4. Linear least-squares

- linear regression
- engineering applications
- solving linear least-squares
- numerical computation
- weighted linear least-squares
- properties of LS estimates

Linear regression

- a linear relationship between variables y and x_k using a linear function:

$$y = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n \triangleq x^T \beta$$

  where $y \in \mathbf{R}^m$, $x \in \mathbf{R}^{m \times n}$, $\beta \in \mathbf{R}^n$

- y contains the measurement variables and is often called the *regressed/response/explained/dependent variable*

- x_k's are the input variables that explain the behavior of y; called the *predictor/explanatory/independent variables*

- β is the *regression coefficient*

- given a data set: $\{(x_i, y_i)\}_{i=1}^m$ we can form a matrix form

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix} \quad \triangleq \quad y = X\beta$$

- the matrix X is sometimes called *the design/regressor matrix*

- given y and X, one would like to estimate β that gives the linear model output match best with y

- in practice, in the presence of noise and disturbance, more data should be collected in order to get a better estimate – leading to *overdetermined* linear equations

- an exact solution to $y = X\beta$ does not usually exist; however, it can be solved by **linear least-squares** formulation

Problem statement

overdetermined linear equations:

$$X\beta = y, \quad X \text{ is } m \times n \text{ with } m > n$$

for most y cannot solve for β

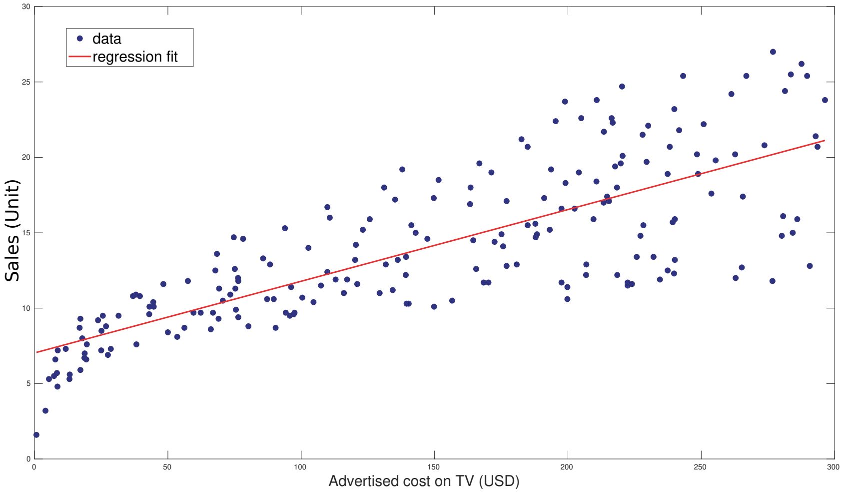
linear least-squares formulation:

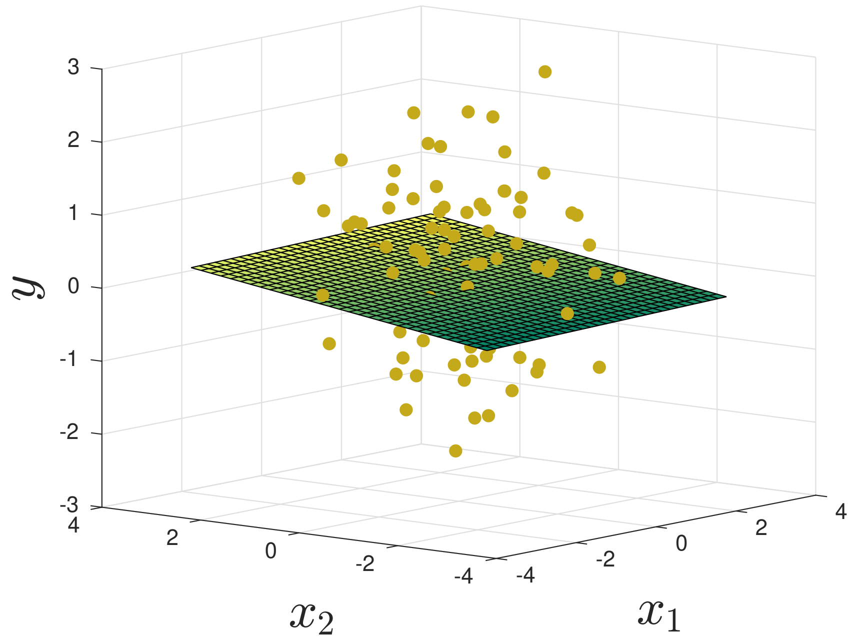
$$\underset{\beta}{\text{minimize}} \quad \|y - X\beta\|_2 = \left(\sum_{i=1}^{m} (\sum_{j=1}^{n} X_{ij}\beta_j - y_i)^2 \right)^{1/2}$$

