Optimalizace

Použití lineární úlohy nejmenších čtverců (a podobných)

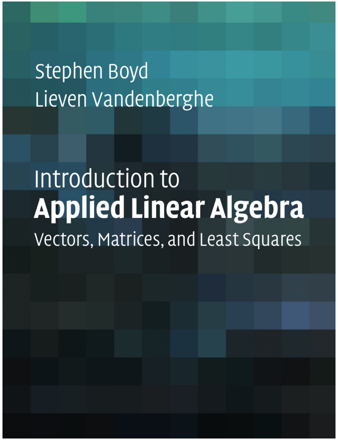
Tomáš Werner

FEL ČVUT

Mnoho aplikací úlohy

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$

je v knize (zdarma ke stažení i se slajdy):



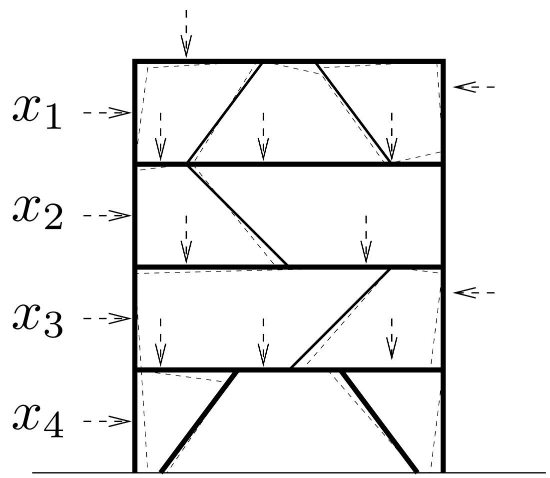
(Slides in this lecture are compiled from various courses taught by S.Boyd and L.Vanderberghe.)

Interpretations of $y = Ax$

- y is measurement or observation; x is unknown to be determined
- x is 'input' or 'action'; y is 'output' or 'result'
- $y = Ax$ defines a function or transformation that maps $x \in \mathbf{R}^n$ into $y \in \mathbf{R}^m$

Linear elastic structure

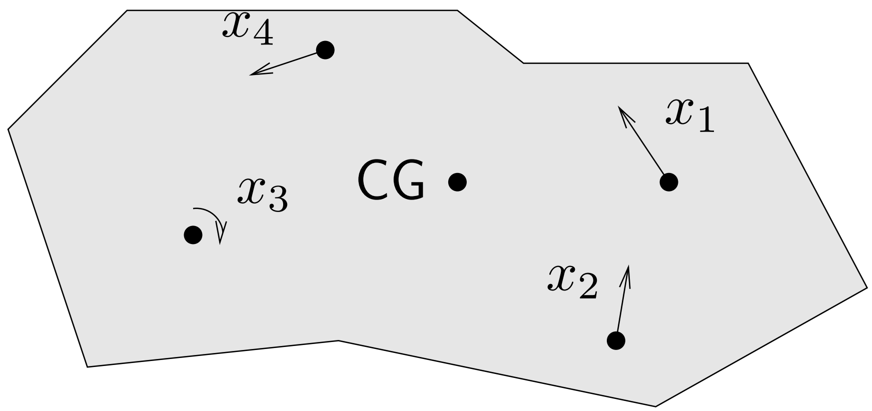
- x_j is external force applied at some node, in some fixed direction
- y_i is (small) deflection of some node, in some fixed direction



(provided x, y are small) we have $y \approx Ax$

- A is called the *compliance matrix*
- a_{ij} gives deflection i per unit force at j (in m/N)

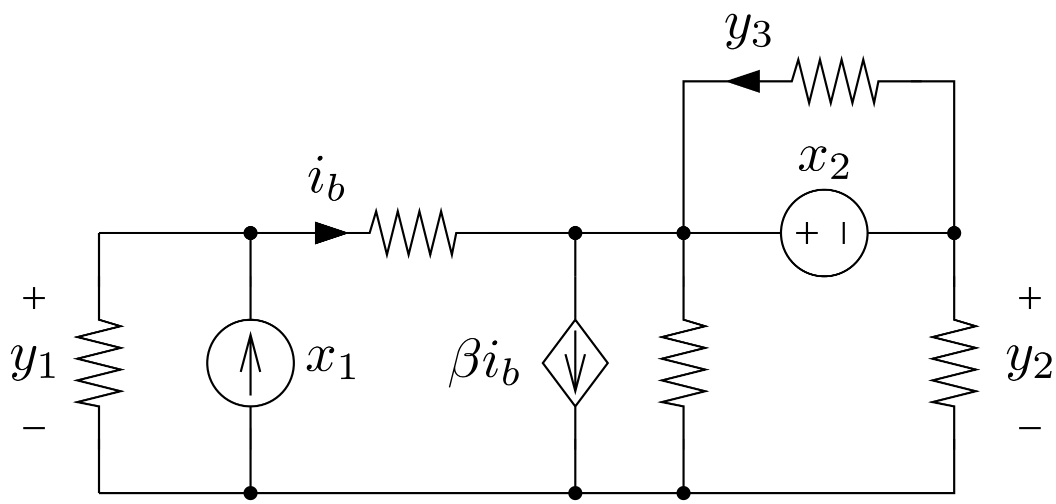
Total force/torque on rigid body



- x_j is external force/torque applied at some point/direction/axis
- $y \in \mathbf{R}^6$ is resulting total force & torque on body
  ($y_1,\ y_2,\ y_3$ are $\mathbf{x}$-, $\mathbf{y}$-, $\mathbf{z}$- components of total force,
  $y_4,\ y_5,\ y_6$ are $\mathbf{x}$-, $\mathbf{y}$-, $\mathbf{z}$- components of total torque)
- we have $y = Ax$
- A depends on geometry
  (of applied forces and torques with respect to center of gravity CG)
- jth column gives resulting force & torque for unit force/torque j

