Smoothing and Interpolation

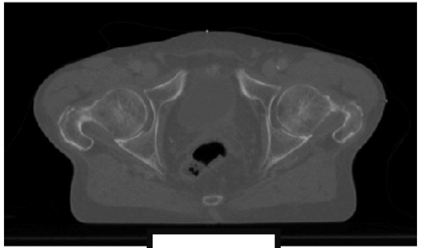
Medical Image Analysis

Koen Van Leemput

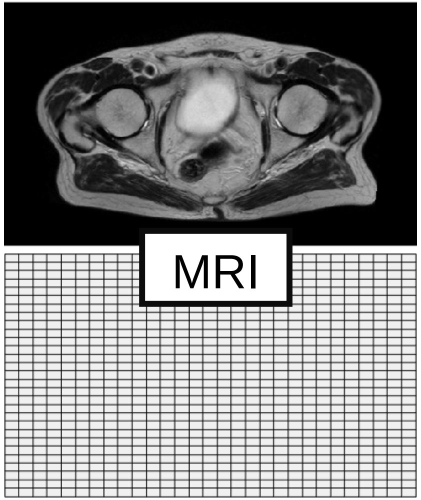
Aalto-yliopisto
Aalto-universitetet
Aalto University

Why care about smoothness?

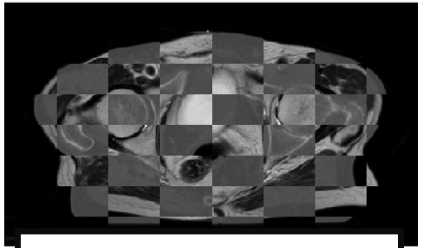
Example: image registration



CT



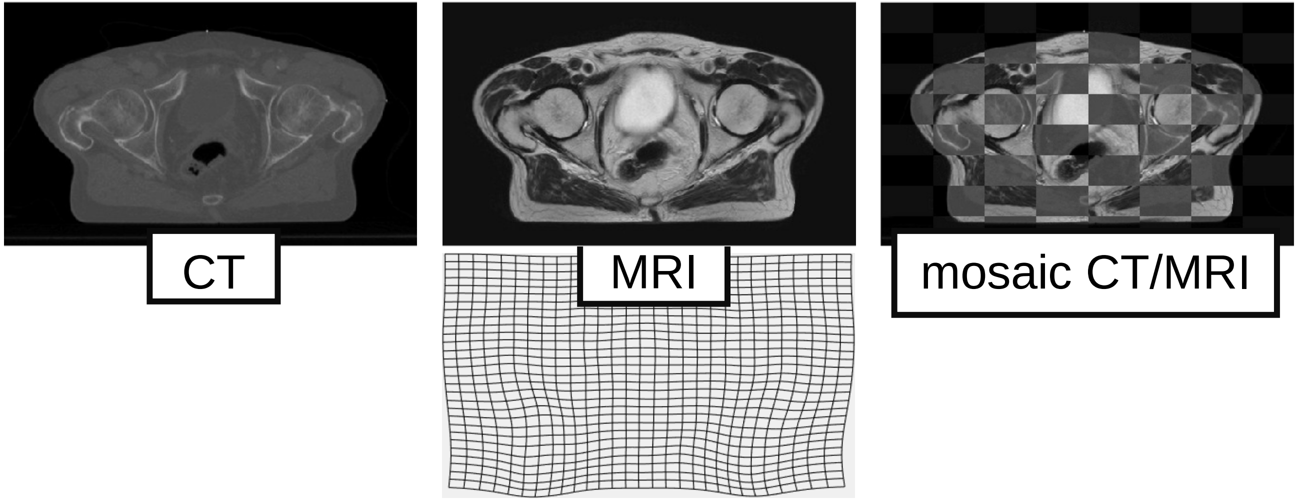
MRI



mosaic CT/MRI

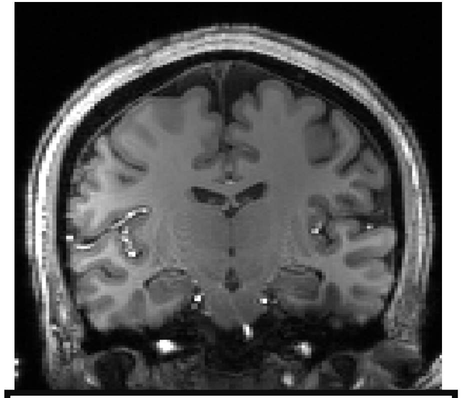
Why care about smoothness?

Example: image registration

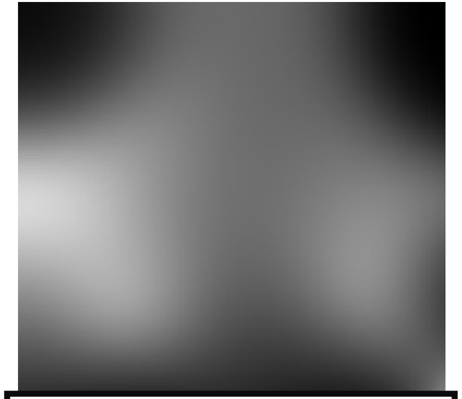


Why care about smoothness?

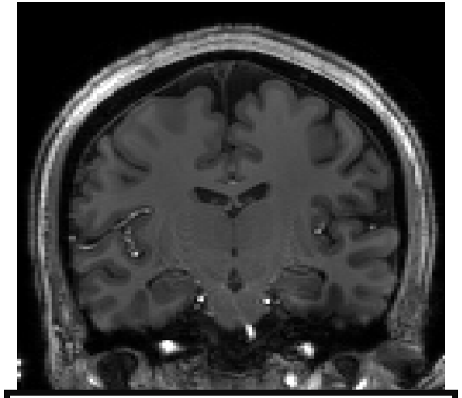
Example: image segmentation



MR scan (7 Tesla)



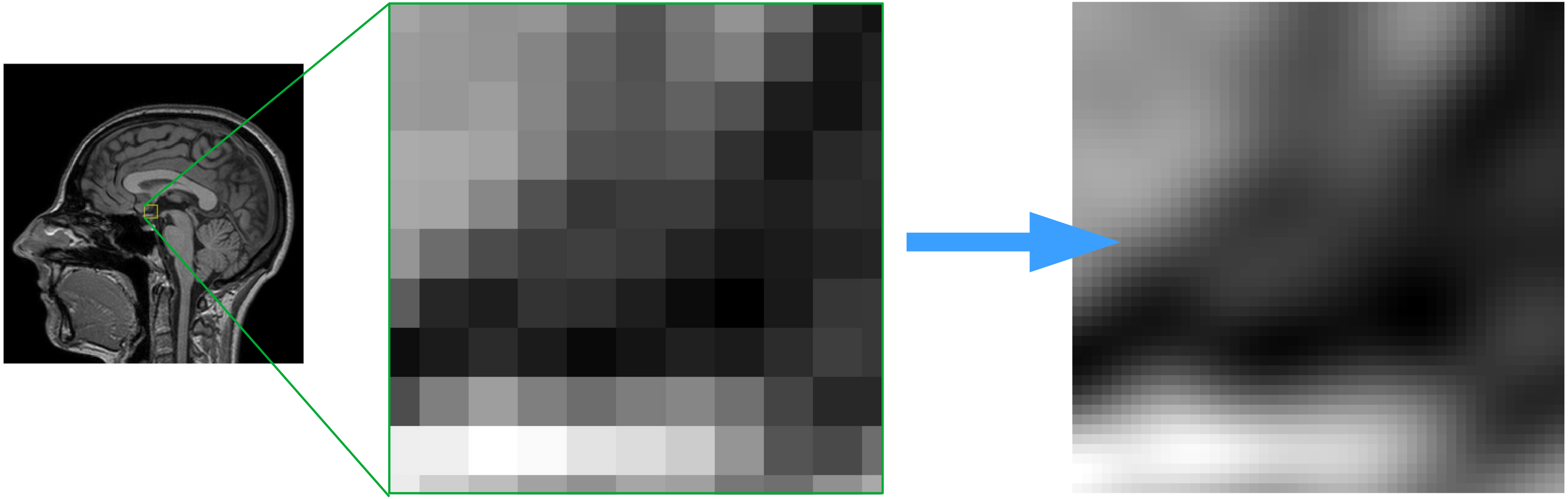
estimated “bias field”