- $r = y - X\beta$ is called *the residual error*
- β with smallest residual norm $\|r\|$ is called *the least-squares solution*
- equivalent to minimizing $\|y - X\beta\|^2$

Fitting linear least-squares

left: explain the sale amount by advertising on TV





- left: sum squared distance of data points to the line is minimum (this line fits best)
- right: for two predictors, LS solution is the normal vector of hyperplane that lies closest to all data points of y

Example 1: data fitting

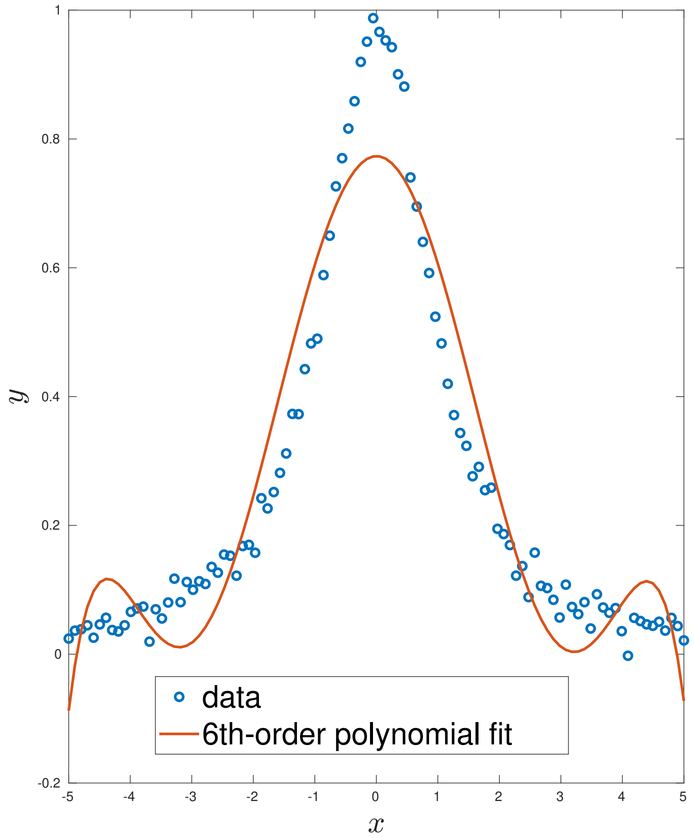
given data points $\{(t_i, y_i)\}_{i=1}^m$, we aim to approximate y using a function $g(t)$

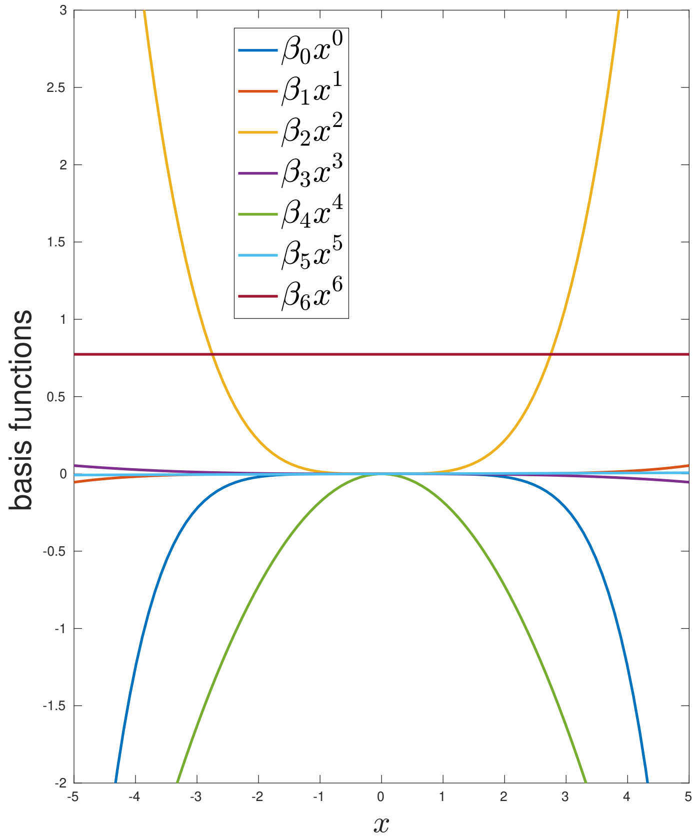
$$y = g(t) := \beta_1 g_1(t) + \beta_2 g_2(t) + \cdots + \beta_n g_n(t)$$