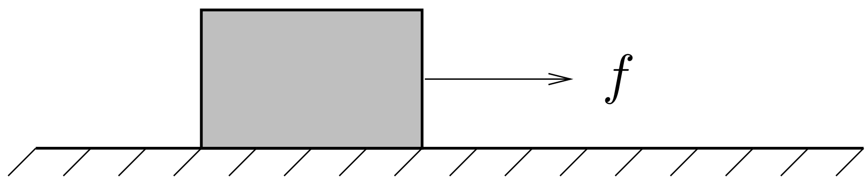
Linear static circuit

interconnection of resistors, linear dependent (controlled) sources, and independent sources



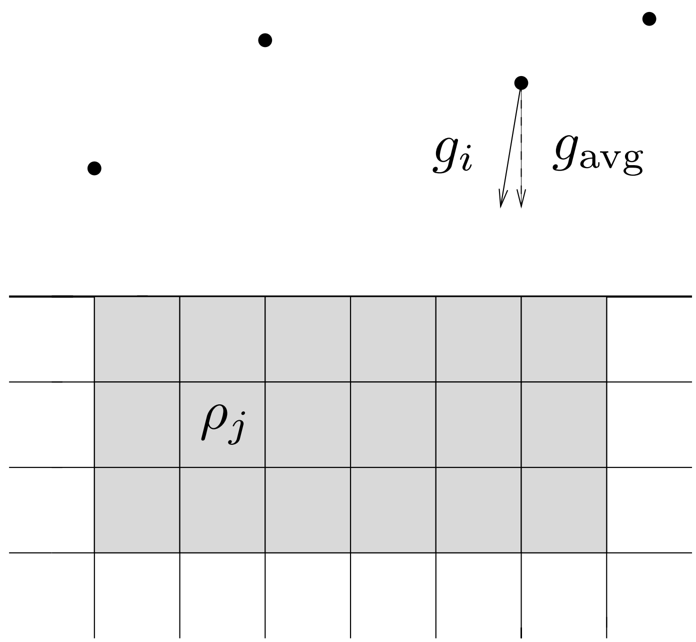
- x_j is value of independent source j
- y_i is some circuit variable (voltage, current)
- we have $y = Ax$
- if x_j are currents and y_i are voltages, A is called the *impedance* or *resistance* matrix

Final position/velocity of mass due to applied forces



- unit mass, zero position/velocity at $t = 0$, subject to force $f(t)$ for $0 \leq t \leq n$
- $f(t) = x_j$ for $j - 1 \leq t < j$, $j = 1, \ldots, n$
  (x is the sequence of applied forces, constant in each interval)
- y_1, y_2 are final position and velocity ($i.e.$, at $t = n$)
- we have $y = Ax$
- a_{1j} gives influence of applied force during $j - 1 \leq t < j$ on final position
- a_{2j} gives influence of applied force during $j - 1 \leq t < j$ on final velocity

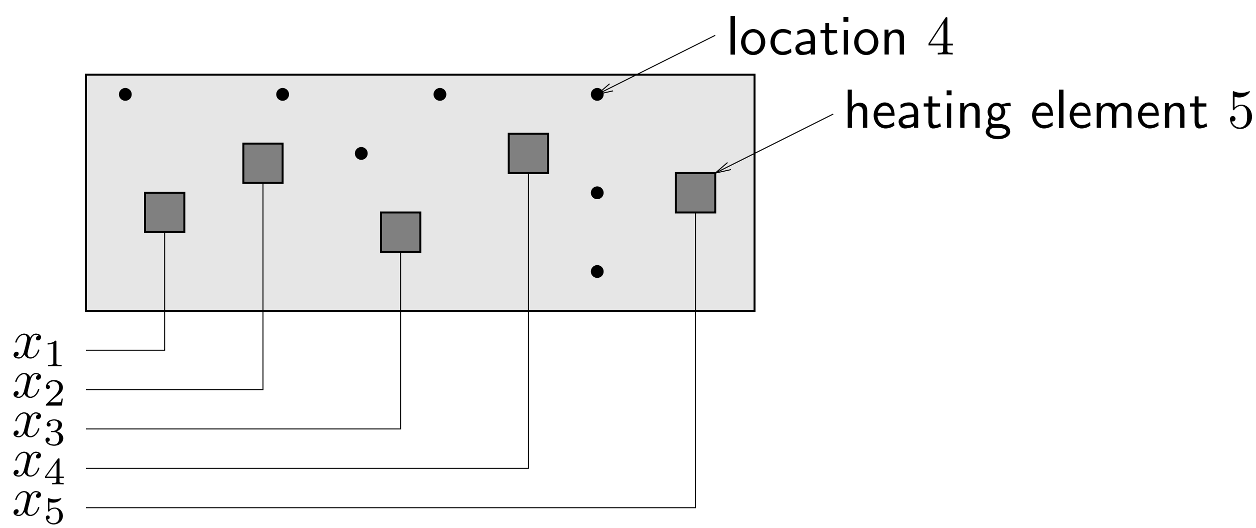
Gravimeter prospecting



- $x_j = \rho_j - \rho_{\text{avg}}$ is (excess) mass density of earth in voxel j;
- y_i is measured *gravity anomaly* at location i, $i.e.$, some component (typically vertical) of $g_i - g_{\text{avg}}$
- $y = Ax$

- A comes from physics and geometry
- jth column of A shows sensor readings caused by unit density anomaly at voxel j
- ith row of A shows sensitivity pattern of sensor i