bias field corrected

Why care about smoothness?

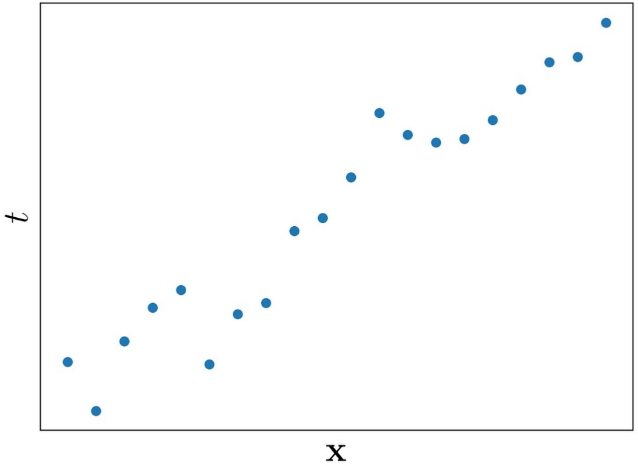
Example: image interpolation



Aalto-yliopisto
Aalto-universitetet
Aalto University

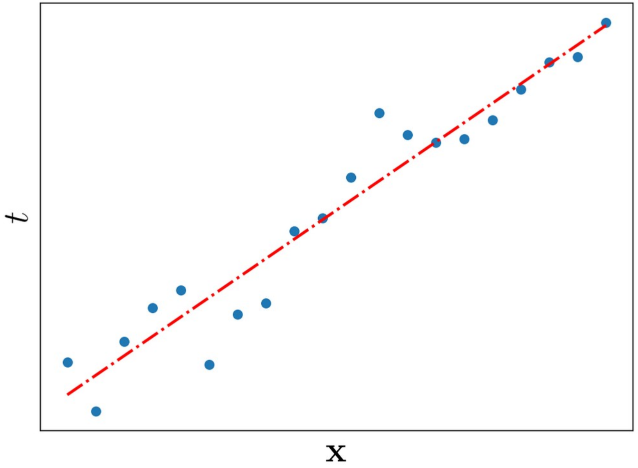
Linear regression

- ✔ Let $\mathbf{x} = (x_1, \ldots, x_D)^{\mathrm{T}}$ denote a spatial position in a D-dimensional space
- ✔ Given N measurements $\{t_n\}_{n=1}^{N}$ at locations $\{\mathbf{x}_n\}_{n=1}^{N}$, what is t at a new location $\mathbf{x}$?



Linear regression

- ✔ Let $\mathbf{x} = (x_1, \ldots, x_D)^{\mathrm{T}}$ denote a spatial position in a D-dimensional space
- ✔ Given N measurements $\{t_n\}_{n=1}^{N}$ at locations $\{\mathbf{x}_n\}_{n=1}^{N}$, what is t at a new location $\mathbf{x}$?



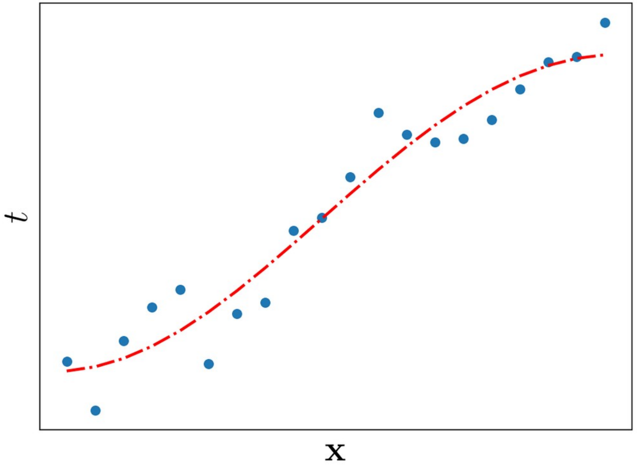
Aalto-yliopisto
Aalto-universitetet
Aalto University

$$y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + \ldots + w_D x_D$$

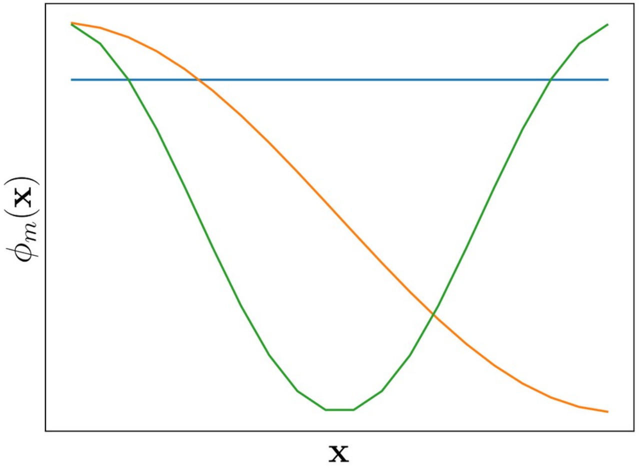
tunable weights

Linear regression

- ✔ Let $\mathbf{x} = (x_1, \ldots, x_D)^{\mathrm{T}}$ denote a spatial position in a D-dimensional space
- ✔ Given N measurements $\{t_n\}_{n=1}^{N}$ at locations $\{\mathbf{x}_n\}_{n=1}^{N}$, what is t at a new location $\mathbf{x}$?



nonlinear basis functions



$$y(\mathbf{x}, \mathbf{w}) = \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x})$$

tunable weights

Linear regression

- ✔ What are “suitable” values for the weights $\mathbf{w} = (w_0, \ldots, w_{M-1})^{\mathrm{T}}$?
- ✔ Minimize the energy $E(\mathbf{w}) = \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2$

Linear regression

- ✔ What are “suitable” values for the weights $\mathbf{w} = (w_0, \ldots, w_{M-1})^{\mathrm{T}}$?
- ✔ Minimize the energy $E(\mathbf{w}) = \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2$