- $g_k(t) : \mathbf{R} \to \mathbf{R}$ is a basis function
  - polynomial functions: $1, t, t^2, \ldots, t^n$
  - sinusoidal functions: $\cos(\omega_k t), \sin(\omega_k t)$ for $k = 1, 2, \ldots, n$
- the linear regression model can be formulated as

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} g_1(t_1) & g_2(t_1) & \cdots & g_n(t_1) \\ g_1(t_2) & g_2(t_2) & \cdots & g_n(t_2) \\ \vdots & & & \vdots \\ g_1(t_m) & g_2(t_m) & \cdots & g_n(t_m) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix} \quad \triangleq \quad y = X\beta$$

- often have $m \gg n$, *i.e.*, explaining y using a few parameters in the model fitting a 6th-order polynomial to data points generated from $f(t) = 1/(1+t^2)$





- (right) the weighted sum of basis functions (x^k) is the fitted polynomial
- the ground-truth function f is nonlinear, but can be decomposed as a sum of polynomials

Example 2: FIR model

given input/output data: $\{(y(t), u(t))\}_{t=0}^{m}$, we aim to estimate FIR model parameters

$$y(t) = \sum_{k=0}^{n-1} h(k)u(t-k)$$

determine $h(0), h(1), \ldots, h(n-1)$ that gives FIR model output closest to y

$$\begin{bmatrix} y(n-1) \\ y(n) \\ \vdots \\ y(m) \end{bmatrix} = \begin{bmatrix} u(n-1) & u(n-2) & \ldots & u(0) \\ u(n) & u(n-1) & \ldots & u(1) \\ \vdots & \vdots & \vdots & \vdots \\ u(m) & u(m-1) & \ldots & u(m-n+1) \end{bmatrix} \begin{bmatrix} h(0) \\ h(1) \\ \vdots \\ h(n-1) \end{bmatrix}$$

- $y(t)$ is a response to $u(t), u(t-1), \ldots, u(t-(n-1))$
- we did not use initial outputs $y(0), y(1), \ldots, y(n-2)$ since there are no historical input data for those outputs

Example 3: scalar first-order model

given data set: $\{(u(t), y(t)\}_{t=1}^{N}$, we aim to estimate a scalar ARX model

$$y(t) = ay(t-1) + bu(t-1) + e(t)$$

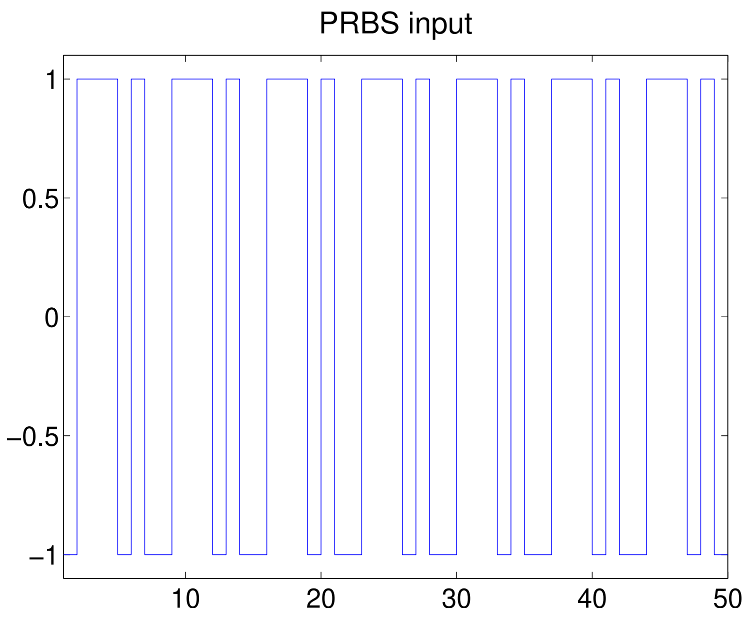
$y(t)$ is linear in model parameters: a, b