Thermal system



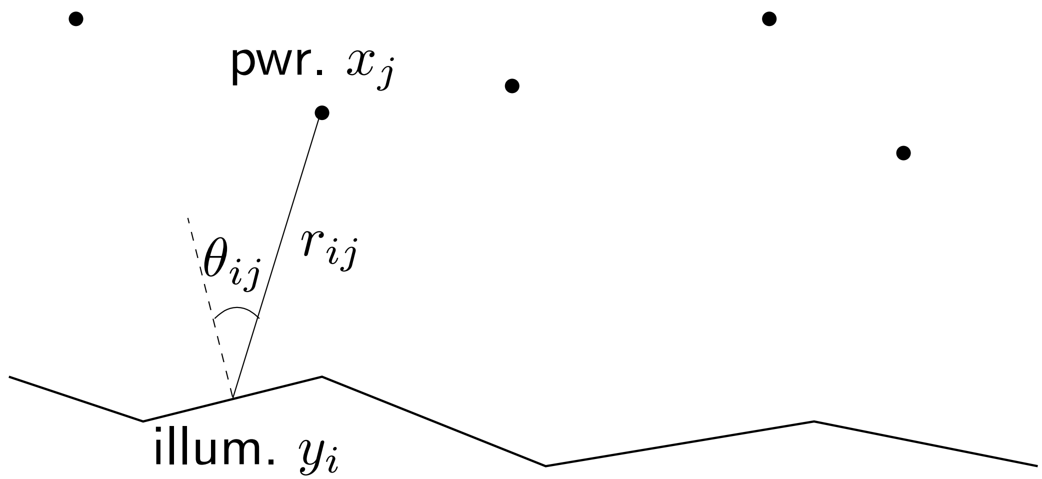
- x_j is power of jth heating element or heat source
- y_i is change in steady-state temperature at location i
- thermal transport via conduction
- $y = Ax$

- a_{ij} gives influence of heater j at location i (in $^\circ$C/W)

- jth column of A gives pattern of steady-state temperature rise due to 1W at heater j

- ith row shows how heaters affect location i

Illumination with multiple lamps



- n lamps illuminating m (small, flat) patches, no shadows
- x_j is power of jth lamp; y_i is illumination level of patch i
- $y = Ax$, where $a_{ij} = r_{ij}^{-2} \max\{\cos\theta_{ij}, 0\}$

  ($\cos\theta_{ij} < 0$ means patch i is shaded from lamp j)
- jth column of A shows illumination pattern from lamp j

Broad categories of applications

linear model or function $y = Ax$

some broad categories of applications:

- estimation or inversion
- control or design
- mapping or transformation

(this list is not exclusive; can have combinations . . .)

Estimation or inversion

$$y = Ax$$

- y_i is ith measurement or sensor reading (which we know)
- x_j is jth parameter to be estimated or determined
- a_{ij} is sensitivity of ith sensor to jth parameter

sample problems:

- find x, given y
- find all x's that result in y ($i.e.$, all x's consistent with measurements)
- if there is no x such that $y = Ax$, find x s.t. $y \approx Ax$ ($i.e.$, if the sensor readings are inconsistent, find x which is almost consistent)

Control or design

$$y = Ax$$

- x is vector of design parameters or inputs (which we can choose)
- y is vector of results, or outcomes
- A describes how input choices affect results

sample problems:

- find x so that $y = y_{\mathrm{des}}$
- find all x's that result in $y = y_{\mathrm{des}}$ (*i.e.*, find all designs that meet specifications)
- among x's that satisfy $y = y_{\mathrm{des}}$, find a small one (*i.e.*, find a small or efficient x that meets specifications)

Mapping or transformation

- x is mapped or transformed to y by linear function $y = Ax$

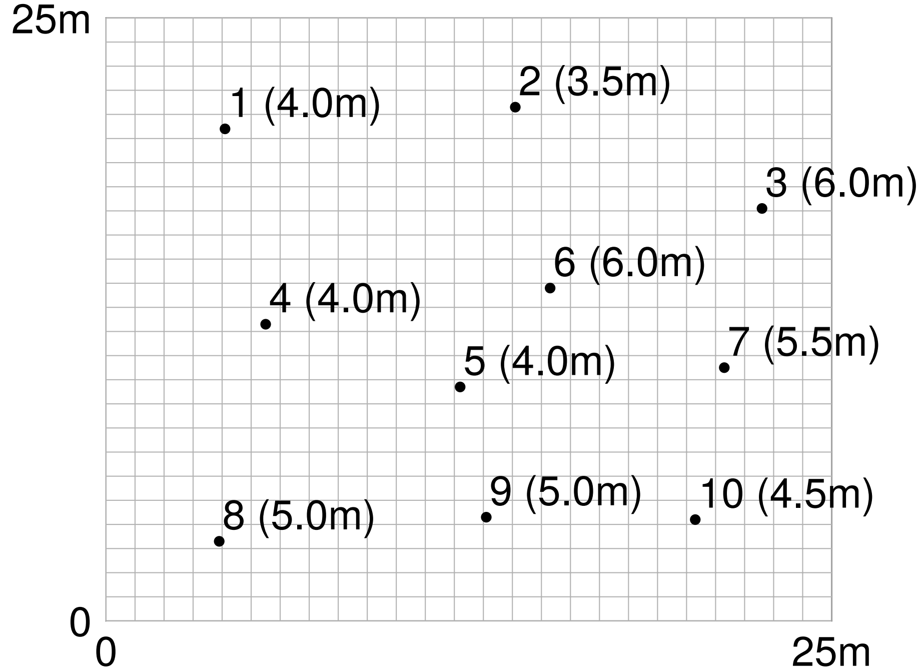
sample problems:

- determine if there is an x that maps to a given y
- (if possible) find *an* x that maps to y
- find *all* x's that map to a given y
- if there is only one x that maps to y, find it ($i.e.$, decode or undo the mapping)

