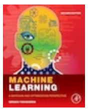
UNIVERSIDADE DO ALGARVE
FACULDADE DE CIÊNCIAS E TECNOLOGIA

Inductive Learning

Readings: (:chapt. 3)

1

Pure inductive learning

- Let *f* be an *unknown* target function

Problem: find a hypothesis *h*
such that $h \approx f$
given a training set of examples

An example is a pair (x, $f(x)$)

2

Pure inductive learning

- The previous problem is an *ill-defined problem*

  In general, data are not enough to identiy an unique hypotesis.

3

Pure inductive learning

- A binary function of two binary variables (x1, x2)

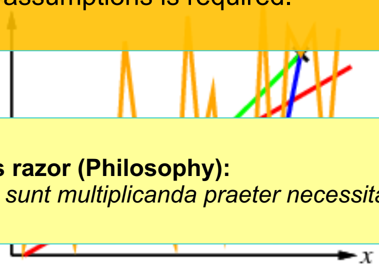
x_1	x_2	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8	h_9	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}	h_{15}	h_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

4

Pure inductive learning

- How to choose among diferent hypothesis (models)?

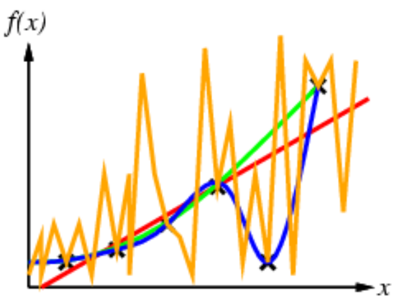
As the problem is ill-defined, an *inductive bias*, i.e., a set of à priori assumptions is required.



Ockham's razor (Philosophy):
"entia non sunt multiplicanda praeter necessitatem"

Ockham's razor – Why?

- Advantages
  - There are less simple hypothesis than complex ones.
  - highly probable that a sufficiently complex hypothesis will fit the data
  - An simple hypothesis that fits data is less probable to be a statistical coincidence.



Inductive learning hypothesis

- Inductive learning hypothesis: Any hypothesis found to approximate the target function well over a sufficient large set of training examples will also approximate the target function well over other unobserved examples.

Inductive Learning

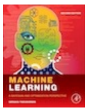
- Generalization: What's the model's performance on new data?
- Overfitting: $\mathcal{H}$ is more complex than f
- Underfitting: $\mathcal{H}$ is less complex than f

To take away

- Inductive learning
  - Pure vs. biased inductive learning
  - Ockham's razor
  - Inductive Learning hypothesis

9

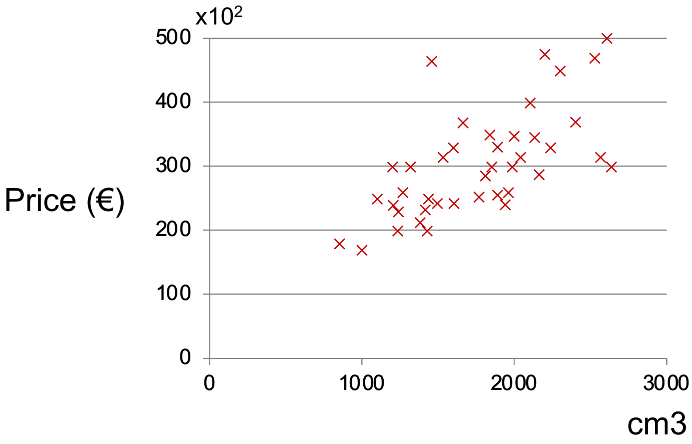
Learning in parametric modeling (Simple linear regression)

Readings: (

:chapt. 3)

10

Working example:
Car's price as a function of engine volume



11

Some notation

Cm3 (x)	Price in € (y)
2104	460
852	178
…	…

D={($x^{(i)}$, $y^{(i)}$) | i=1,…, m} – Data set

m - Number of examples (elements) in the data set
$x^{(i)}$ - input variable (feature) of the i-th example
$y^{(i)}$ – desired output variable of the i-th example