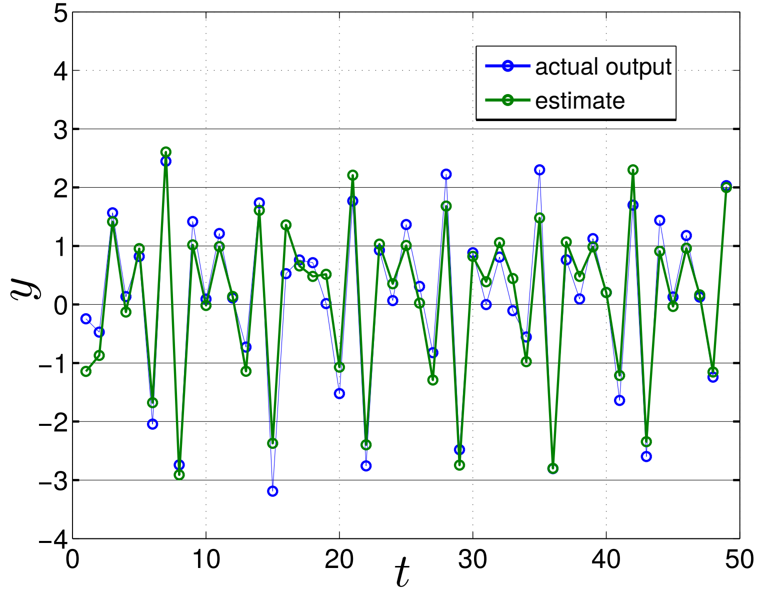
$$\begin{bmatrix} y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} y(1) & u(1) \\ y(2) & u(2) \\ \vdots & \vdots \\ y(N-1) & u(N-1) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

- the model is first-order, the equation is initialized with $y(1), u(1)$
- before collecting data, one chooses u to appropriately stimulate the system
- an impulse input is a bad choice as the whole second column is almost zero

data generation:

- $a = 0.8, b = 1$ are true parameters
- e is white noise with variance 0.1
- PRBS input





estimated parameters: $\hat{a} = 0.75, \hat{b} = 1.08$

Closed-form of least-squares estimate

the zero gradient condition of LS objective is

$$\frac{d}{d\beta}\|y - X\beta\|_2^2 = -X^T(y - X\beta) = 0$$

which is equivalent to the **normal equation**

$$X^T X\beta = X^T y$$

if X is **full rank**:

- least-squares solution can be found by solving the normal equations
- n equations in n variables with a positive definite coefficient matrix
- the closed-form solution is $\beta = (X^T X)^{-1} X^T y$
- $(X^T X)^{-1} X^T$ is a *left inverse* of X

Properties of full rank matrices

suppose X is an $m \times n$ matrix; we always have

$$\mathbf{rank}(X) \leq \text{min}(m, n)$$

if X is full rank with $m \geq n$ (tall matrix)