Example: illumination

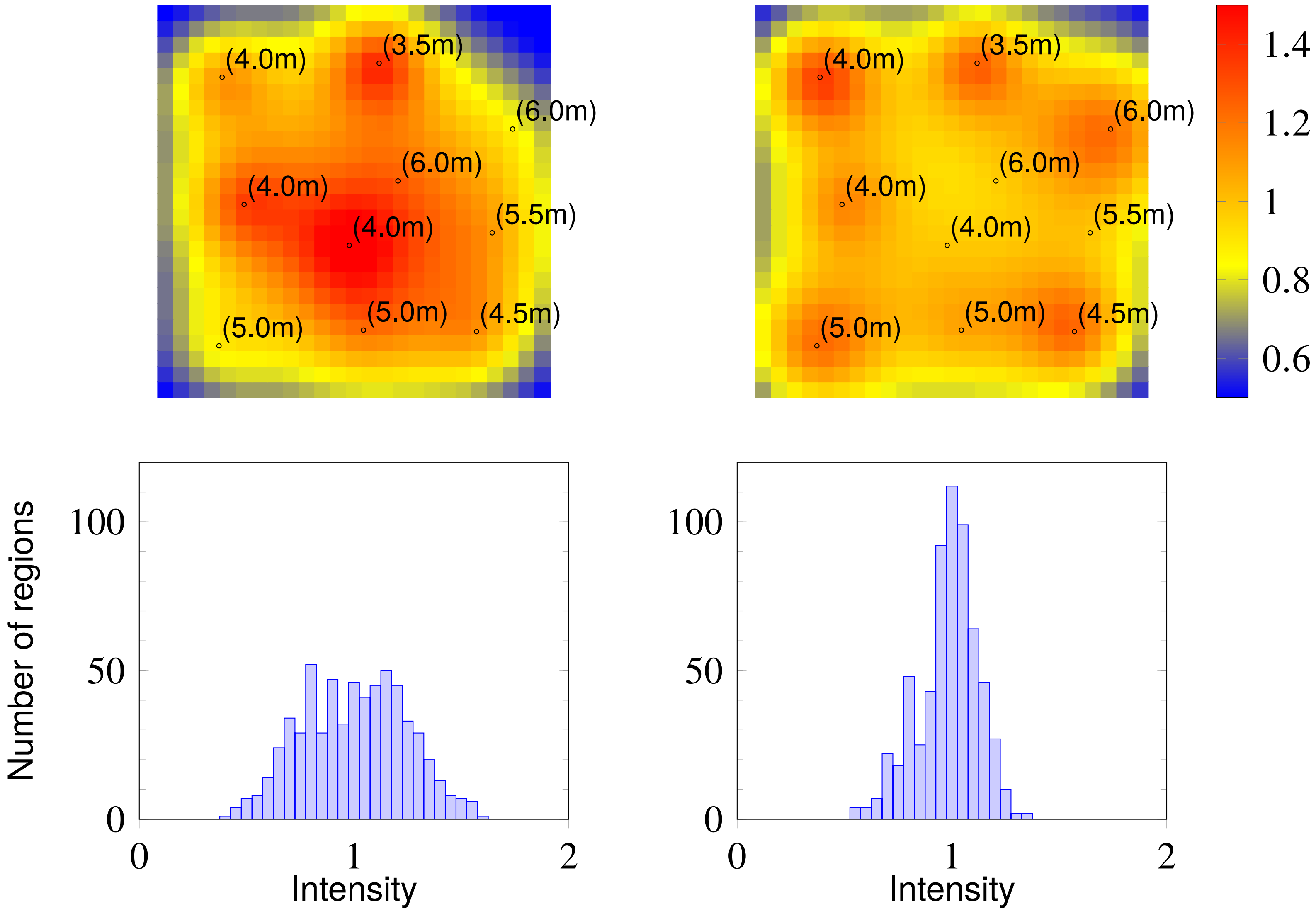
- n lamps at given positions above an area divided in m regions
- A_{ij} is illumination in region i if lamp j is on with power 1 and other lamps are off
- x_j is power of lamp j
- $(Ax)_i$ is illumination level at region i
- b_i is target illumination level at region i

Example: $m = 25^2$, $n = 10$; figure shows position and height of each lamp



Example: illumination

- left: illumination pattern for equal lamp powers ($x = \mathbf{1}$)
- right: illumination pattern for least squares solution $\hat{x}$, with $b = \mathbf{1}$



Linear-in-parameters model

we choose the model $\hat{f}(x)$ from a family of models

$$\hat{f}(x) = \theta_1 f_1(x) + \theta_2 f_2(x) + \cdots + \theta_p f_p(x)$$

- the functions f_i are scalar valued *basis functions* (chosen by us)
- the basis functions often include a constant function (typically, $f_1(x) = 1$)
- the coefficients $\theta_1, \ldots, \theta_p$ are the model *parameters*
- the model $\hat{f}(x)$ is linear in the parameters θ_i
- if $f_1(x) = 1$, this can be interpreted as a regression model

$$\hat{y} = \beta^T \tilde{x} + v$$

with parameters $v = \theta_1$, $\beta = \theta_{2:p}$ and new features $\tilde{x}$ generated from x:

$$\tilde{x}_1 = f_2(x), \quad \ldots, \quad \tilde{x}_p = f_p(x)$$

Least squares model fitting

- fit linear-in-parameters model to data set $(x^{(1)}, y^{(1)}), \ldots, (x^{(N)}, y^{(N)})$
- residual for data sample i is

$$r^{(i)} = y^{(i)} - \hat{f}(x^{(i)}) = y^{(i)} - \theta_1 f_1(x^{(i)}) - \cdots - \theta_p f_p(x^{(i)})$$

- least squares model fitting: choose parameters θ by minimizing MSE

$$\frac{1}{N}\left((r^{(1)})^2 + (r^{(2)})^2 + \cdots + (r^{(N)})^2\right)$$

- this is a least squares problem: minimize $\|A\theta - y^{\mathrm{d}}\|^2$ with

$$A = \begin{bmatrix} f_1(x^{(1)}) & \cdots & f_p(x^{(1)}) \\ f_1(x^{(2)}) & \cdots & f_p(x^{(2)}) \\ \vdots & & \vdots \\ f_1(x^{(N)}) & \cdots & f_p(x^{(N)}) \end{bmatrix}, \qquad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_p \end{bmatrix}, \qquad y^{\mathrm{d}} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(N)} \end{bmatrix}$$

