UNIVERSITY OF LEEDS

Class: Machine Learning

Linear Models for Regression

Instructor: Matteo Leonetti

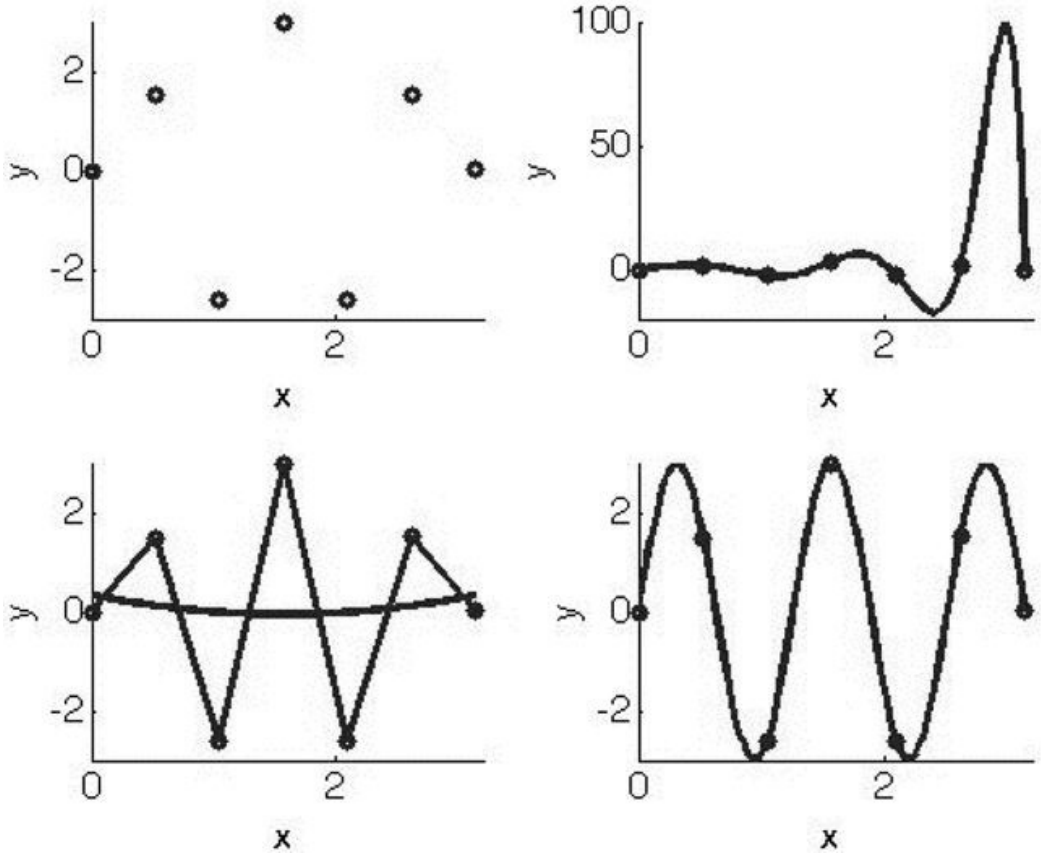
Learning outcomes

UNIVERSITY OF LEEDS

- Approximate a function through linear regression with different basis functions
- Compute the pseudo-inverse of a matrix
- Derive the bias-variance decomposition

Back to regression

UNIVERSITY OF LEEDS



Linear regression

UNIVERSITY OF LEEDS

Unknown function $t = f(\boldsymbol{x})$

Samples: $\langle \boldsymbol{x}_i, t_i \rangle$

We approximate the function with a linear model (in the parameters):

$$y(\boldsymbol{x}, \boldsymbol{w}) = w_0 + w_1 x_1 + w_2 x_3 + \cdots + w_M x_M$$

$$y(\boldsymbol{x}, \boldsymbol{w}) = w_0 + \sum_{i=1}^{M} w_i \phi_i(x)$$

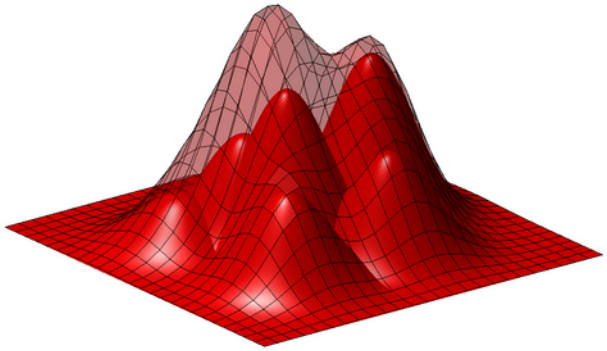
Basis functions:

Basis functions

UNIVERSITY OF LEEDS

Polynomial basis: $\phi_i(x) = x^i$

$$y(x, \boldsymbol{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M$$



"Gaussian" basis: $\phi_i(x) = \exp\left\{-\frac{(x-\mu_i)^2}{2s^2}\right\}$

sigmoid basis: $\phi_i(x) = \sigma\left(\frac{x-\mu}{s}\right) = \sigma_{\mu,s}$

$$\sigma(a) = \frac{1}{1+\exp(-a)}$$

System of equations

UNIVERSITY OF LEEDS

Samples: $\langle \boldsymbol{x_i}, t_i \rangle = \langle x_{i,1}, x_{i,2}, \cdots, x_{i,m}, t_i \rangle$

Let's set: $\phi_0 = 1$ So that: $y(\boldsymbol{x}, \boldsymbol{w}) = w_0 + \sum_{i=1}^{M} w_i \phi_i(x) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x})$

For each point: $\boldsymbol{x}_j$ We denote: $\phi_i(\boldsymbol{x}_j) = \Phi_{i,j}$

Each point imposes a constraint:

$$\boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}_i) = t_i$$

For N points in the data set, we have N equations.