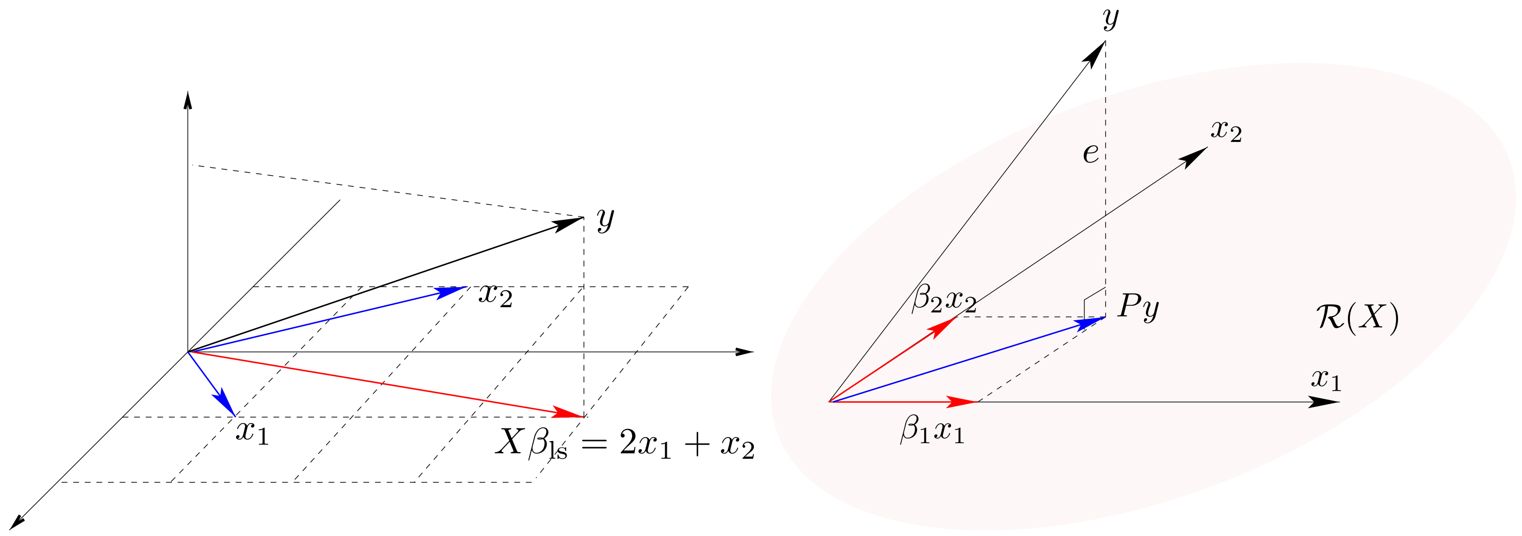
- $\mathbf{rank}(X) = n$ and $\mathcal{N}(X) = \{0\}$ ($Xz = 0 \Leftrightarrow z = 0$)
- $X^T X$ is positive definite: for any $z \neq 0$ then

$$z^T X^T X z = \|Xz\|^2 > 0$$

similarly, if X is full rank with $m \leq n$ (fat matrix)

- $\mathbf{rank}(X) = m$ and $\mathcal{N}(X^T) = \{0\}$
- XX^T is positive definite

Geometric interpretation of a LS problem

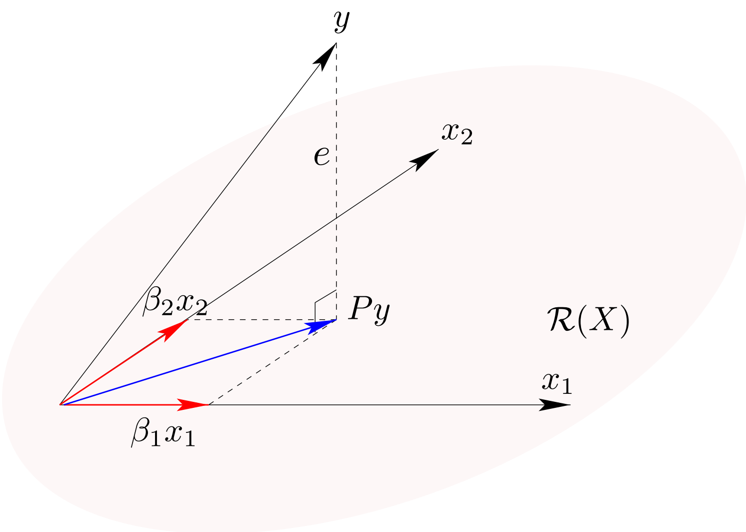


- $\|y - X\beta\|_2$ is the distance from y to

$$X\beta = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

- solution β_{ls} gives the linear combination of the columns of X closest to y
- $X\beta_{\mathrm{ls}}$ is the **orthogonal projection** of y to the range of X

Orthogonal projection



orthogonality condition

$$x_k^T(y - Py) = 0, \quad \forall k$$

the optimal residual $\perp$ to any vector in $\mathcal{R}(X)$

- Py is the orthogonal projection of y onto $\mathcal{R}(X)$ spanned by $x_1, \ldots, x_n$
- Py gives the best approximation; for any $\hat{y} \in \mathcal{R}(X)$ and $\hat{y} \neq Py$

$$\|y - Py\| < \|y - \hat{y}\|$$

- from the orthogonality condition and Py is a linear combination of $\{x_k\}$

$$x_k^T y = x_k^T P y = \sum_{j=1}^{n} x_k^T x_j \beta_j, \quad \forall k$$

$$\begin{bmatrix} x_1^T y \\ x_2^T y \\ \vdots \\ x_n^T y \end{bmatrix} = \begin{bmatrix} x_1^T x_1 & x_1^T x_2 & \ldots & x_1^T x_n \\ x_2^T x_1 & x_2^T x_2 & \ldots & x_2^T x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_n^T x_1 & x_n^T x_2 & \ldots & x_n^T x_n \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}$$