Example: polynomial approximation

$$\hat{f}(x) = \theta_1 + \theta_2 x + \theta_3 x^2 + \cdots + \theta_p x^{p-1}$$

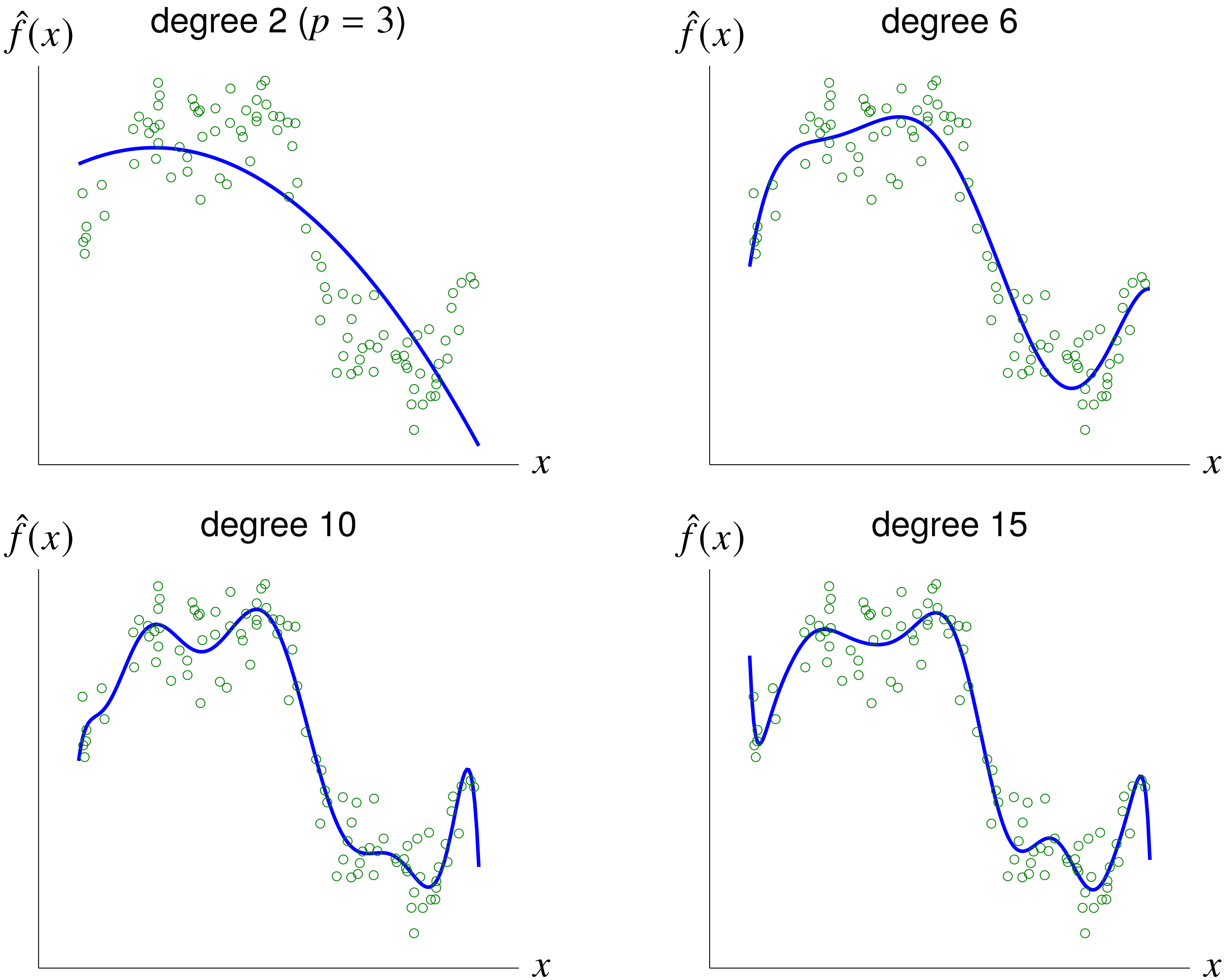
- a linear-in-parameters model with basis functions $1, x, \ldots, x^{p-1}$
- least squares model fitting: choose parameters θ by minimizing MSE

$$\frac{1}{N}\left((y^{(1)} - \hat{f}(x^{(1)}))^2 + (y^{(2)} - \hat{f}(x^{(2)}))^2 + \cdots + (y^{(N)} - \hat{f}(x^{(N)}))^2\right)$$

- in matrix notation: minimize $\|A\theta - y^{\mathrm{d}}\|^2$ with

$$A = \left[\begin{array}{ccccc} 1 & x^{(1)} & (x^{(1)})^2 & \cdots & (x^{(1)})^{p-1} \\ 1 & x^{(2)} & (x^{(2)})^2 & \cdots & (x^{(2)})^{p-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x^{(N)} & (x^{(N)})^2 & \cdots & (x^{(N)})^{p-1} \end{array}\right], \qquad y^{\mathrm{d}} = \left[\begin{array}{c} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(N)} \end{array}\right]$$