Can we find a weight vector that satisfies all the constraints?

An Example - data

UNIVERSITY OF LEEDS

$\langle -2, 18.28 \rangle$

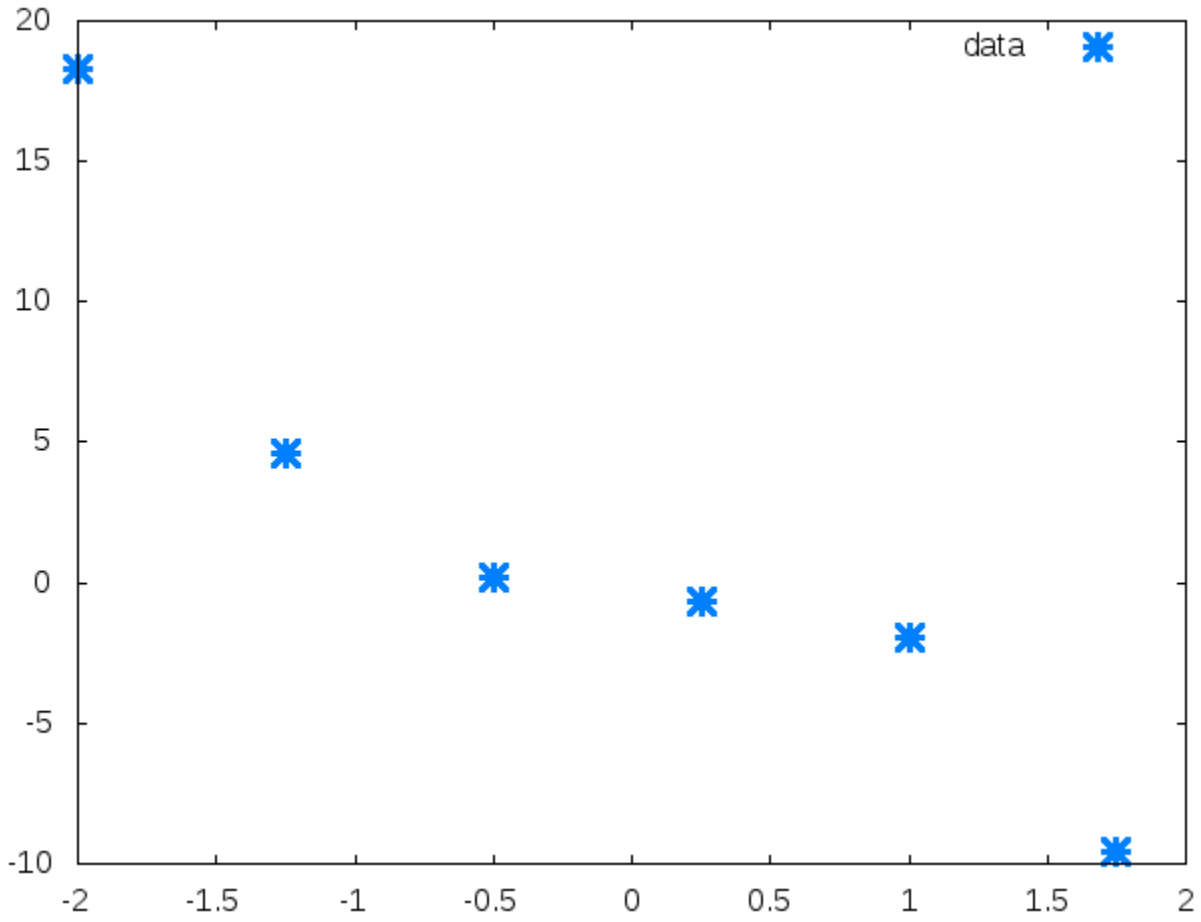
$\langle -1.25, 4.64 \rangle$

$\langle -0.5, 0.16 \rangle$

$\langle 0.25, -0.63 \rangle$

$\langle 1.75, -9.56 \rangle$

$\langle 1, -1.95 \rangle$



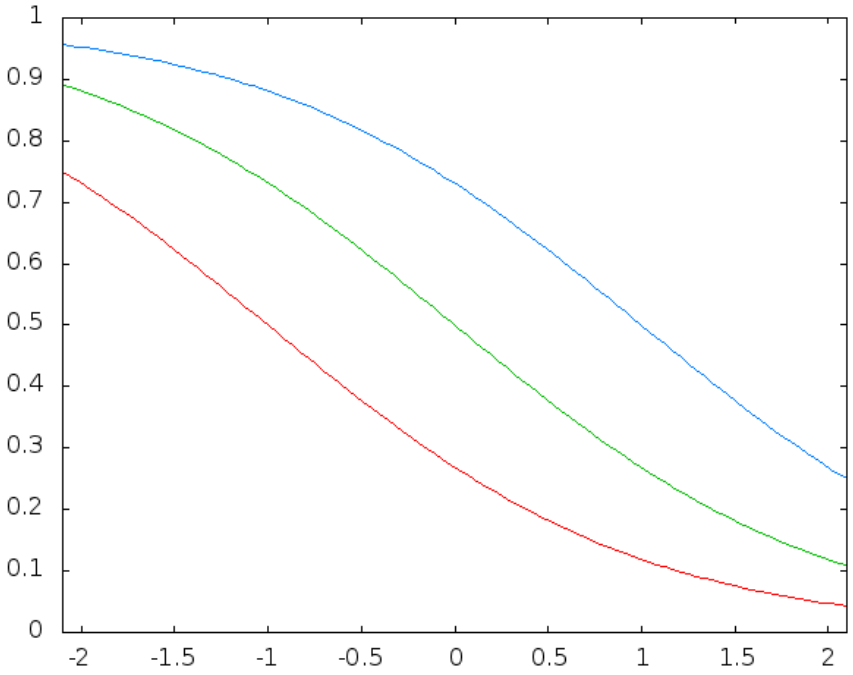
Choose model

UNIVERSITY OF LEEDS

How many sigmoids? Centred where? $y = ?$

Let's use these three:

$$y = w_0 + w_1 \sigma_{-1,1}(x) + w_2 \sigma_{0,1}(x) + w_3 \sigma_{1,1}(x)$$



Equations

UNIVERSITY OF LEEDS

We create a system of equations by evaluating the basis functions on the data points:

$$w_0 + w_1 \sigma_{-1,1}(x) + w_2 \sigma_{0,1}(x) + w_3 \sigma_{1,1}(x) = t$$

$\Phi_{1,3}$

$\langle -2, 18.28 \rangle$ $\quad w_0 + w_1 \sigma_{-1,1}(-2) + w_2 \sigma_{0,1}(-2) + w_3 \sigma_{1,1}(-2) = 18.28$

$\langle -1.25, 4.64 \rangle$ $\quad w_0 + w_1 \sigma_{-1,1}(-1.25) + w_2 \sigma_{0,1}(-1.25) + w_3 \sigma_{1,1}(-1.25) = 4.64$

$\langle -0.5, 0.16 \rangle$ $\quad w_0 + w_1 \sigma_{-1,1}(-0.5) + w_2 \sigma_{0,1}(-0.5) + w_3 \sigma_{1,1}(-0.5) = 0.16$

$\langle 0.25, -0.63 \rangle$ $\quad \ldots$

$\langle 1, -1.95 \rangle$ $\quad \ldots$

$\Phi_{3,2}$

$\langle 1.75, -9.56 \rangle$ $\quad \ldots$

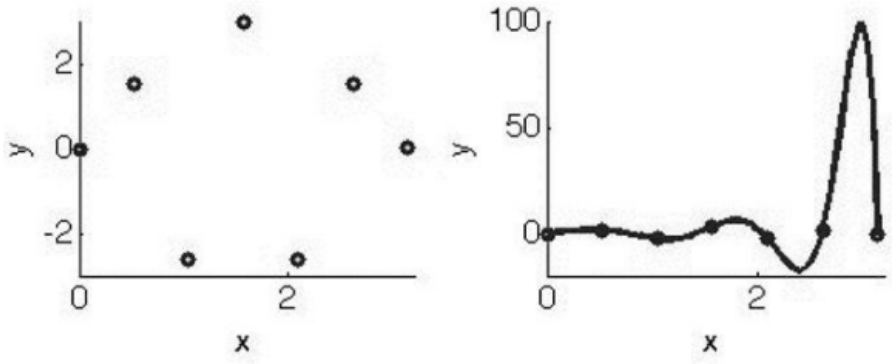
In matrix form

UNIVERSITY OF LEEDS

$$\begin{bmatrix} \Phi_{1,1} & \Phi_{1,2} & \cdots & \Phi_{1,M+1} \\ \Phi_{2,1} & \Phi_{2,2} & \cdots & \Phi_{2,M+1} \\ \vdots & \vdots & \vdots & \vdots \\ \Phi_{N,1} & \Phi_{N,2} & \cdots & \Phi_{N,M+1} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_M \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{bmatrix}$$

or: $\boldsymbol{\Phi}\, \boldsymbol{w} = \boldsymbol{t}$

If $\boldsymbol{\Phi}$ square and nonsingular: $\boldsymbol{w} = \boldsymbol{\Phi}^{-1} \boldsymbol{t}$



No error, the fitted function goes through every point

Exact solution

UNIVERSITY OF LEEDS

If we only had 4 points, we could find an exact solution, that is a function that goes through the points:

$$\boldsymbol{w} = \boldsymbol{\Phi}^{-1}\boldsymbol{t}$$

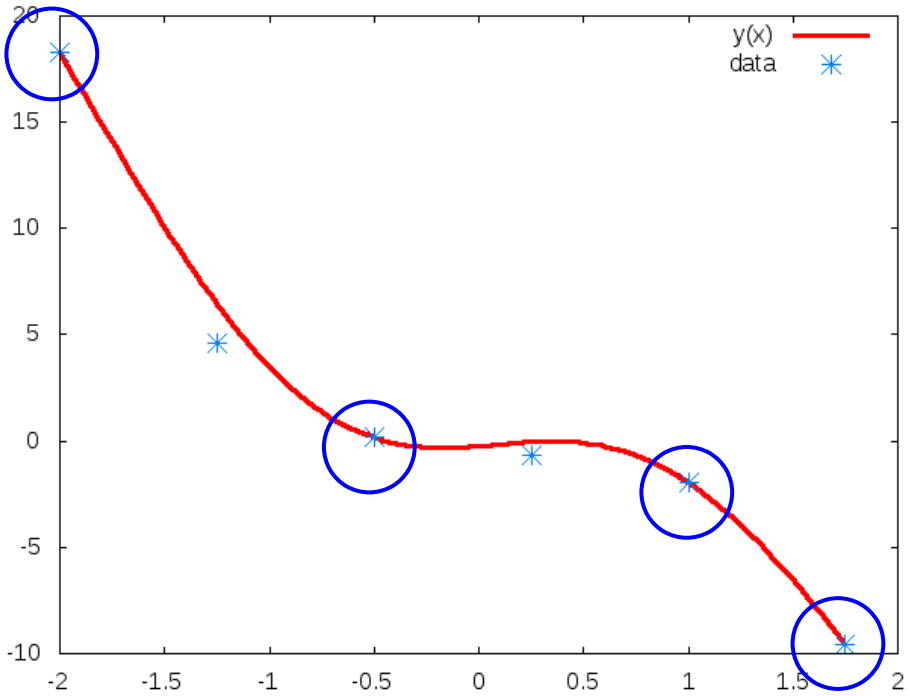
$\langle -2, 18.28 \rangle$

$\langle -0.5, 0.16 \rangle$

$\langle 1, -1.95 \rangle$

$\langle 1.75, -9.56 \rangle$

$$\vec{w} = \langle -34.99, 184.249, -312.42, 209.20 \rangle$$



Overdetermined vector

UNIVERSITY OF LEEDS

If vector w overdetermined (more equations than variables) → no exact solution available.

Then (guess what), define an error and minimize.

You would like:

$$\boldsymbol{\phi}_i^T(\boldsymbol{x}_i)\boldsymbol{w} = t_i \quad \forall\, \boldsymbol{x}_i$$

So a good error seems to be the difference between the left and right-end side of that equation:

$$E = \frac{1}{2}\sum_i^N (\boldsymbol{\phi}_i^T \boldsymbol{w} - t_i)^2$$

Sum-of-squares error

UNIVERSITY OF LEEDS

Let's ignore the vector notation for a minute:

$$E = \frac{1}{2}\sum_{i=1}^{N}(y_i - t_i)^2 = \frac{1}{2}\sum_{i=1}^{N}(\phi_i w - t_i)^2$$

What's the gradient?

$$\nabla E = \sum_{i=1}^{N}\phi_i(\phi_i w - t_i)$$

Sum-of-squares error

UNIVERSITY OF LEEDS

Reintroducing vectors:

$$E = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - t_i\right)^2 = \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{\phi}_i^T \boldsymbol{w} - t_i\right)^2 = \frac{1}{2}\left\|\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{t}\right\|^2$$

$$\nabla E = \sum_{i=1}^{N}\left(\boldsymbol{\phi}_i^T \boldsymbol{w} - t_i\right)\boldsymbol{\phi}_i = \boldsymbol{\Phi}^T\left(\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{t}\right)$$

Least squares solution

UNIVERSITY OF LEEDS

$$\Phi^T(\Phi w - t) = 0 \qquad \Rightarrow \Phi^T \Phi w = \Phi^T t$$

Necessary condition for a vector w to be a minimum

$\Phi^T \Phi$ Is a square matrix.

If it is also non singular, we can invert it and solve the equation above:

$$w = (\Phi^T \Phi)^{-1} \Phi^T t$$

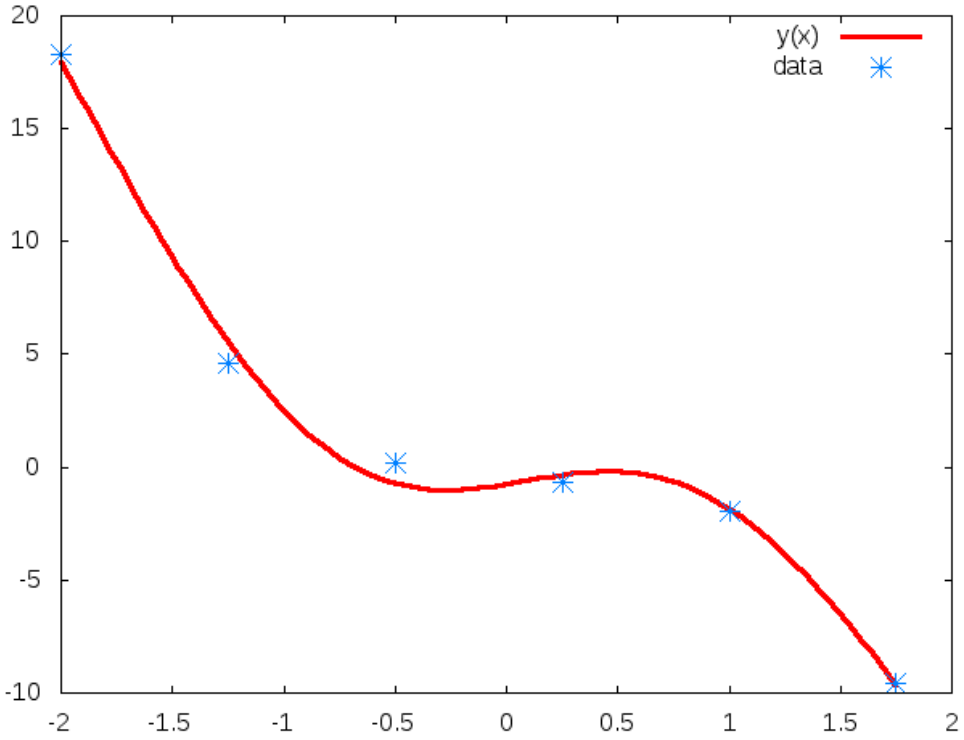
where $\Phi_p = (\Phi^T \Phi)^{-1} \Phi^T$ is the *pseudoinverse* of Φ