- this also leads to the **normal equation**: $X^T X x = X^T y$
- $X\beta_{\rm ls} = Py$ with

$$P = X(X^T X)^{-1} X^T$$

  (provided that X has **full rank**)

Numerical computation

we can solve a least-squares problem via

- Cholesky factorization: factor $X^TX \succ 0$ into LL^T where L is lower triangular
- QR factorization

most programming languages provide built-in commands

returned output	MATLAB	Python
$\hat{\beta}$	`X\y`	scipy.linalg.lstsq
estimated model	`fitlm`	sklearn.linear_model.LinearRegression

the closed-form $\hat{\beta} = (X^TX)^{-1}X^Ty$ is for analysis purpose

we do not actually compute $\hat{\beta}$ from this expression

Solving least-squares via QR factorization

for any tall $X \in \mathbf{R}^{m\times n}$, we have QR factorization:

$$X = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = Q_1 R_1$$

where $Q \in \mathbf{R}^{m\times m}$ orthonormal, $Q_1 \in \mathbf{R}^{m\times n}$, $R_1 \in \mathbf{R}^{n\times n}$ upper triangular, invertible

- multiplication by orthogonal matrix does not change the norm, so

$$\begin{aligned}
\|X\beta - y\|^2 &= \left\| \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} \beta - y \right\|^2 \\
&= \left\| \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}^T \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} \beta - \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} y \right\|^2 \\
&= \left\| \begin{bmatrix} R_1\beta - Q_1^T y \\ -Q_2^T y \end{bmatrix} \right\|^2 = \|R_1\beta - Q_1^T y\|^2 + \|Q_2^T y\|^2
\end{aligned}$$

- the least-squares objective can be minimized by the choice

$$\beta_{\mathrm{ls}} = R_1^{-1} Q_1^T y$$

  which makes the first term zero

- residual with optimal β is

$$X\beta_{\mathrm{ls}} - y = -Q_2 Q_2^T y$$

- $Q_1 Q_1^T$ gives projection on $\mathcal{R}(X)$

$$P = X(X^T X)^{-1} X^T = Q_1 R_1 (R_1^T R_1)^{-1} R_1^T Q_1^T = Q_1 Q_1^T$$

- $Q_2 Q_2^T$ gives projection on $\mathcal{R}(X)^\perp$

$$P^\perp = I - P = I - Q_1 Q_1^T = Q_2 Q_2^T$$

Weighted least-squares

given W a positive definite matrix that can be factorized as $W = L^T L$

a weighted least-squares (WLS) problem is

$$\underset{x}{\text{minimize}} \;\; (X\beta - y)^T W (X\beta - y)$$

- equivalent formulation: $\text{minimize}_x \;\; \|L(X\beta - y)\|^2$
- can be solved from the modified normal equation

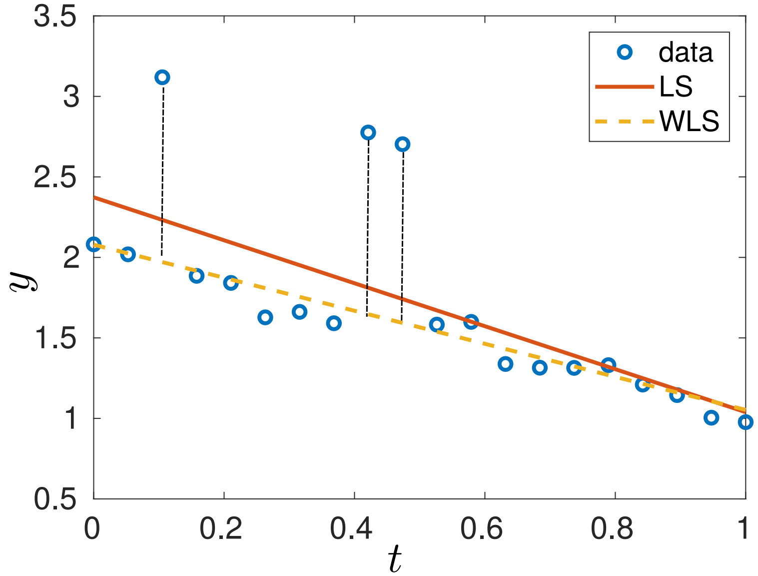
$$X^T W X \beta = X^T W y$$

- the solution is $\hat{\beta}_{\mathrm{wls}} = (X^T W X)^{-1} X^T W y$ (if X is full rank)
- $X\beta_{\mathrm{wls}}$ is the *orthogonal projection* on $\mathcal{R}(X)$ w.r.t the new inner product