Task: find w that minimizes $E(w) = (5 - 4w)^2 + (3 - 2w)^2$

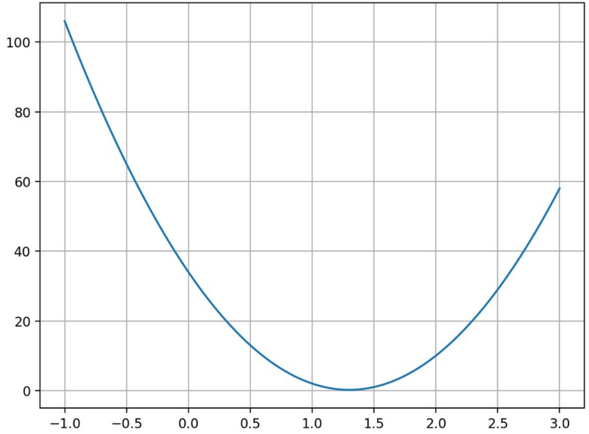
Linear regression

- ✔ What are "suitable" values for the weights $\mathbf{w} = (w_0, \ldots, w_{M-1})^\mathrm{T}$?
- ✔ Minimize the energy $E(\mathbf{w}) = \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2$

Task: find w that minimizes $E(w) = (5 - 4w)^2 + (3 - 2w)^2$

$$\frac{dE(w)}{dw} = -8(5 - 4w) - 4(3 - 2w) = -52 + 40w$$

$$\frac{dE(w)}{dw} = 0 \quad \Rightarrow \quad w = 1.3$$



Linear regression

- ✔ What are “suitable” values for the weights $\mathbf{w} = (w_0, \ldots, w_{M-1})^{\mathrm{T}}$?
- ✔ Minimize the energy $E(\mathbf{w}) = \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2$

Linear regression

- ✔ What are “suitable” values for the weights $\mathbf{w} = (w_0, \ldots, w_{M-1})^{\mathrm{T}}$?

- ✔ Minimize the energy $E(\mathbf{w}) = \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2$

$$\frac{\partial E(\mathbf{w})}{\partial w_m} = -2 \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right) \phi_m(\mathbf{x}_n)$$

Linear regression

- ✔ What are "suitable" values for the weights $\mathbf{w} = (w_0, \ldots, w_{M-1})^{\mathrm{T}}$?
- ✔ Minimize the energy $E(\mathbf{w}) = \sum_{n=1}^{N} \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2$

$$\nabla E(\mathbf{w}) = \begin{pmatrix} \frac{\partial E(\mathbf{w})}{\partial w_0} \\ \vdots \\ \frac{\partial E(\mathbf{w})}{\partial w_{M-1}} \end{pmatrix} = -2\boldsymbol{\Phi}^{\mathrm{T}} (\mathbf{t} - \boldsymbol{\Phi}\mathbf{w}), \qquad \boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \ldots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \ldots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \ldots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

$$\mathbf{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$$

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

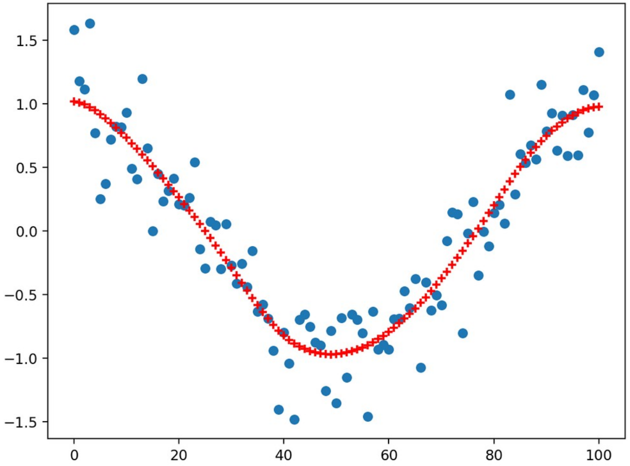
Smoothing

Let's concentrate on one-dimensional (1D) "images":

- ✔ Functions of the form $y(x, \mathbf{w}) = \sum_{m=0}^{M-1} w_m \phi_m(x),$ where the location x is a scalar
- ✔ Measurement points are defined on a regular grid: $x_1 = 0, x_2 = 1, \ldots x_N = N-1$

"Denoising":

- ✔ The measurements $t_n, n = 1, \ldots, N$ are noisy observations
- ✔ Recover the underlying signal $\hat{t}_n = y(x_n, \mathbf{w})$ at the locations x_n



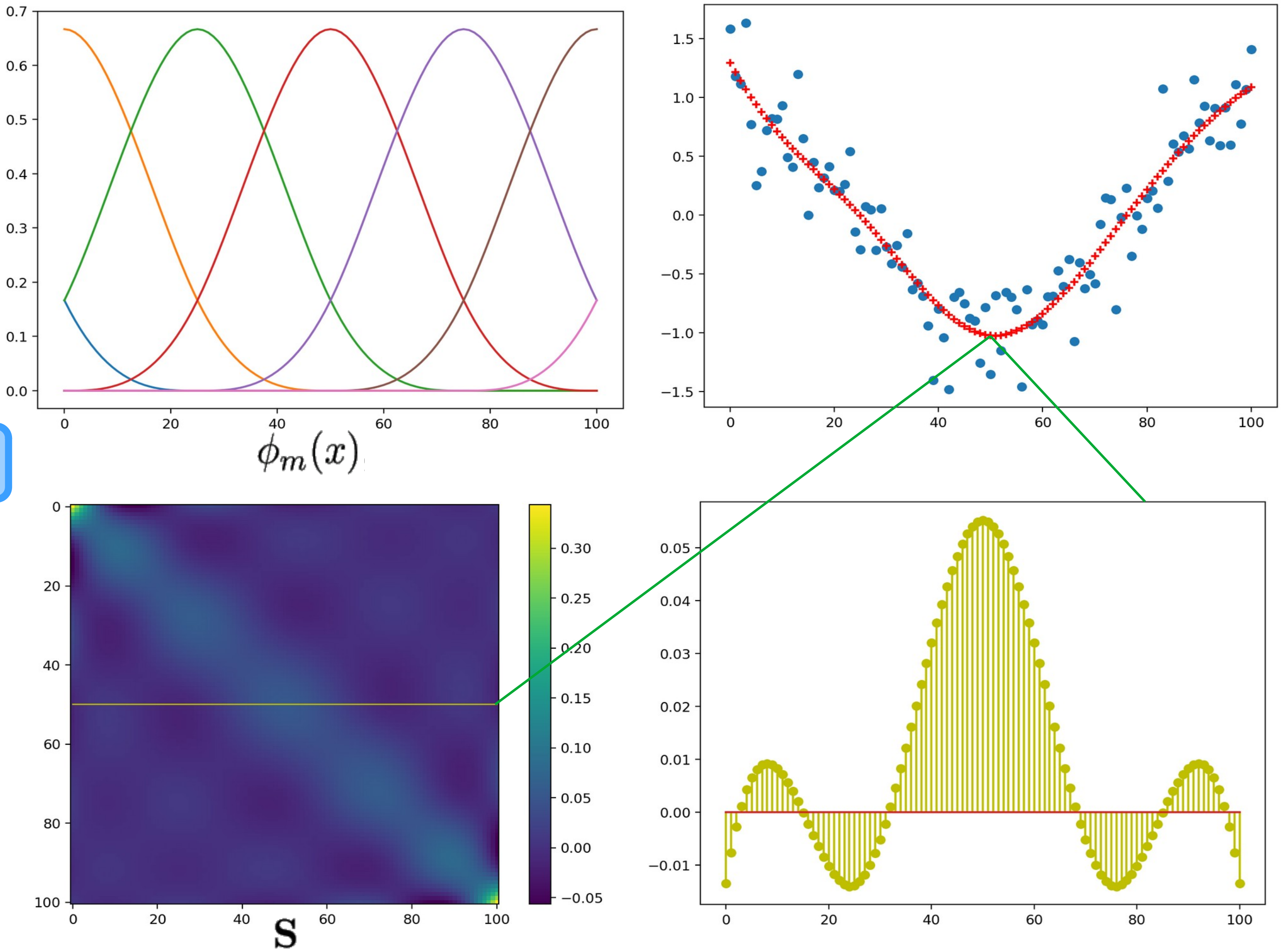
Smoothing

- ✔ We aim to recover $\hat{\mathbf{t}} = (\hat{t}_1, \ldots, \hat{t}_N)^{\mathrm{T}}$ from $\mathbf{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$