Least-squares solution

UNIVERSITY OF LEEDS

$$\boldsymbol{w} = \left(\boldsymbol{\Phi}^{\boldsymbol{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\boldsymbol{T}}\boldsymbol{t}$$

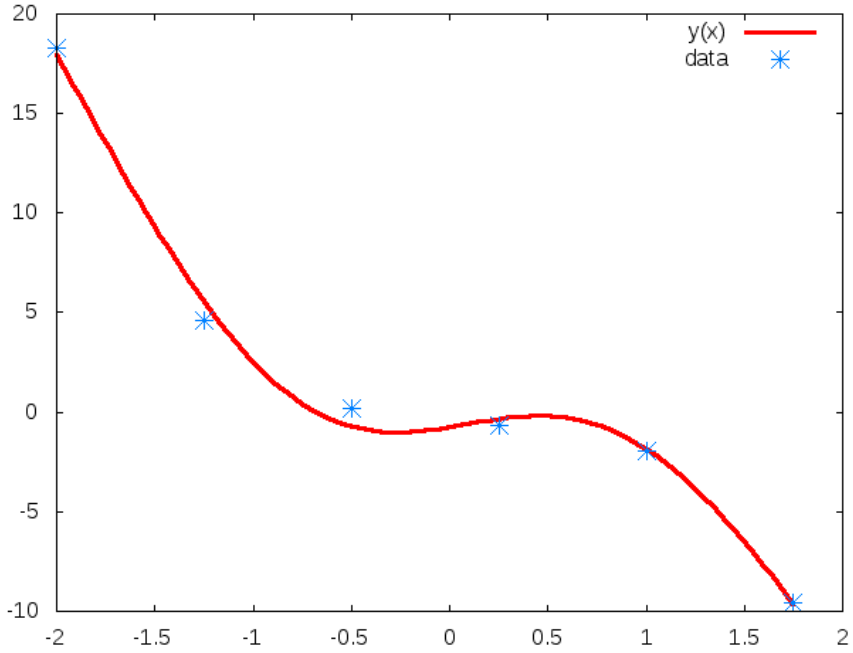
$$\boldsymbol{w} = \langle -36.53, 198.30, -339.45\,, 225.09 \rangle$$



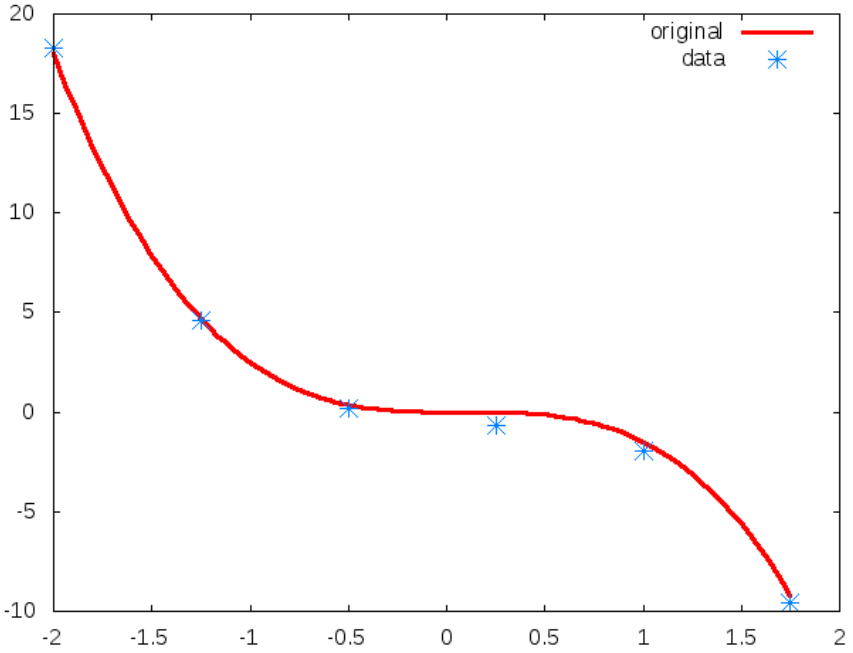
Comparison

UNIVERSITY OF LEEDS

Least-squares



Original function



$$f(x) = 0.5x^2 - 2x^3$$

Exercise

UNIVERSITY OF LEEDS

Given the dataset: {(0,0), (0,1), (1,0)}, find the least-squares solution for the parameters in the regression of the function: $y = w_1 + w_2 x^2$

1) Evaluate the bases on the points: $\boldsymbol{\Phi} = ?$

2) Compute: $\boldsymbol{\Phi}^T \boldsymbol{\Phi} = ?$

3) Invert: $\left(\boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} = ?$

4) Compute the pseudo-inverse: $\boldsymbol{\Phi}_p = \left(\boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T = ?$

5) Compute w! $\boldsymbol{w} = \boldsymbol{\Phi}_p \boldsymbol{t} = ?$

Exercise

UNIVERSITY OF LEEDS

Given the dataset: {(0,0), (0,1), (1,0)}, find the least-squares solution for the parameters in the regression of the function: $y = w_1 + w_2 x^2$

1) Evaluate the bases on the points: $\mathbf{\Phi} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$

Exercise

UNIVERSITY OF LEEDS

Given the dataset: {(0,0), (0,1), (1,0)}, find the least-squares solution for the parameters in the regression of the function: $y = w_1 + w_2 x^2$

1) Evaluate the bases on the points: $\mathbf{\Phi} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$