12

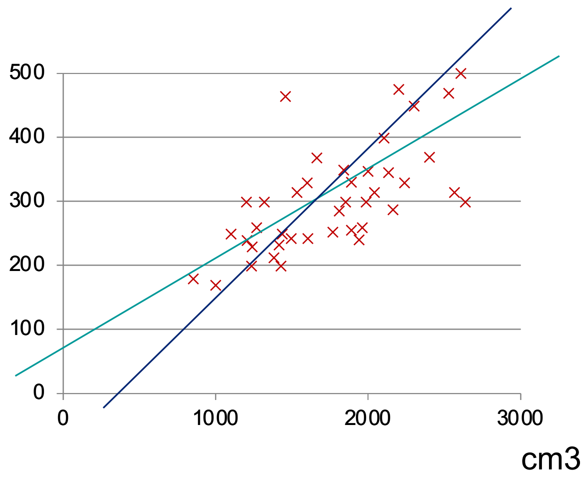
Hypothesis

$$h_\theta(x) = \theta_0 + \theta_1 x$$

where θ_0, θ_1 are real-valued coefficients or weights

13

Working example:
Car's price as a function of its engine



14

Performance function (or loss function)

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \Big(\underbrace{h_\theta(x^{(i)}) - y^{(i)}}_{\text{resídual}} \Big)^2$$

Objective: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

19-Sep-24 http://tiny.cc/ml2024-25 12 – 15

15

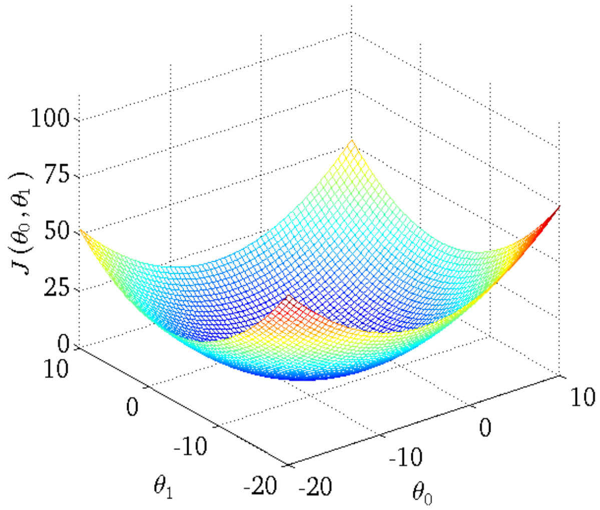
Y-intercept and sample slope

- Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x$
- Y-Intercept: $\theta_0 = \bar{y} - \theta_1 \bar{x}$
- Sample' slope $\theta_1 = \frac{SS_{xy}}{SS_{xx}} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$