$$\langle x, y \rangle_W = \langle Wx, y \rangle$$

Interpretation of WLS

when m-measurements contain some outliers (samples 3,9,10)



using $W = \mathbf{diag}(w_1, w_2, \ldots, w_m)$ gives WLS objective: $\sum_{i=1}^m w_i(y_i - x_i^T \beta)^2$

- use relatively **low** w_3, w_9, w_{10} to penalize **less** on those samples
- the linear model tends not to adapt to outliers – making WLS a more robust method than LS

Assumptions for analyzing LS estimates

a general regression model is of the form

$$y = \mathbf{E}[y|x] + e$$

- $\mathbf{E}[y|x]$ is the conditional mean of y when x is given (the best estimate in MMSE)
- e is uncertainty or noise; assumed to have zero mean
- generally, $\mathbf{E}[y|x]$ is nonlinear in x
- LS framework assumes that $\mathbf{E}[y|x]$ is linear in x

analysis of LS estimate relies on the data generating process (DGP)

$$y_i = x_i^T \beta + e_i, \quad i = 1, 2, \ldots, N$$

where β is the true (unknown) parameter; given $\{(x_i, y_i)\}_{i=1}^N$, we estimate β using LS framework

Analysis of the LS estimate (static case)

assumptions:

- e is noise with zero mean and covariance matrix Σ
- the least-square estimate: $\beta_{\text{ls}} = \operatorname{argmin} \|y - X\beta\|_2 = (X^T X)^{-1} X^T y$
- the sensor matrix X is *deterministic*

then the following properties hold:

- β_{ls} is an unbiased estimate of β ($\mathbf{E}\hat{\beta} = \beta$, or $\hat{\beta} = \beta$ when $e = 0$)
- the covariance matrix of β_{ls} is given by

$$\mathbf{cov}(\beta_{\text{ls}}) = (X^T X)^{-1} X^T \Sigma X (X^T X)^{-1}$$

the expression of $\mathbf{cov}(\beta_{\mathrm{ls}}) = (X^TX)^{-1}X^T\Sigma X(X^TX)^{-1}$ suggests that

- if X can be arbitrarily chosen, pick X that the covariance is small
- the covariance of the LS estimate depends on noise covariance

special case: noise covariance is diagonal

- $\Sigma = \mathbf{diag}(\sigma_1^2, \ldots, \sigma_N^2)$ (heteroskedasticity): e_i has different variances
- $\Sigma = \sigma^2 I$ (homoskedasticity): e_i has uniform variance

for homoskedasticity case, the covariance of the LS estimate reduces to

$$\mathbf{cov}(\beta_{\mathrm{ls}}) = \sigma^2(X^TX)^{-1}$$

note: X has N rows; as N increases, $(X^TX)^{-1}$ gets smaller making β_{ls} less uncertain

BLUE property

under the dgp: $y = X\beta + e$ and *homoskedasticity* of e, the LS estimator

$$\beta_{\mathrm{ls}} = (X^T X)^{-1} X^T y$$

is the **best linear unbiased estimator (BLUE)** of β

assume $\hat{\beta} = By$ is any other linear estimator of β

- require $BA = I$ in order for $\hat{\beta}$ to be unbiased
- $\mathbf{cov}(\hat{\beta}) = BB^T$
- $\mathbf{cov}(\beta_{\mathrm{ls}}) = BX(X^T X)^{-1} X^T B^T$ (apply $BX = I$)

for an orthogonal projection matrix, we have $I - P \succeq 0$

$$\mathbf{cov}(\hat{\beta}) - \mathbf{cov}(\beta_{\mathrm{ls}}) = B(I - X(X^T X)^{-1} X^T) B^T \succeq 0$$