2) Compute: $\mathbf{\Phi}^T \mathbf{\Phi} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$

Exercise

UNIVERSITY OF LEEDS

Given the dataset: {(0,0), (0,1), (1,0)}, find the least-squares solution for the parameters in the regression of the function: $y = w_1 + w_2 x^2$

1) Evaluate the bases on the points: $\mathbf{\Phi} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$

2) Compute: $\mathbf{\Phi}^T \mathbf{\Phi} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$

3) Invert: $\left(\mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} = \begin{bmatrix} 0.5 & -0.5 \\ -0.5 & 1.5 \end{bmatrix}$

Exercise

UNIVERSITY OF LEEDS

Given the dataset: {(0,0), (0,1), (1,0)}, find the least-squares solution for the parameters in the regression of the function: $y = w_1 + w_2 x^2$

1) Evaluate the bases on the points: $\mathbf{\Phi} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$

2) Compute: $\mathbf{\Phi}^T \mathbf{\Phi} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$

3) Invert: $\left(\mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} = \begin{bmatrix} 0.5 & -0.5 \\ -0.5 & 1.5 \end{bmatrix}$

4) Compute the pseudo-inverse: $\mathbf{\Phi}_p = \left(\mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^T = \begin{bmatrix} 0.5 & 0.5 & 0 \\ -0.5 & -0.5 & 1 \end{bmatrix}$

Exercise

UNIVERSITY OF LEEDS

Given the dataset: {(0,0), (0,1), (1,0)}, find the least-squares solution for the parameters in the regression of the function: $y = w_1 + w_2 x^2$

1) Evaluate the bases on the points: $\boldsymbol{\Phi} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$

2) Compute: $\boldsymbol{\Phi}^T \boldsymbol{\Phi} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$

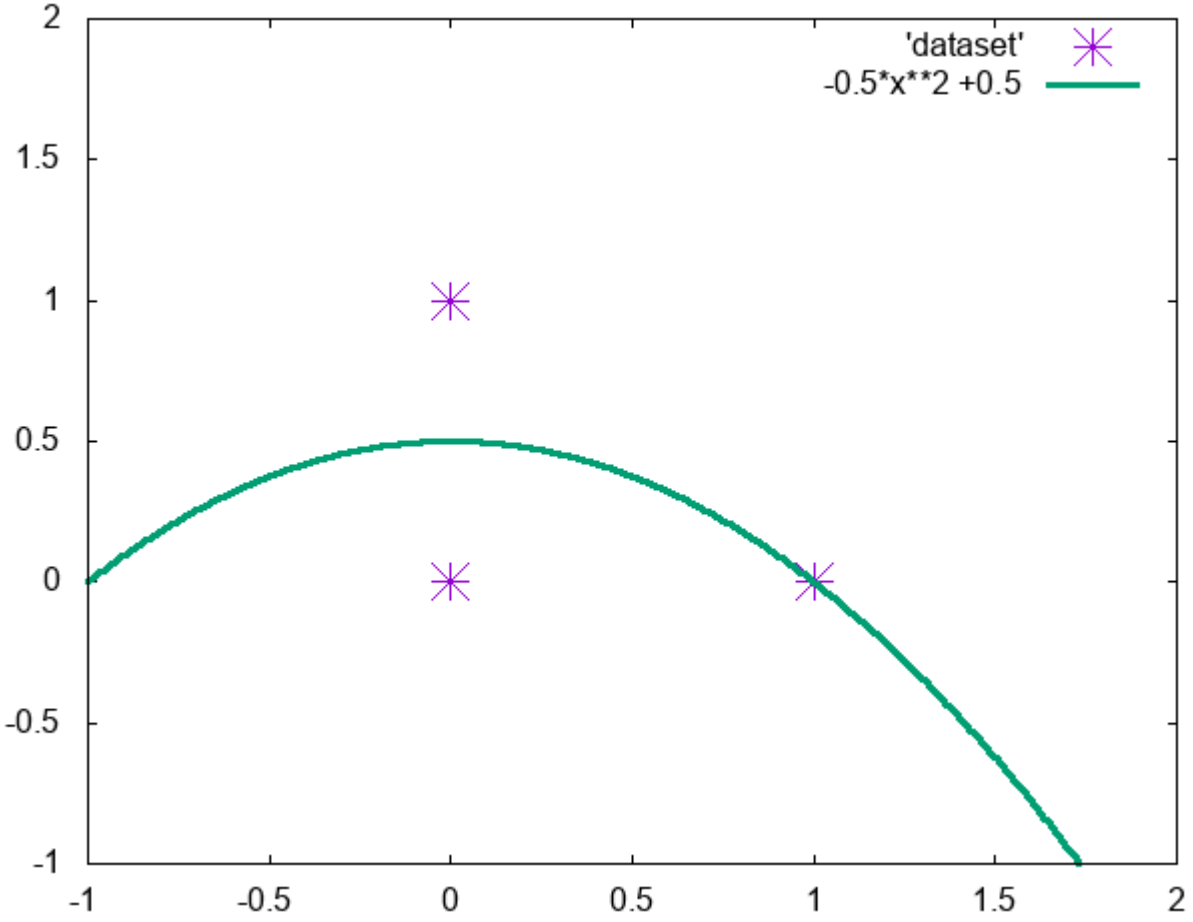
3) Invert: $\left(\boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} = \begin{bmatrix} 0.5 & -0.5 \\ -0.5 & 1.5 \end{bmatrix}$

4) Compute the pseudo-inverse: $\boldsymbol{\Phi}_p = \left(\boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T = \begin{bmatrix} 0.5 & 0.5 & 0 \\ -0.5 & -0.5 & 1 \end{bmatrix}$

5) Compute w! $\boldsymbol{w} = \boldsymbol{\Phi}_p \boldsymbol{t} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ -0.5 & -0.5 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.5 \\ -0.5 \end{bmatrix}$

Result

UNIVERSITY OF LEEDS



Sequential learning

UNIVERSITY OF LEEDS

The closed-form solution requires to process the whole data set.

