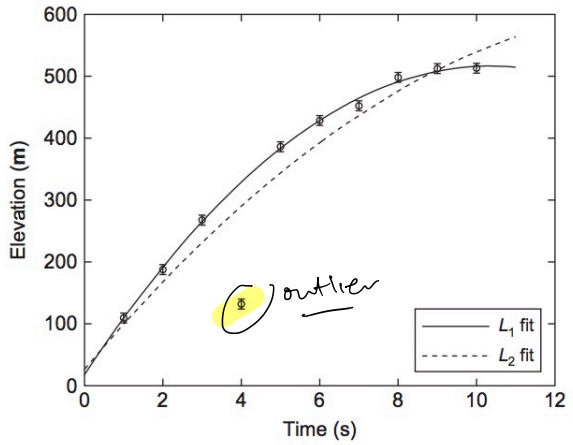
Perhaps the most important purpose of robustness is to safeguard against occcasional gross errors. Correspondingly, most approaches to robustness are based on the following intuitive requirement:

A discordant small minority should never be able to override the evidence of the majority of the observations.

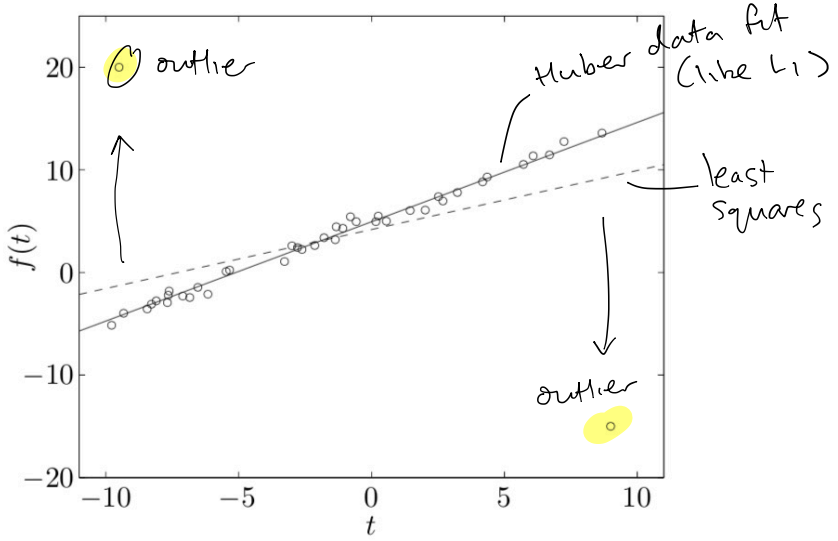
→ Trade-offs! Sensitivity vs Stability

Examples

(Aster et. al Example 2.4)



(Boyd & V. Example 6.2):



Appendix: reformulation of robust regression as linear programming (see eg Boyd & Vandenberghe)

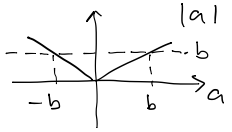
V1: $\boxed{\min \; \|Ax - y\|_1} = |r_1| + |r_2| + \ldots + |r_m|$

where $|r_i| = \begin{cases} r_i \text{ if } r_i > 0 \\ -r_i \text{ if } r_i < 0 \end{cases}$

→ almost linear (piecewise)

Note: $|a| \leq b$

equiv to $-b \leq a \leq b$



& minimising $|a|$

equiv to $\min_{a,b} b$

note: both var.

st. $|a| \leq b$

ie $-b \leq a \leq b$

proof? exercise :) (see also pic ↑)

So we use

V2: $\boxed{\begin{array}{l} \min_{t,x} \; I^T t + 0^T x \\ \text{st.} \; -t \leq Ax - b \leq t \end{array}}$

Linear programming (linear objective, linear constraints)