β_{ls} has smaller covariance than other linear estimators

Generalized least-squares estimators

for correlated noise with $\mathbf{cov}(e) = \Sigma$

we can derive BLUE estimator by scaling $y = X\beta + e$ with $\Sigma^{-1/2}$

$$\Sigma^{-1/2}y = \Sigma^{-1/2}X\beta + \Sigma^{-1/2}e$$

the **generalized least-squares estimator** of β is

$$\beta_{\text{gls}} = (X^T\Sigma^{-1}X)^{-1}X^T\Sigma^{-1}y$$

which has BLUE property under heteroskedasticity of noise

- this is a special case of *weighted least-squares* problem with $W = \Sigma^{-1}$
- noise covariance is typically unknown; we replace Σ with its estimate

Analysis of the LS estimate (stochastic case)

suppose we apply the LS method to a dynamical system

$$y(t) = H(t)\beta + e(t)$$

- the observations $y(1), y(2), \ldots, y(N)$ are available
- β is the dynamical model parameter

typically, $x(t)$ contains the past outputs and inputs

$$y(1), \ldots, y(t-1), u(1), \ldots u(t-1)$$

(hence $x = H(t)$ is *no longer* deterministic)

and $e(t)$ is white noise with covariance Σ

the LS estimate $\hat{\beta}_N$ (depending on N) given by

$$\hat{\beta}_N = \left[\frac{1}{N}\sum_{t=1}^{N} H(t)^T H(t)\right]^{-1} \left[\frac{1}{N}\sum_{t=1}^{N} H(t)^T y(t)\right]$$

has the following properties (under some assumptions):

- $\hat{\beta}_N$ is consistent, *i.e.*, it converges to the true parameter in probability

$$\mathbf{plim}\,\hat{\beta}_N = \theta \quad \Longleftrightarrow \quad \lim_{N\to\infty} P(|\hat{\beta}_N - \beta| < \epsilon) = 1$$

- $\sqrt{N}(\hat{\beta} - \beta)$ is asymptotically Gaussian distributed $\mathcal{N}(0, P)$ where

$$P = \Sigma_x^{-1}\Sigma_{ux}\Sigma_x^{-1}$$

Σ_x involves $\mathbf{E}[H(t)^T H(t)]$ and Σ_{ux} involes $\mathbf{E}[H(t)e(t)e(t)^T H(t)^T]$

the consistency results of LS estimate are based on *some assumptions*

$$
\begin{aligned}
\hat{\beta}_N - \beta &= \left(\frac{1}{N} \sum_{t=1}^{N} H(t)^T H(t) \right)^{-1} \left\{ \frac{1}{N} \sum_{t=1}^{N} H(t)^T y(t) - \left(\frac{1}{N} \sum_{t=1}^{N} H(t)^T H(t) \right) \beta \right\} \\
&= \left(\frac{1}{N} \sum_{t=1}^{N} H(t)^T H(t) \right)^{-1} \left(\frac{1}{N} \sum_{t=1}^{N} H(t)^T e(t) \right)
\end{aligned}
$$

hence, $\hat{\beta}_N$ is consistent if

- $\mathbf{E}[H(t)^T H(t)]$ is nonsingular
  satisfied in most cases, except u is not persistently exciting of order n
- $\mathbf{E}[H(t)^T e(t)] = 0$
  not satisfied in most cases, except $e(t)$ is white noise

Summary

- LS method can be applied to models that are linear in the parameters
- a LS solution is unique if there is no colinearity (X is full rank)
- the method is mature, can be solve efficiently and is available in many softwares
- LS estimate has the BLUE property under the assumption that the noise in data generating process is homoskedastic
- LS estimate is consistent if the additive noise is uncorrelated with the regressors and the system is persistently excited

Related topics

- significance test: examine which predictors are significant to be included
- variable selection: best subset selection, step-wise regression
- qualitative input variables: use dummy variables
- some nonlinear relationship between y and x can be formulated as LS
- non-constant noise variance: some transformation of data, *e.g.*, $\log(\cdot)$ is applied
- regularization

References

L. Ljung, *System Identification: Theory for the User*, Prentice Hall, Second edition, 1999

Chapter 3 in

G.James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, Springer, 2013

Chapter 4 in
T. Söderström and P. Stoica, *System Identification*, Prentice Hall, 1989

Chapter 2-3 in
Linear least-squares and *The solution of a least-squares problem*, EE103, Lieven Vandenberghe, UCLA, `http://www.ee.ucla.edu/~vandenbe/ee103.html`