Example



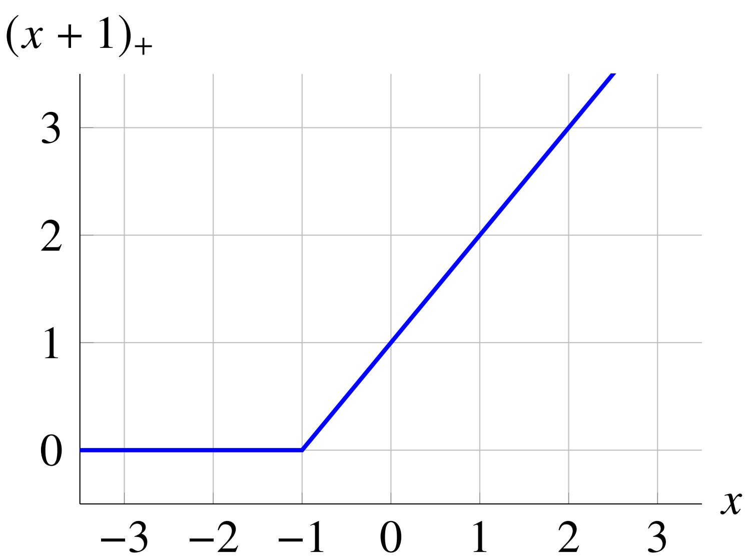
data set of 100 examples

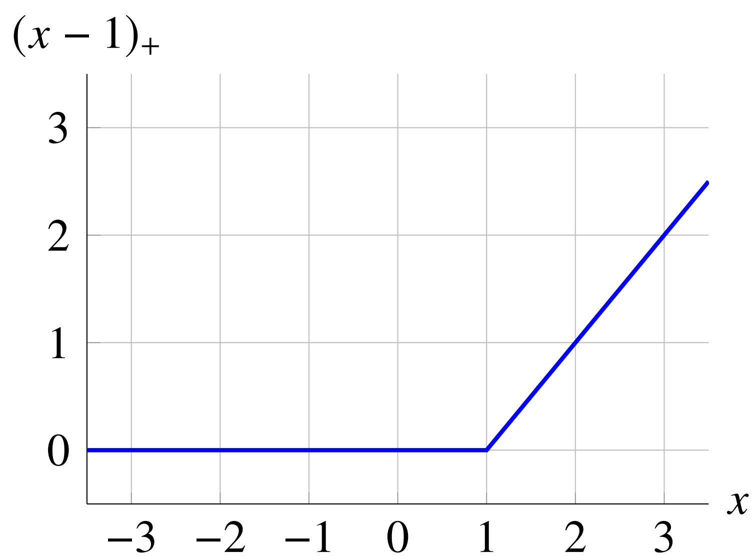
Piecewise-affine function

- define *knot points* $a_1 < a_2 < \cdots < a_k$ on the real axis
- piecewise-affine function is continuous, and affine on each interval $[a_k, a_{k+1}]$
- piecewise-affine function with knot points $a_1, \ldots, a_k$ can be written as

$$\hat{f}(x) = \theta_1 + \theta_2 x + \theta_3 (x - a_1)_+ + \cdots + \theta_{2+k}(x - a_k)_+$$

where $u_+ = \max\{u, 0\}$



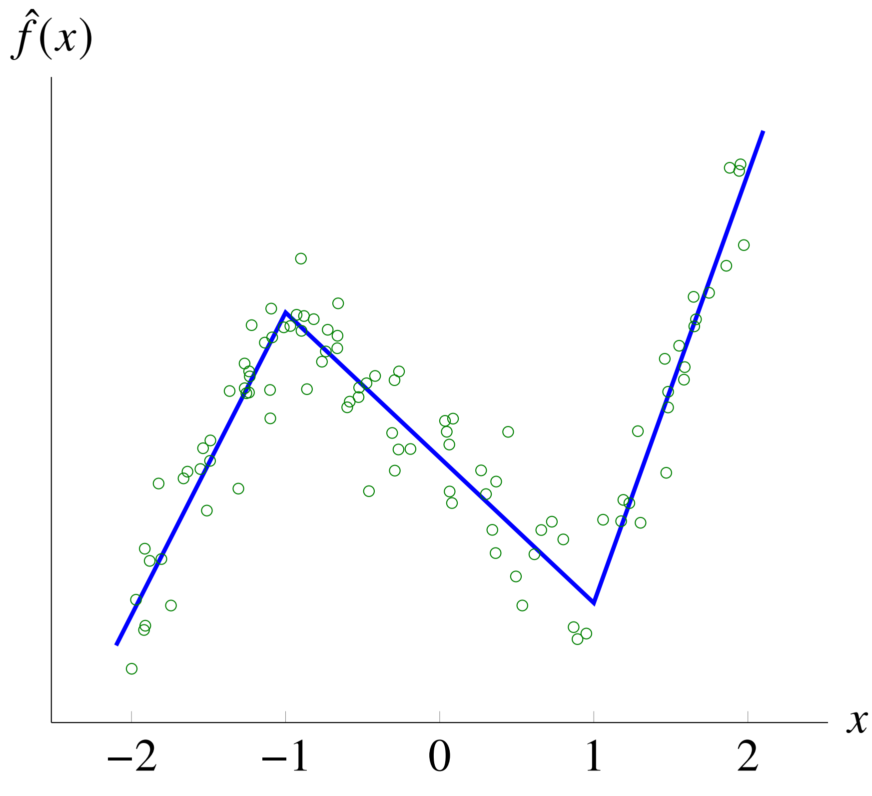


Piecewise-affine function fitting

piecewise-affine model is in linear in the parameters θ, with basis functions

$$f_1(x) = 1, \quad f_2(x) = x, \quad f_3(x) = (x - a_1)_+, \quad \ldots, \quad f_{k+2}(x) = (x - a_k)_+$$

Example: fit piecewise-affine function with knots $a_1 = -1$, $a_2 = 1$ to 100 points



Auto-regressive (AR) time series model

$$\hat{z}_{t+1} = \beta_1 z_t + \cdots + \beta_M z_{t-M+1}, \qquad t = M, M+1, \ldots$$

- $z_1, z_2, \ldots$ is a time series
- $\hat{z}_{t+1}$ is a prediction of z_{t+1}, made at time t
- prediction $\hat{z}_{t+1}$ is a linear function of previous M values $z_t, \ldots, z_{t-M+1}$
- M is the *memory* of the model