- ✔ Since $\hat{\mathbf{t}} = \mathbf{\Phi}\mathbf{w}$ and $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$:

$$\boxed{\hat{\mathbf{t}} = \mathbf{S}\mathbf{t} \text{ with } \mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}}$$

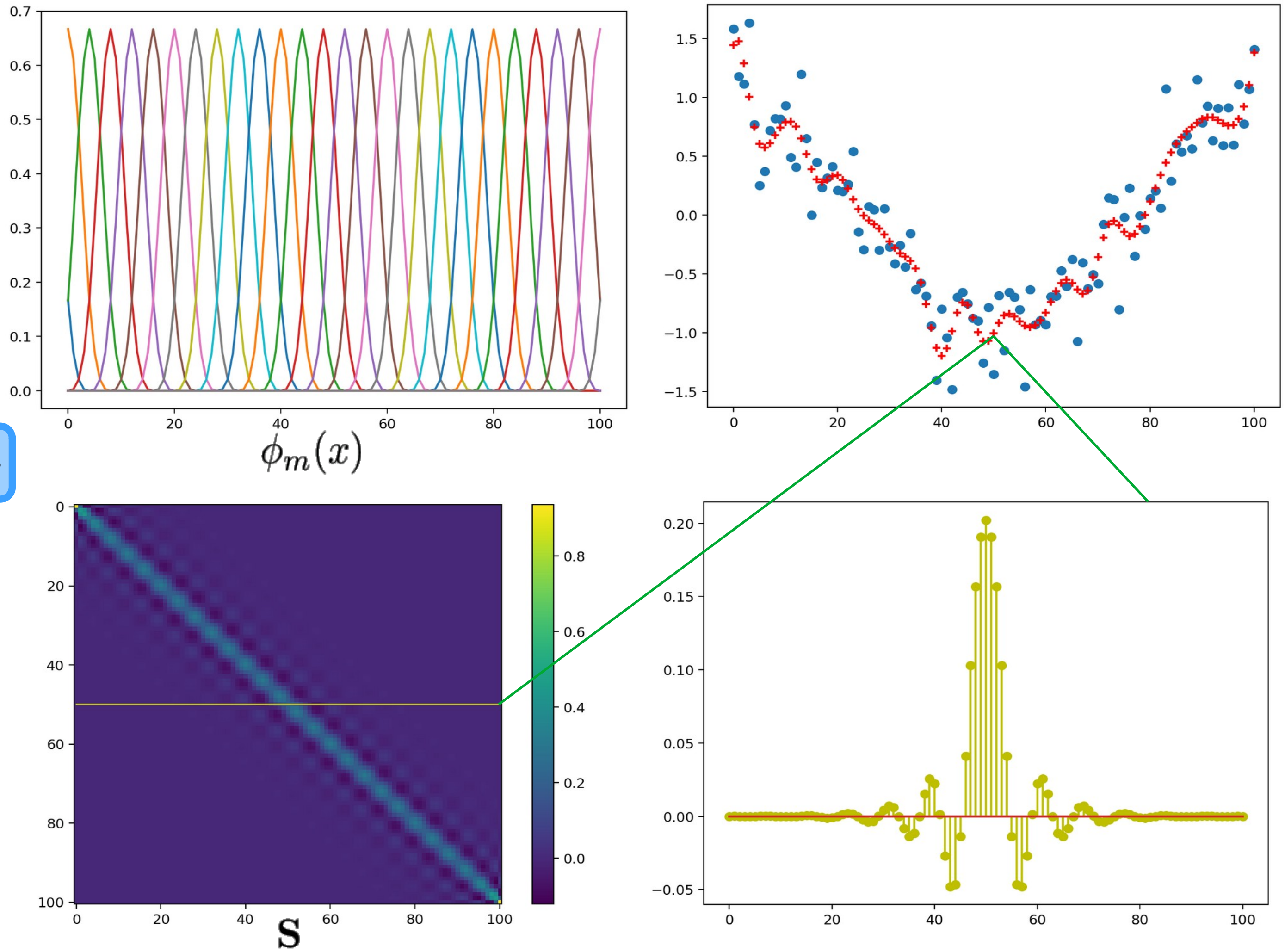
Example



M=7 basis functions

Example

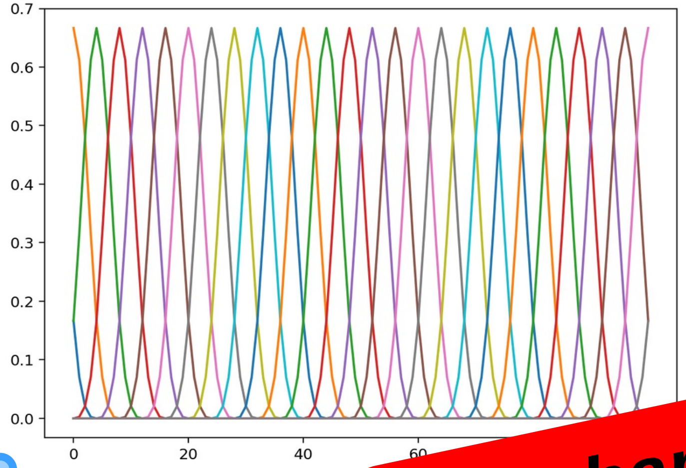
M=28 basis functions

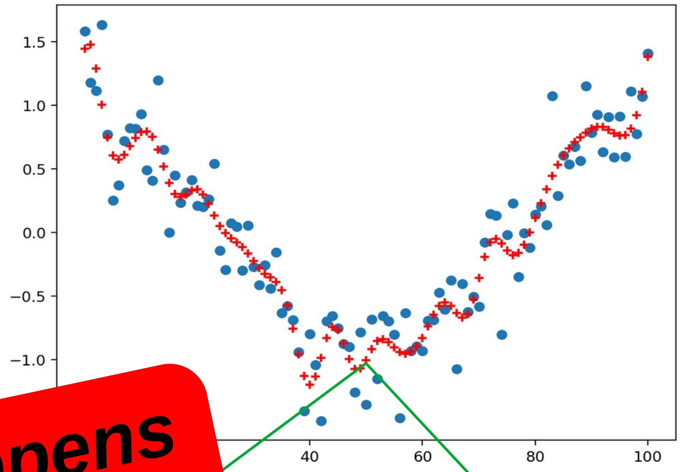


Aalto-yliopisto
Aalto-universitetet
Aalto University

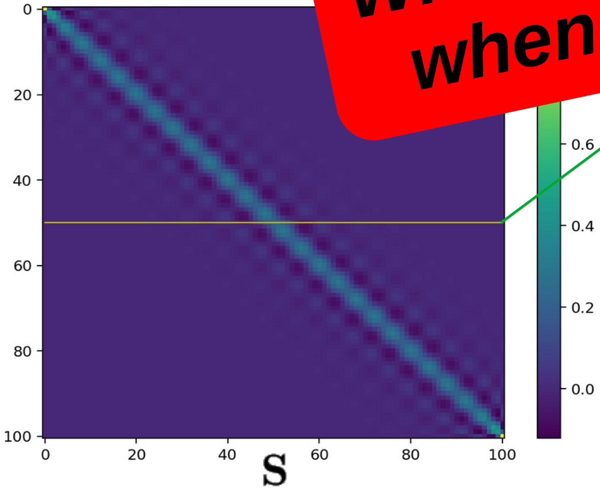
Example

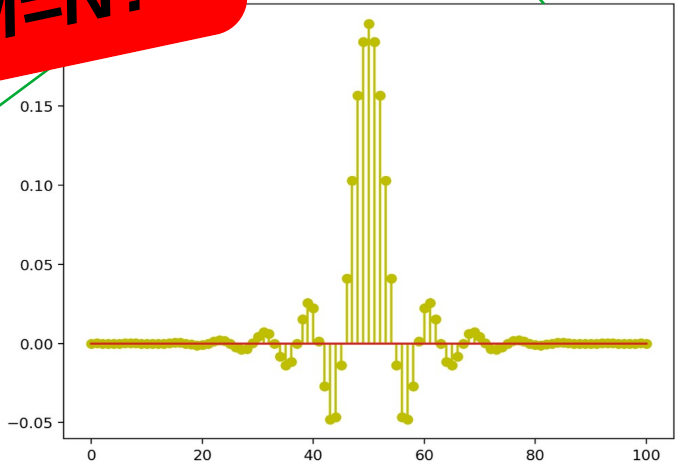
M=28 basis functions





What happens when M=N?





Discuss with your neighbor

Consider the case $\mathbf{t} = (2, 1, 3)^T$ **and** $\mathbf{\Phi} = (1, 1, 1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

Discuss with your neighbor

Consider the case $\mathbf{t} = (2, 1, 3)^T$ **and** $\mathbf{\Phi} = (1, 1, 1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

Discuss with your neighbor

Consider the case $\mathbf{t} = (2, 1, 3)^T$ **and** $\mathbf{\Phi} = (1, 1, 1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

$\mathbf{w} = 3^{-1}6 = 2$

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

Discuss with your neighbor

Consider the case $\mathbf{t} = (2,1,3)^T$ **and** $\mathbf{\Phi} = (1,1,1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

Discuss with your neighbor

Consider the case $\mathbf{t} = (2, 1, 3)^T$ **and** $\boldsymbol{\Phi} = (1, 1, 1)^T$

Task 1: what is $\mathbf{w} = \left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$?

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \boldsymbol{\Phi}\left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\boldsymbol{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