19-Sep-24 http://tiny.cc/ml2024-25 12 – 16

16

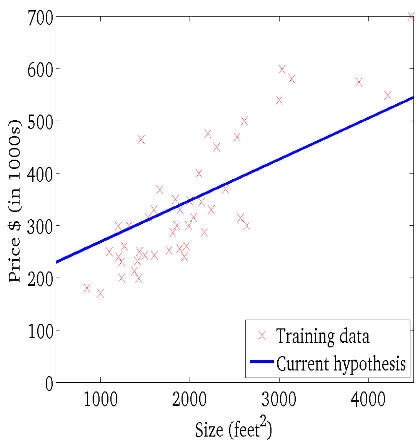
J as function of θ_0 and θ_1



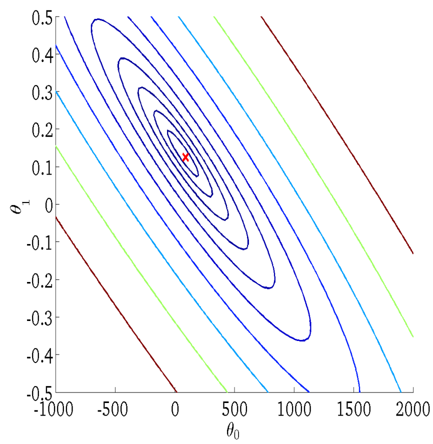
17

Contour of $J(\theta_0, \theta_1)$

$h_\theta(x)$

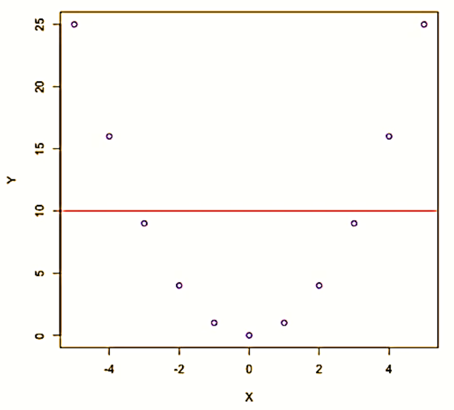


$J(\theta_0, \theta_1)$

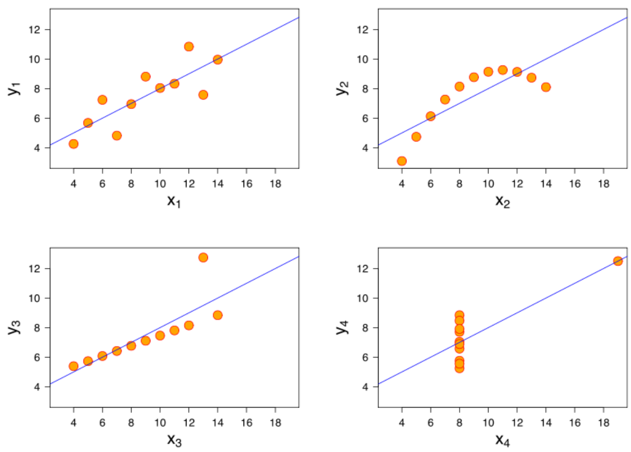


18

Linear regression of a quadratic function



F. J. Anscombe's quartet (1973)

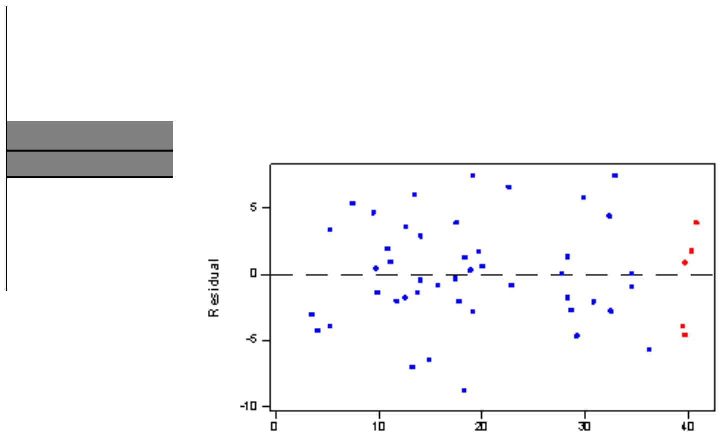


Residuals

- Residuals are the difference between the observed (y) and predicted responses (h).
- For the normal error regression model, we assume that the error term is normally distributed with null mean and constant variance.
- If the model is appropriate for the data, this should be reflected in the residuals. In particular, the residual vs. input graphic should be centered in zero and show no tendency.

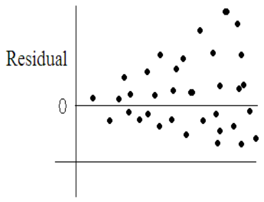
21

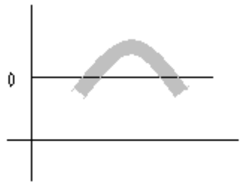
Residuals, the pattern we would like to see

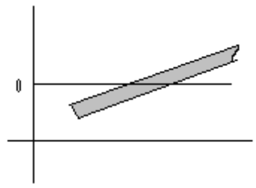


22

Residuals with some kind of anomalies







To take away

- Inductive learning
  - Pure vs. biased inductive learning
  - Ockham's razor
  - Inductive Learning hypothesis
- Simple Linear regression
  - Notation
  - Performance or total loss function
  - Y-Intercept, slope, and residuals
  - Limitations

