Linear Least Squares for estimating static parameters, using a "batch" of data.

$$\underline{\tilde{y}} = H\underline{\hat{x}} + \underline{e}$$

$$\begin{bmatrix} \tilde{y}_1 \\ \vdots \\ \tilde{y}_m \end{bmatrix} = \begin{bmatrix} h_1(t_1) & \cdots & h_n(t_1) \\ \vdots & & \vdots \\ h_1(t_m) & \cdots & h_n(t_m) \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_n \end{bmatrix} + \begin{bmatrix} e_1 \\ \vdots \\ e_m \end{bmatrix}$$

Given $\underline{\tilde{y}}$, find $\underline{\hat{x}}$ that minimizes the sum square of the residuals.

$$J = \frac{1}{2}\underline{e}^T\underline{e} \quad \leftarrow$$ cost function to minimize

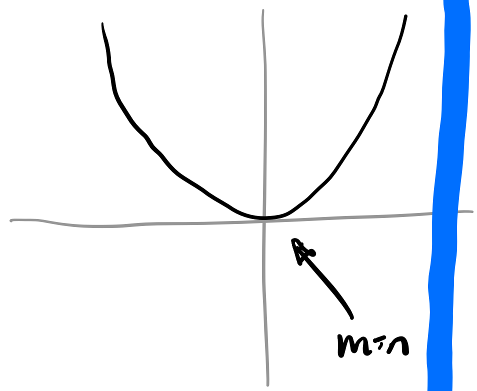
Plug $\underline{\tilde{y}}$ eqn into $\underline{e}$:

$$J = \frac{1}{2}\left(\underline{\tilde{y}}^T\underline{\tilde{y}} - 2\underline{\tilde{y}}^T H\underline{\hat{x}} + \underline{\hat{x}}^T H^T H\underline{\hat{x}}\right)$$

Find $\underline{\hat{x}}$ that minimizes J.

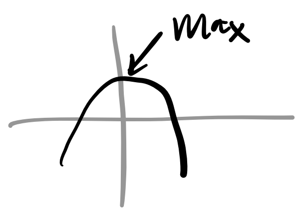
Remember a simpler case:

find x that minimizes $y = x^2$



(1) find where $\frac{\partial y}{\partial x} = 0$

(2) check to make sure $\frac{\partial^2 y}{\partial x^2} \geq 0$



(otherwise we could have found

Find $\hat{\underline{x}}$ that minimizes J.

(1) Need Jacobian of J to be zero

$$\nabla_{\hat{\underline{x}}} J \equiv \begin{bmatrix} \frac{\partial J}{\partial \hat{x}_1} \\ \vdots \\ \frac{\partial J}{\partial \hat{x}_n} \end{bmatrix} = H^T H \hat{\underline{x}} - H^T \tilde{\underline{y}} = 0$$

(2) Need Hessian of J to be $\geq$ zero

$$\nabla^2_{\hat{\underline{x}}} J \equiv \frac{\partial^2 J}{\partial \hat{\underline{x}} \, \partial \hat{\underline{x}}^T} = H^T H \text{ must be}$$

≥ 0 positive definite

positive definite:

if $\underline{x}^T B \underline{x} \geq 0$ for all $\underline{x} \neq 0$,

then B is positive definite

if B is symmetric (or Hermitian)

and all the eigenvalues of $B > 0$ & real

then B is pos-def.

$M^T M$ is always symmetric

if M is full rank, then

$M^T M$ will be pos-def.

from the Jacobian eqn we have:

$$(H^T H)\hat{\underline{x}} = H^T \tilde{\underline{y}}$$

if $H^T H$ is pos-def, ie. symmetric and

full rank (rank = n = # of states in $\underline{x}$)

then $H^T H$ can be inverted:

Least Squares Approx.

$$\boxed{\underline{\hat{x}} = \underbrace{(H^T H)^{-1} H^T}_{\text{this is the left psuedo inverse}} \underline{\tilde{y}}}$$

Implement on example 1 in note book.

Extensions

Weighted LLS: $\underline{\hat{x}} = (H^T W H)^{-1} H^T W \underline{\tilde{y}}$

W = weight matrix

use this if some measurements are of unequal precision.

optimal W: R^{-1}, the inverse of the measurement covariance matrix

Constrained LLS: Some measurements are perfect.

$$\begin{bmatrix} \tilde{\underline{y}}_1 \\ \vdots \\ \tilde{\underline{y}}_2 \end{bmatrix} = \begin{bmatrix} H_1 \\ \vdots \\ H_2 \end{bmatrix} \hat{\underline{x}} + \begin{bmatrix} \underline{e}_1 \\ \vdots \\ 0 \end{bmatrix}$$

perfect measurements → $\tilde{\underline{y}}_2$

original LLS

"correction"

we will see this form again later!

$$\hat{\underline{x}} = \bar{\underline{x}} + \overbrace{K(\tilde{\underline{y}}_2 - H_2 \bar{\underline{x}})}^{\text{"correction"}}$$

$$K = (H_1^T W_1 H_1)^{-1} H_2^T \left[H_2 (H_1^T W_1 H_1)^{-1} H_2^T \right]^{-1}$$

$$\bar{\underline{x}} = (H_1^T W_1 H_1)^{-1} H_1^T W_1 \tilde{\underline{y}}_1$$

Skip:

Matrix Decompositions: we will come back to these

Computing $(H^T H)^{-1}$ can be expensive or difficult.

Alternatives include: QR & SVD

Review

orthogonal vectors: u & v are orthogonal if the angle between them is 90°

that is only true if $u^T v = 0$

orthonormal vectors: orthogonal vectors that each have unit length: $\text{norm}(u) = 1$

orthogonal matrix: all columns are orthonormal

if Q is orthogonal: $Q^T Q = I$

$$\Rightarrow Q^T = Q^{-1}$$

QR decomp: faster LLS

if H is full rank, then H can be written as:

$$H = QR \qquad H \text{ is } m \times n$$

where Q is $m \times n$ & orthogonal $(Q^T Q = I)$

R is upper right triangular $n \times n$

all zeros below the diagonal

$$H^T H = R^T Q^T Q R = R^T R$$

So LLS becomes: $\underline{\hat{x}} = R^{-1} Q^T \tilde{y}$

easy to do

QR decomp can be found by modified Gram-schmidt algorithm.

Singular Value Decomposition: more accurate LLS

$H = U S V^T$ $\quad$ H is $m \times n$

U is $m \times n$ w/ orthonormal cols

S is $n \times n$ diagonal

V is $n \times n$ orthogonal matrix

$$\underline{\hat{x}} = V S^{-1} U^T \underline{\tilde{y}}$$