$$\hat{\mathbf{t}} = \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}$$

$$\mathbf{S} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \cdot 3^{-1} \cdot \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$$

Discuss with your neighbor

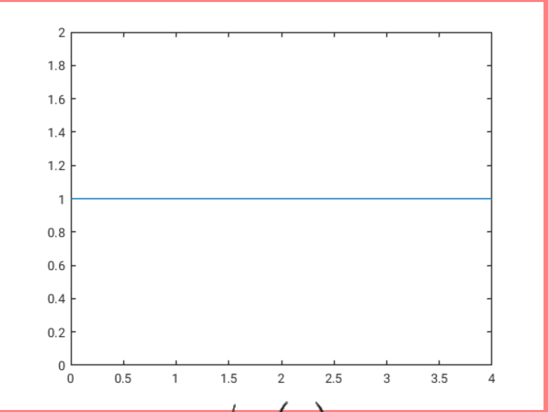
Consider the case $\mathbf{t} = (2, 1, 3)^T$ **and** $\mathbf{\Phi} = (1, 1, 1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

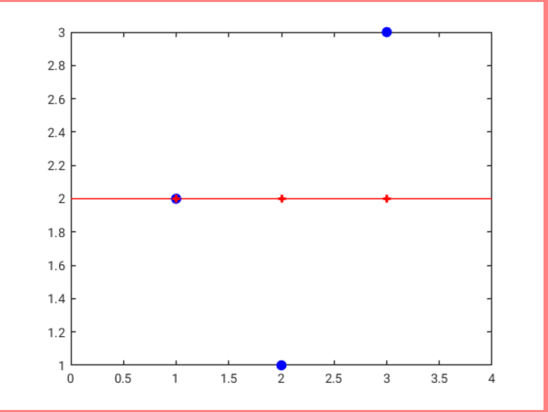
Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$



$\phi_m(x)$



Discuss with your neighbor

Consider the case $\mathbf{t} = (2,1,3)^T$ **and** $\boldsymbol{\Phi} = (1,1,1)^T$

Task 1: what is $\mathbf{w} = \left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$?

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \boldsymbol{\Phi}\left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\boldsymbol{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

Discuss with your neighbor

Consider the case $\mathbf{t} = (2,1,3)^T$ **and** $\mathbf{\Phi} = (1,1,1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$

$$\left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1} = \mathbf{\Phi}^{-1}\left(\mathbf{\Phi}^T\right)^{-1}$$

$$\mathbf{S} = \overbrace{\mathbf{\Phi}\mathbf{\Phi}^{-1}}^{\mathbf{I}} \underbrace{\left(\mathbf{\Phi}^T\right)^{-1}\mathbf{\Phi}^T}_{\mathbf{I}}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\hat{\mathbf{t}} = \mathbf{t} = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$$

Discuss with your neighbor

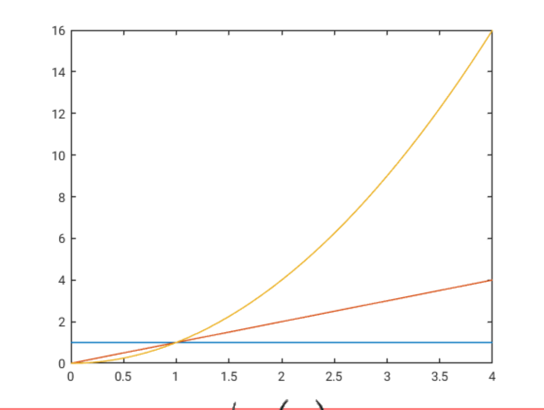
Consider the case $\mathbf{t} = (2,1,3)^T$ **and** $\mathbf{\Phi} = (1,1,1)^T$

Task 1: what is $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$?