If $$E = \sum_n E_n$$

As with neural networks, it is also possible to consider one point at a time

$$w^{(t+1)} = w^{(t)} - \eta \nabla E_n$$

For a least-squares problem:

$$w^{(t+1)} = w^{(t)} - \eta \boldsymbol{\phi}_n^T (\boldsymbol{\phi}_n^T \boldsymbol{w}^{(t)} - \boldsymbol{t}_n)$$

known as the Least-mean-squares (LMS) algorithm

Bias and Variance

UNIVERSITY OF LEEDS

Bias-Variance decomposition

UNIVERSITY OF LEEDS

Underlying function plus noise: $f(\vec{x})+\epsilon$ Generates different datasets D

Simplified notation: $\hat{y} \equiv y(\vec{x}; D)$ $f \equiv f(\vec{x})$

Expected value of the error with respect to datasets D (I excluded noise from these calculations, the book has a version with noise):

$$E_D[(\hat{y}-f)^2] = E_D[(\hat{y}-E_D[\hat{y}]+E_D[\hat{y}]-f)^2]$$

$$= E_D[(\hat{y}-E_D[\hat{y}])^2 + 2(\hat{y}-E_D[\hat{y}])(E_D[\hat{y}]-f) + (E_D[\hat{y}]-f)^2]$$

$$= E_D[(\hat{y}-E_D[\hat{y}])^2] + 2E_D[(\hat{y}-E_D[\hat{y}])(E_D[\hat{y}]-f)] + E_D[(E_D[\hat{y}]-f)^2]$$

$$2E_D[(\hat{y}-E_D[\hat{y}])](E_D[\hat{y}]-f)$$

$$E_D[(\hat{y}-E_D[\hat{y}])] = E_D[\hat{y}]-E_D[\hat{y}] = 0$$

Bias-Variance decomposition

UNIVERSITY OF LEEDS

$$= E_D[(\hat{y} - E_D[\hat{y}])^2] + 2\,E_D[(\hat{y} - E_D[\hat{y}])(E_D[\hat{y}] - f)] + E_D[(E_D[\hat{y}] - f)^2]$$