Recalling the gradient method

25

Problem formulation

- Find the x* such that minimizes

$$f : R^n \to R$$

where f is a smooth C^1 function

26

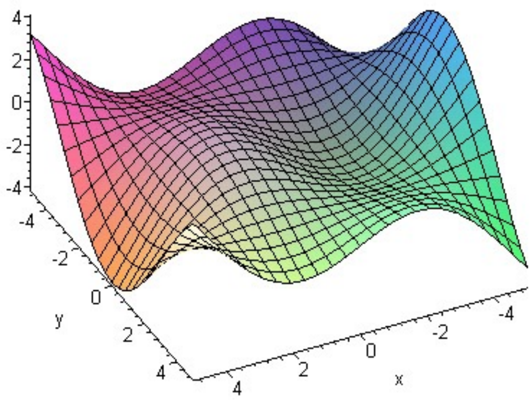
The gradient

$$f : R^n \to R$$

$$\nabla f(x_1,\dots,x_n) := \left(\frac{\partial f}{\partial x_1},\dots, \quad \frac{\partial f}{\partial x_n} \right)^{\mathrm{T}}$$

27

Application example



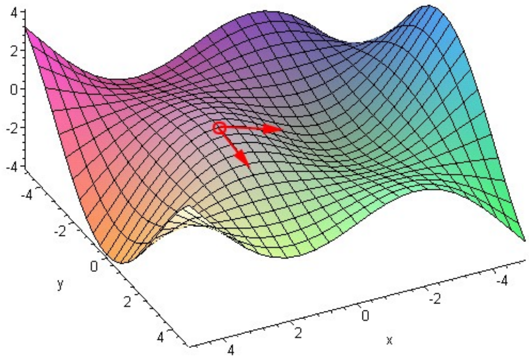
$$f := (x, y) \to \cos\left(\frac{1}{2}x\right)\cos\left(\frac{1}{2}y\right)x$$

28

Directional derivatives

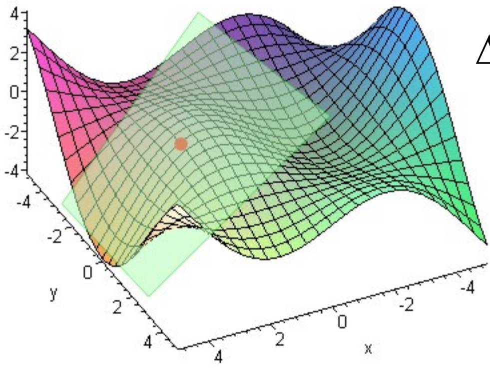
$$\frac{\partial f(x,y)}{\partial y}$$

$$\frac{\partial f(x,y)}{\partial x}$$



The gradient properties

- The gradient defines a (hyper) plane approximating the function infinitesimally



$$\Delta z = \frac{\partial f}{\partial x} \cdot \Delta x + \frac{\partial f}{\partial y} \cdot \Delta y$$

The gradient properties

- **Proposition**: let $f : R^n \to R$ be a smooth C^1 function around p, if f has local minimum (or maximum) at p then,