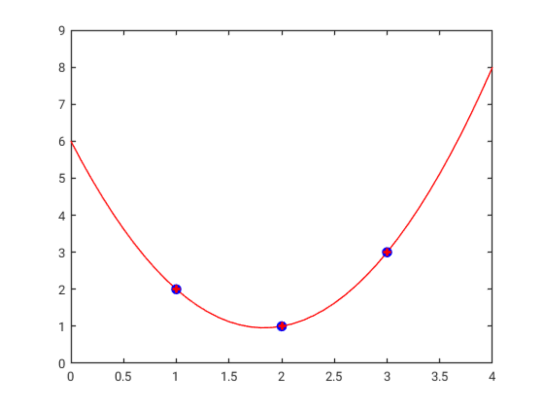
Task 2: what are $\hat{\mathbf{t}} = \mathbf{S}\mathbf{t}$ and $\mathbf{S} = \mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$?

Can you explain (e.g., draw) what's happening?

Task 3: same as task 2 but when $\mathbf{\Phi} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix}$



$\phi_m(x)$

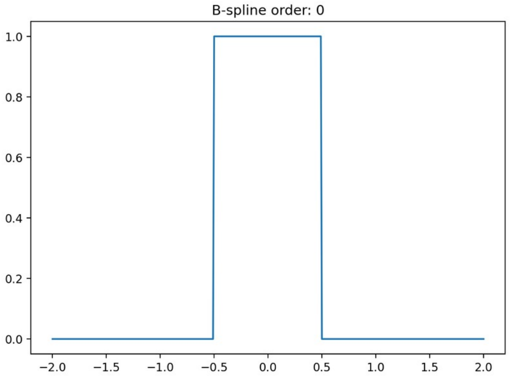


A? ***When M=N, no smoothing is applied!***

Interpolation

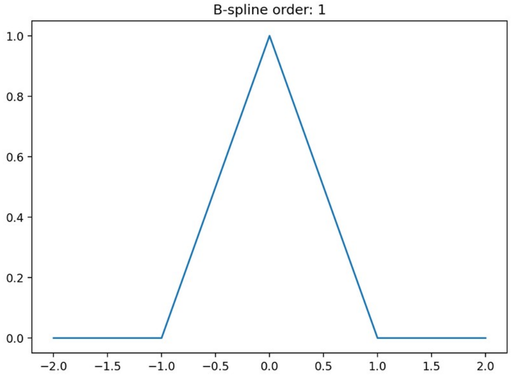
Meet the B-spline family:

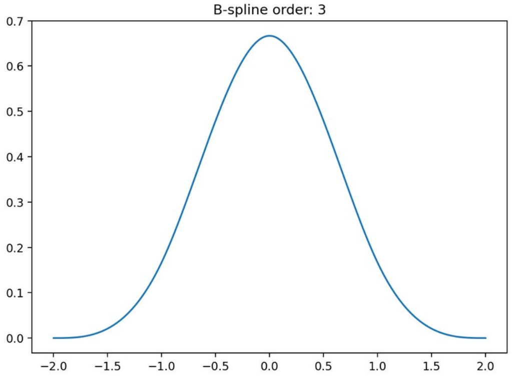
✔ $$\beta^0(x) = \begin{cases} 1, & -\frac{1}{2} < x < \frac{1}{2} \\ \frac{1}{2}, & |x| = \frac{1}{2} \\ 0, & \text{otherwise,} \end{cases}$$



✔ $$\beta^p(x) = \underbrace{\left(\beta^0 * \beta^0 * \cdots * \beta^0\right)}_{(p+1)\text{ times}}(x)$$ where $$\left(f * g\right)(x) = \int_{\tau=-\infty}^{\infty} f(\tau)g(x-\tau)\mathrm{d}\tau$$

"order"



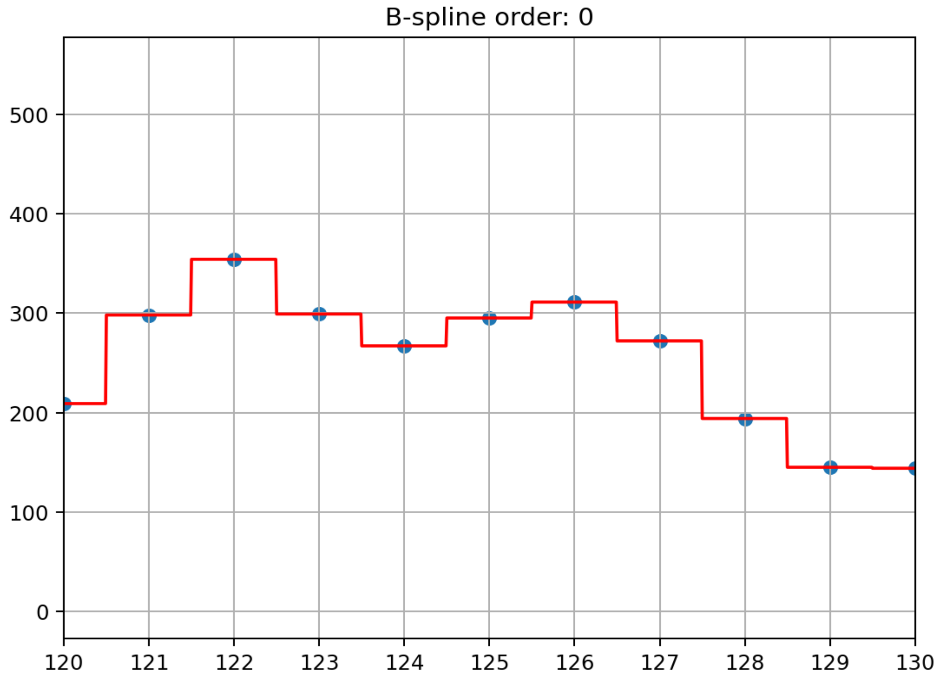


Interpolation

Use $M = N$ **basis functions:** $\phi_m(x) = \beta^p(x - m), \quad m = 0, \ldots, N - 1$

✔ $y(x, \mathbf{w})$ will pass exactly through t_n at the integer locations x_n

Order 0: "nearest neighbor" interpolation
(almost)

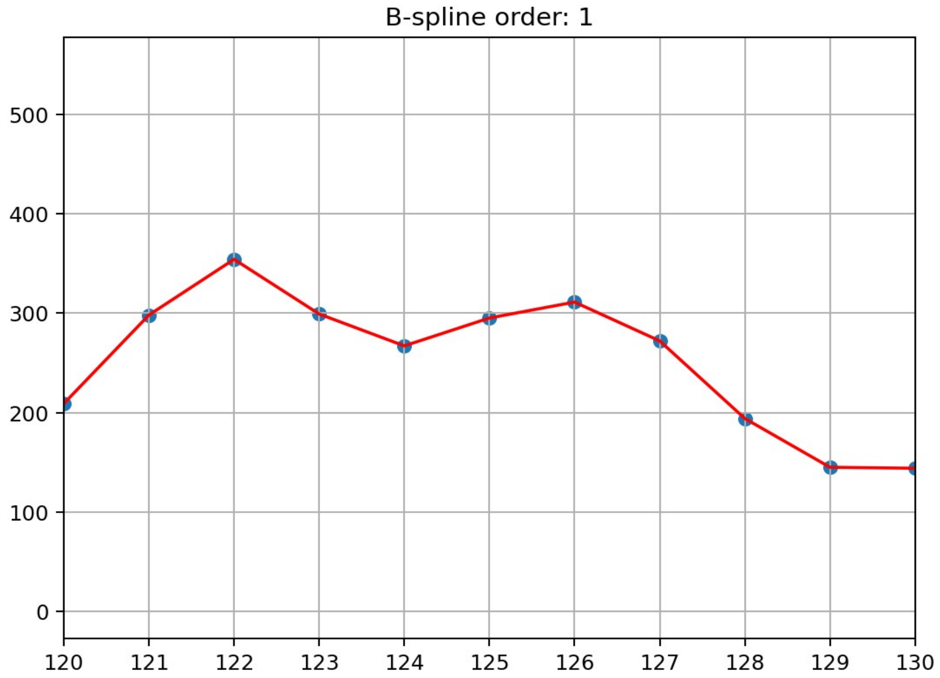


Interpolation

Use $M = N$ **basis functions:** $\phi_m(x) = \beta^p(x - m), \quad m = 0, \ldots, N-1$