$$= E_D[(\hat{y} - E_D[\hat{y}])^2] + E_D[(E_D[\hat{y}] - f)^2]$$

Variance

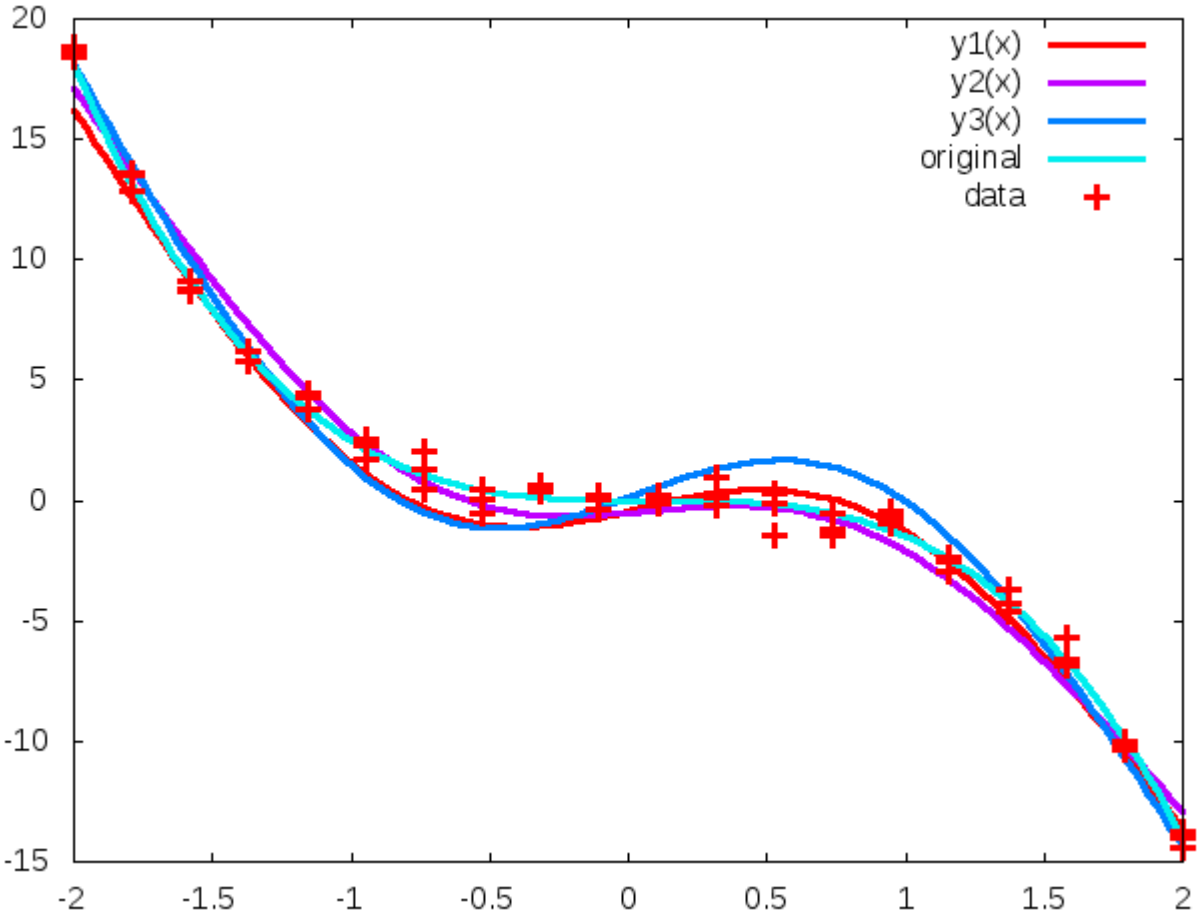
Bias^2

Recall that the variance of a random variable is: $var(X) = E[(X - E[X])^2]$

Not limited to regression with mean squared error: general phenomenon

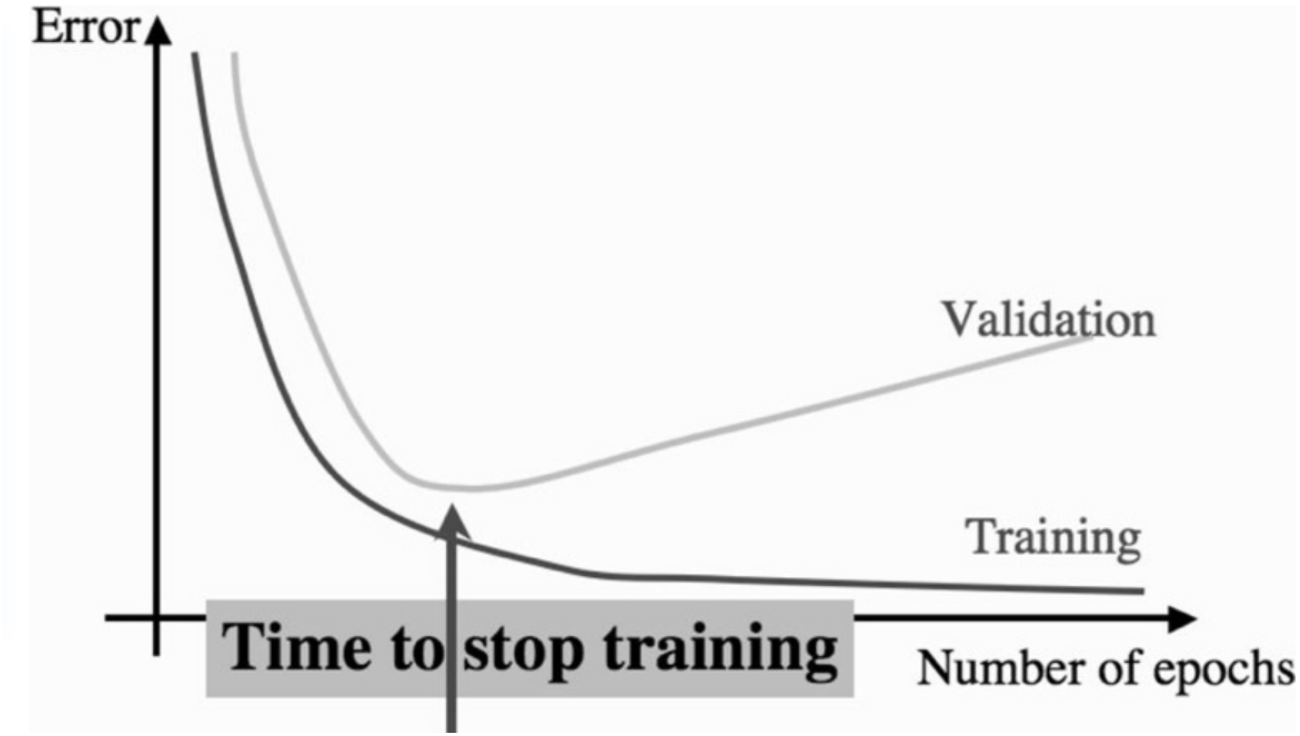
Bias-Variance

UNIVERSITY OF LEEDS

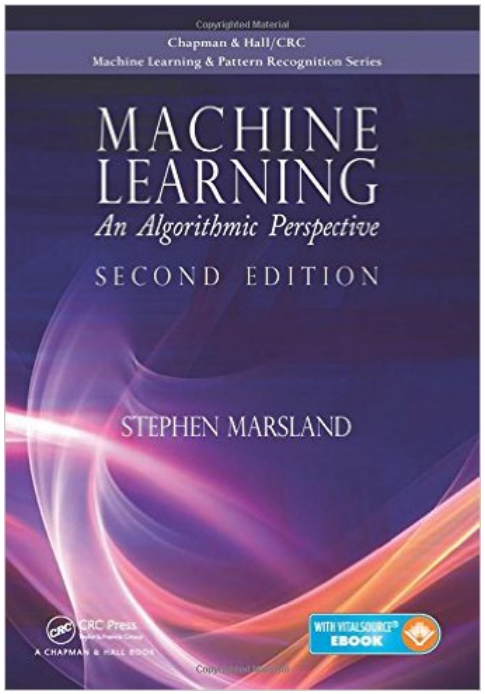


On training and validation error

UNIVERSITY OF LEEDS



UNIVERSITY OF LEEDS



Chapter 2.5 and 3.5