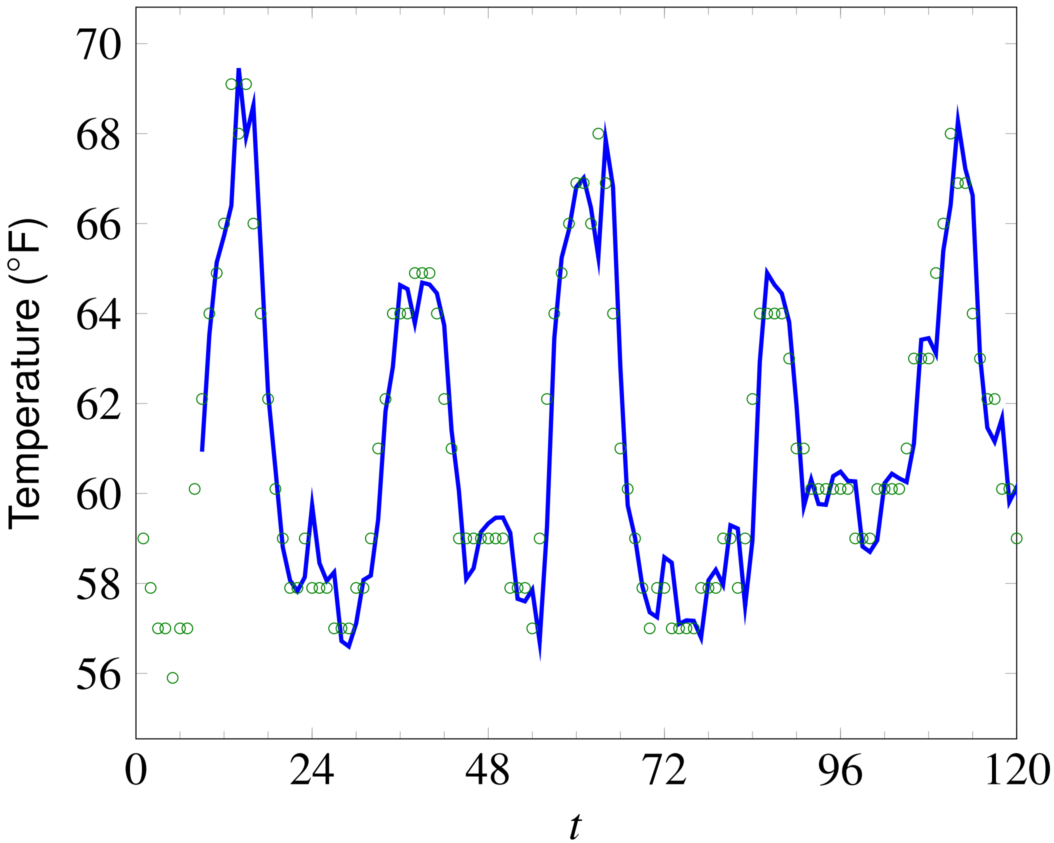
Least squares fitting of AR model: given oberved data $z_1, \ldots, z_T$, minimize

$$(z_{M+1} - \hat{z}_{M+1})^2 + (z_{M+2} - \hat{z}_{M+2})^2 + \cdots + (z_T - \hat{z}_T)^2$$

this is a least squares problem: minimize $\|A\beta - y^{\mathrm{d}}\|^2$ with

$$A = \begin{bmatrix} z_M & z_{M-1} & \cdots & z_1 \\ z_{M+1} & z_M & \cdots & z_2 \\ \vdots & \vdots & & \vdots \\ z_{T-1} & z_{T-2} & \cdots & z_{T-M} \end{bmatrix}, \qquad \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix}, \qquad y^{\mathrm{d}} = \begin{bmatrix} z_{M+1} \\ z_{M+2} \\ \vdots \\ z_T \end{bmatrix}$$

Example: hourly temperature at LAX



- blue line shows prediction by AR model of memory $M = 8$
- model was fit on time series of length $T = 744$ (May 1–31, 2016)
- plot shows first five days

Generalization and validation

Generalization ability: ability of model to predict outcomes for new, unseen data

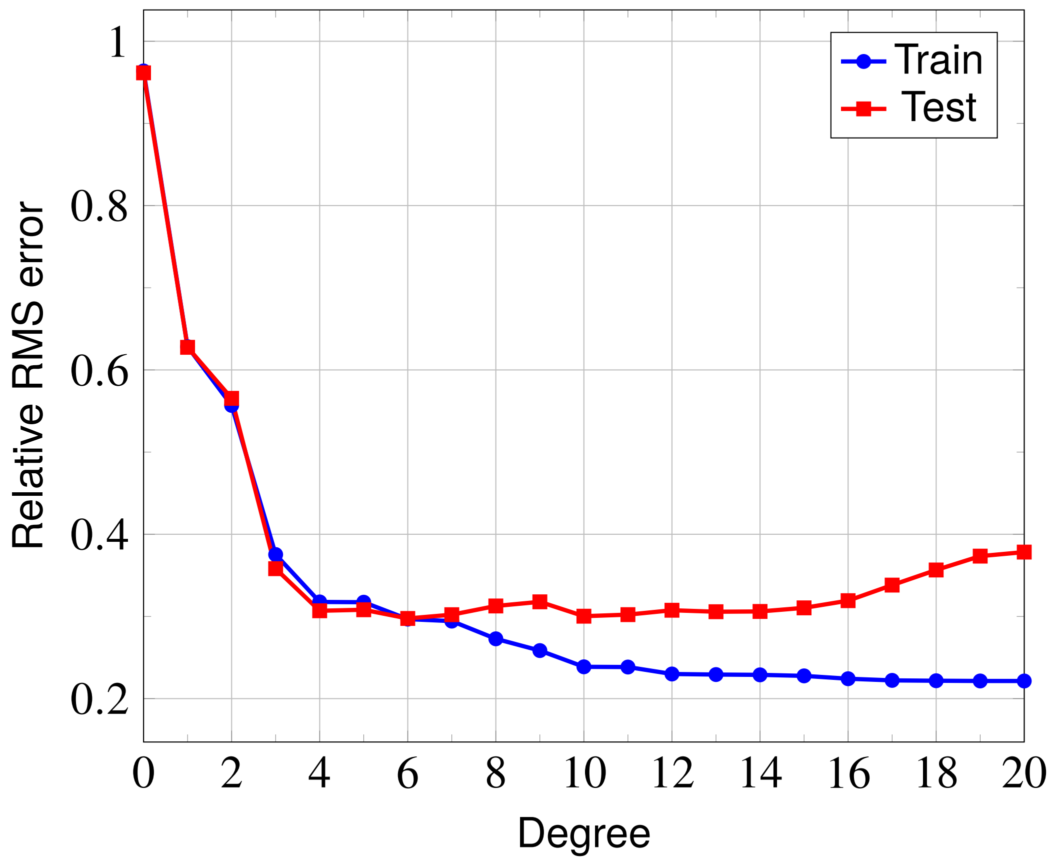
Model validation: to assess generalization ability,