✔ $y(x, \mathbf{w})$ will pass exactly through t_n at the integer locations x_n

Order 1: “linear” interpolation

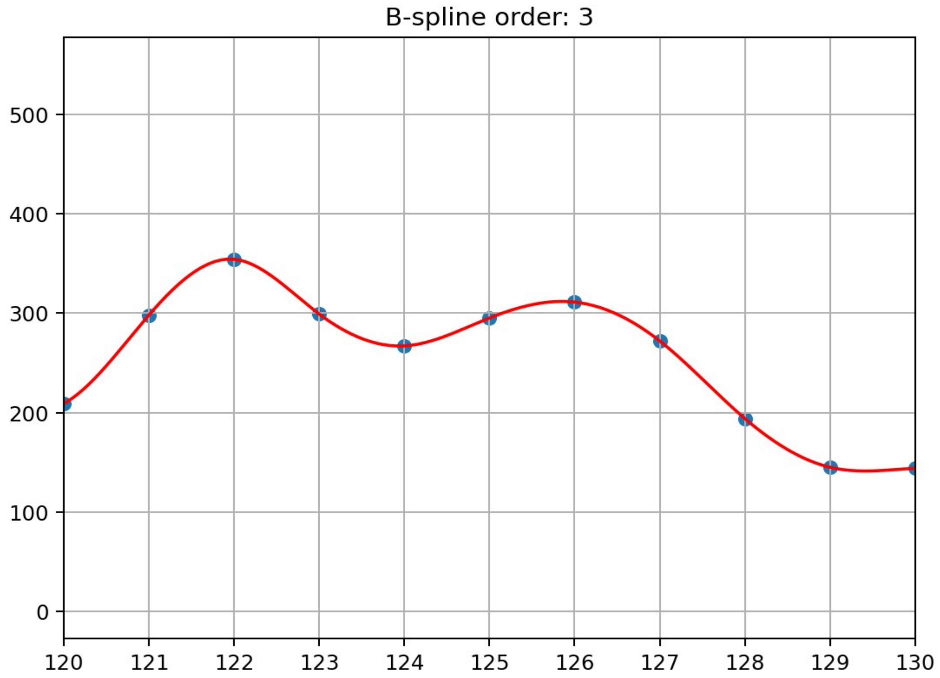


Interpolation

Use $M = N$ **basis functions:** $\phi_m(x) = \beta^p(x-m), \quad m = 0, \ldots, N-1$

✔ $y(x, \mathbf{w})$ will pass exactly through t_n at the integer locations x_n

Order 3: "cubic" interpolation



Going to higher dimensions

Re-arrange pixels in 2D images of size $N_1 \times N_2$ **into 1D signals of length** $N = N_1 N_2$

- ✔ Vectorize the image to be smoothed or interpolated

$$\mathbf{T} = \begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,N_2} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,N_2} \\ \vdots & \vdots & \ddots & \vdots \\ t_{N_1,1} & t_{N_1,2} & \cdots & t_{N_1,N_2} \end{pmatrix} \Longrightarrow \mathbf{t} = \mathrm{vec}(\mathbf{T}) = \begin{pmatrix} t_{1,1} \\ \vdots \\ t_{N_1,1} \\ t_{1,2} \\ \vdots \\ t_{N_1,2} \\ \vdots \\ t_{N_1,N_2} \end{pmatrix}$$

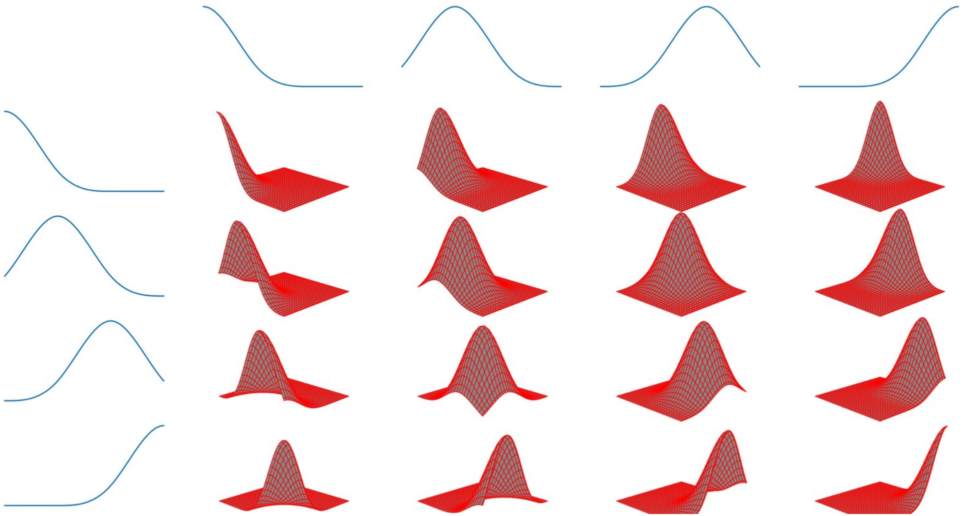
- ✔ Also vectorize each of the 2D basis functions

Solve the resulting 1D problem as before

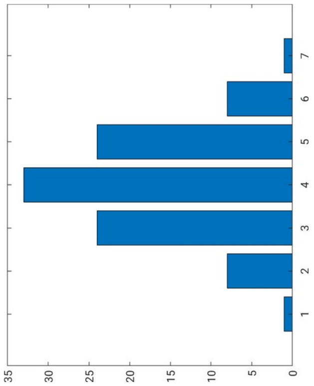
Separable basis functions

Create 2D basis functions from two sets of 1D basis functions

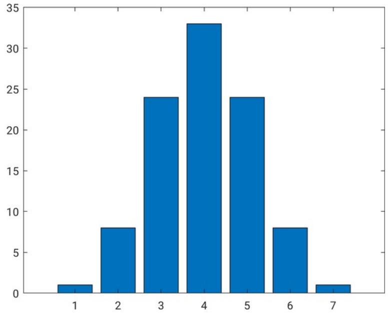
- ✔ Kronecker product: $\mathbf{\Phi} = \mathbf{\Phi}_2 \otimes \mathbf{\Phi}_1$, where $\mathbf{A} \otimes \mathbf{B} = \left(\begin{array}{c|c|c} a_{1,1}\mathbf{B} & a_{2,1}\mathbf{B} & \cdots \\ \hline a_{2,1}\mathbf{B} & a_{2,2}\mathbf{B} & \cdots \\ \hline \vdots & \vdots & \ddots \end{array}\right)$
- ✔ Column $m = m_1 + m_2 M_1$ in $\mathbf{\Phi}$ contains a vectorized version of $\phi_m(\mathbf{x}) = \phi_{m_1}(x_1)\phi_{m_2}(x_2)$