$$(\nabla f)_p = \bar{0}$$

Gradient (steepest descent) algorithm

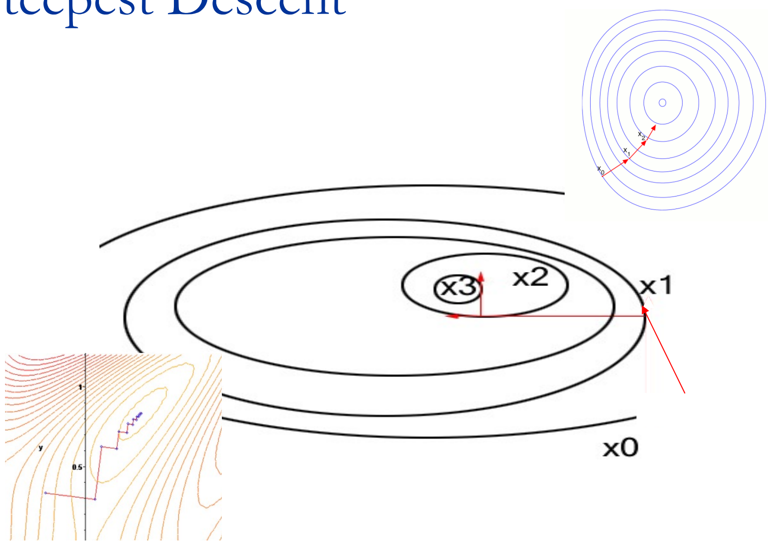
i=0

$x_0 \in R^n$

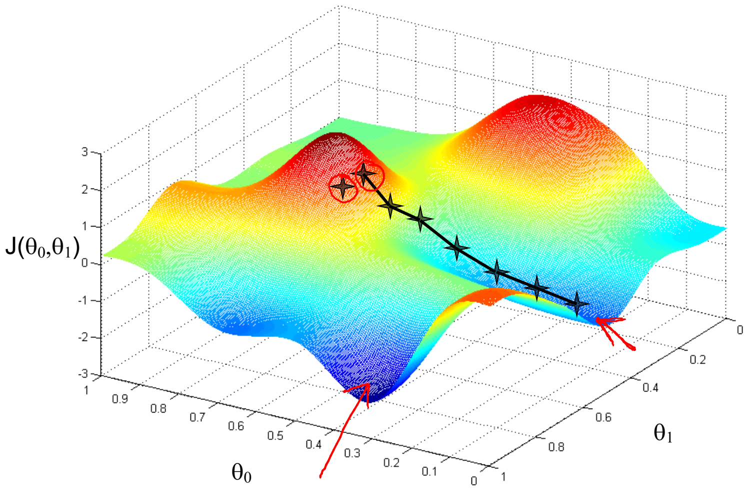
Repeat until $\nabla f(x_i) = 0$

1. compute *search direction* $h_i = -\nabla f(x_i)$
2. update $x_{i+1} = x_i + \eta \cdot h_i$

Steepest Descent



The effect of initial conditions

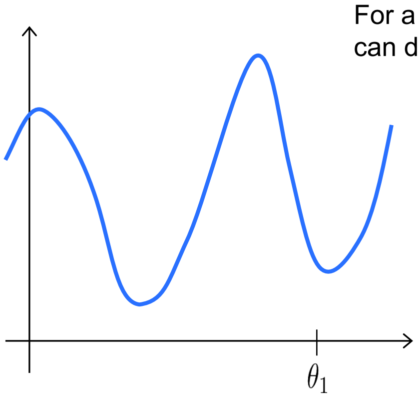


Local minima and weight initialization

- The stepest decent gradient finds a minimum, not necessarily a global minimum,
- Run the algorithm *N* times with small different random initial values for the weights.

Learning step size parameter, η

For a too high η, gradient can diverge



For a too small η convergence will be too slow

Gradient and regression in brief

Gradient

repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

$$(\text{for } j = 1 \text{ and } j = 0)$$

}

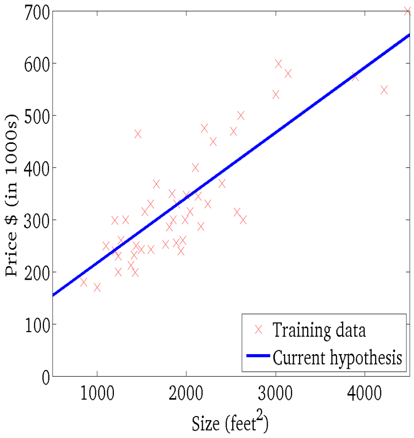
Linear regression model

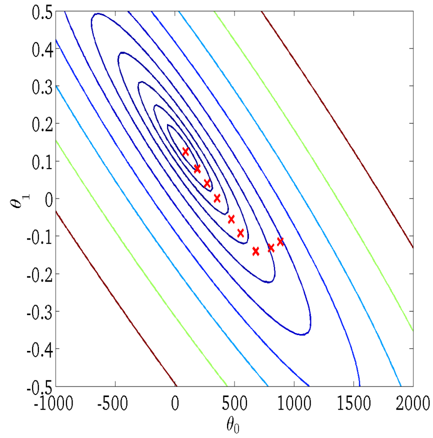
$$h_\theta(x) = \theta_0 + \theta_1 x$$

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

37

Evolution of J(θ_0, θ_1)





38

To take away today