- divide data in two sets: *training set* and *test (or validation)* set
- use training set to fit model
- use test set to get an idea of generalization ability
- this is also called *out-of-sample validation*

Over-fit model

- model with low prediction error on training set, bad generalization ability
- prediction error on training set is much smaller than on test set

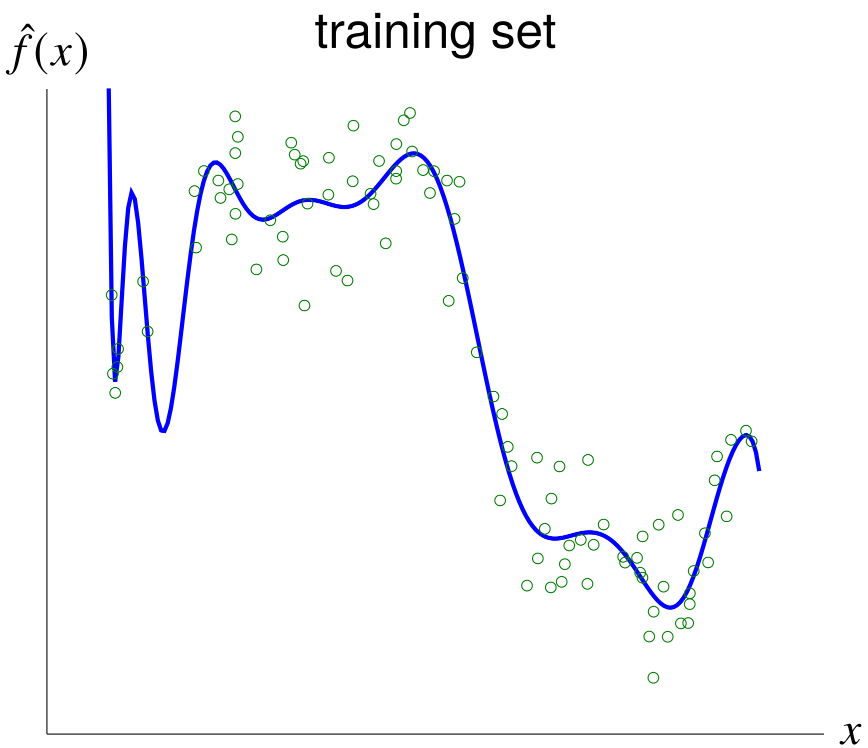
Example: polynomial fitting

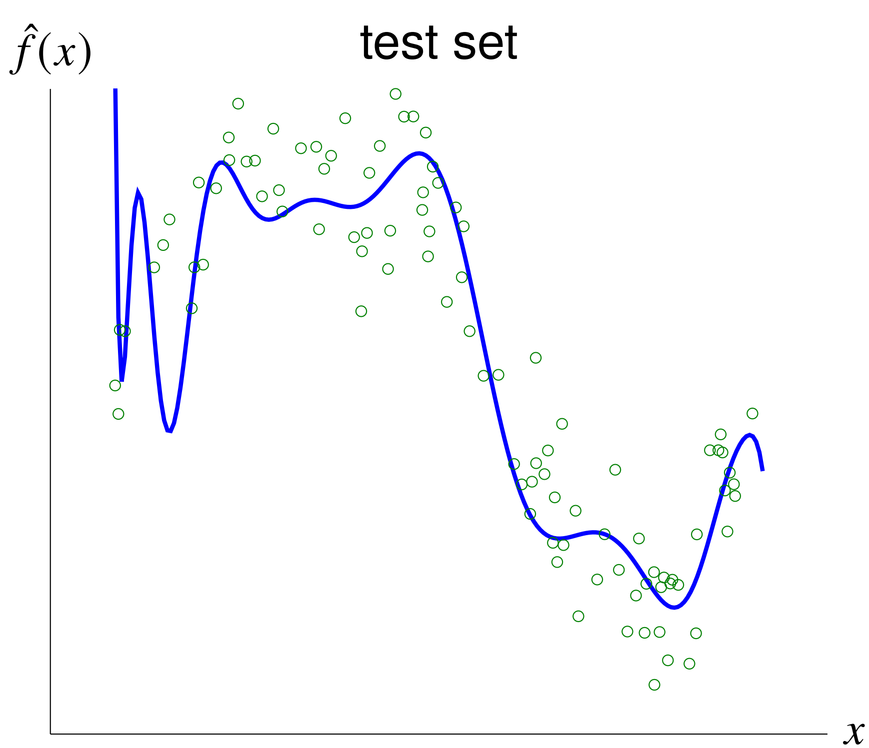


- training set is data set of 100 points used on page 9.11
- test set is a similar set of 100 points
- plot suggests using degree 6

Over-fitting

polynomial of degree 20 on training and test set





over-fitting is evident at the left end of the interval