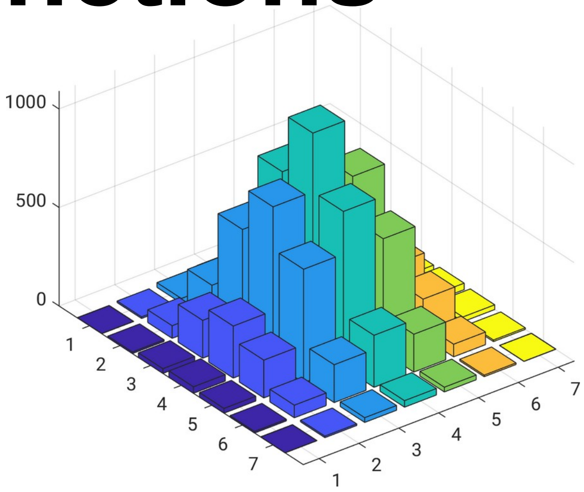
Separable basis functions



x



=



$$\begin{pmatrix} 1 \\ 8 \\ 24 \\ 33 \\ 24 \\ 8 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 8 & 24 & 33 & 24 & 8 & 1 \end{pmatrix} = \begin{pmatrix} 1\cdot 1 & 1\cdot 8 & 1\cdot 24 & 1\cdot 33 & 1\cdot 24 & 1\cdot 8 & 1\cdot 1 \\ 8\cdot 1 & 8\cdot 8 & 8\cdot 24 & 8\cdot 33 & 8\cdot 24 & 8\cdot 8 & 8\cdot 1 \\ 24\cdot 1 & 24\cdot 8 & 24\cdot 24 & 24\cdot 33 & 24\cdot 24 & 24\cdot 8 & 24\cdot 1 \\ 33\cdot 1 & 33\cdot 8 & 33\cdot 24 & 33\cdot 33 & 33\cdot 24 & 33\cdot 8 & 33\cdot 1 \\ 24\cdot 1 & 24\cdot 8 & 24\cdot 24 & 24\cdot 33 & 24\cdot 24 & 24\cdot 8 & 24\cdot 1 \\ 8\cdot 1 & 8\cdot 8 & 8\cdot 24 & 8\cdot 33 & 8\cdot 24 & 8\cdot 8 & 8\cdot 1 \\ 1\cdot 1 & 1\cdot 8 & 1\cdot 24 & 1\cdot 33 & 1\cdot 24 & 1\cdot 8 & 1\cdot 1 \end{pmatrix}$$

Exploiting separability

- ✔ When $\mathbf{\Phi} = \mathbf{\Phi}_2 \otimes \mathbf{\Phi}_1$, we can compute $\mathbf{c} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$ faster as follows:

$$\mathbf{C} = \mathbf{\Phi}_1^{\mathrm{T}}\mathbf{T}\mathbf{\Phi}_2 \quad \text{where} \quad \mathrm{vec}(\mathbf{C}) = \mathbf{c}$$

- ✔ As a result, we can also compute $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$ as:

$$\mathbf{W} = \left(\mathbf{\Phi}_1^{\mathrm{T}}\mathbf{\Phi}_1\right)^{-1}\mathbf{\Phi}_1^{\mathrm{T}}\mathbf{T}\mathbf{\Phi}_2\left(\mathbf{\Phi}_2^{\mathrm{T}}\mathbf{\Phi}_2\right)^{-1} \quad \text{where} \quad \mathrm{vec}(\mathbf{W}) = \mathbf{w}$$

Exploiting separability

- ✔ When $\mathbf{\Phi} = \mathbf{\Phi}_2 \otimes \mathbf{\Phi}_1$, we can compute $\mathbf{c} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$ faster as follows:

$$\mathbf{C} = \mathbf{\Phi}_1^{\mathrm{T}}\mathbf{T}\mathbf{\Phi}_2 \quad \text{where} \quad \mathrm{vec}(\mathbf{C}) = \mathbf{c}$$

- ✔ As a result, we can also compute $\mathbf{w} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$ as:

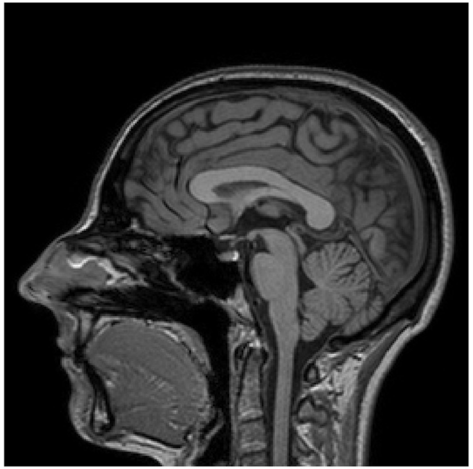
$$\mathbf{W} = \left(\mathbf{\Phi}_1^{\mathrm{T}}\mathbf{\Phi}_1\right)^{-1}\mathbf{\Phi}_1^{\mathrm{T}}\mathbf{T}\mathbf{\Phi}_2\left(\mathbf{\Phi}_2^{\mathrm{T}}\mathbf{\Phi}_2\right)^{-1} \quad \text{where} \quad \mathrm{vec}(\mathbf{W}) = \mathbf{w}$$

Example: naively interpolating a 256x256 image (M = 256*256 = 65,536 basis functions in 2D!)

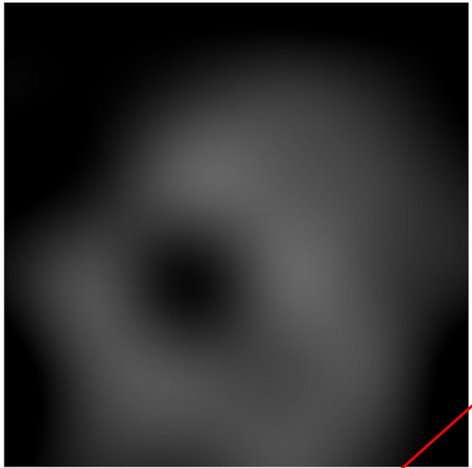
Storing $\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}$ takes 32 GB, vs. 1MB to store both $\mathbf{\Phi}_1^{\mathrm{T}}\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2^{\mathrm{T}}\mathbf{\Phi}_2$

Inverting $\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}$ is almost <u>**10 million times slower**</u> than inverting both $\mathbf{\Phi}_1^{\mathrm{T}}\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2^{\mathrm{T}}\mathbf{\Phi}_2$

Smoothing in 2D



$\hat{\mathrm{T}} = \mathrm{S}$



$$\hat{\mathbf{T}} = \mathbf{S}_1 \mathbf{T} \mathbf{S}_2^{\mathrm{T}}$$

smoothing in the **column-direction**

$$\mathbf{S}_1 = \boldsymbol{\Phi}_1 \left(\boldsymbol{\Phi}_1^{\mathrm{T}} \boldsymbol{\Phi}_1 \right)^{-1} \boldsymbol{\Phi}_1^{\mathrm{T}}$$

smoothing in the **row-direction**

Aalto-yliopisto
Aalto-universitetet
Aalto University

Interpolation in 2D