University of Amsterdam
Informatics Institute

AMLAB
Amsterdam
Machine Learning Lab

Machine Learning 1

Lecture 12.2 - Kernel Methods
Gaussian Processes - Kernelizing Bayesian Linear Regression

Erik Bekkers

(Bishop 6.4.0, 6.4.1)

Slide credits: Patrick Forré and Rianne van den Berg

Image credit: Kirillm | Getty Images

Revisit Bayesian linear regression

- Given input $\mathbf{x}$ and target t, we assumed

$$t = \boldsymbol{\phi}(\mathbf{x})^T\mathbf{w} + \varepsilon \quad \text{with} \quad \varepsilon \sim \mathcal{N}(0,\beta^{-1})$$

- So for observations $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix}$ and $\mathbf{t} = \begin{bmatrix} t_1 \\ \vdots \\ t_N \end{bmatrix}$ we have

Likelihood !

$$\mathbf{t} = \boldsymbol{\Phi}\mathbf{w} + \boldsymbol{\varepsilon}\,, \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \beta^{-1}\mathbf{I})\,, \qquad p(\mathbf{t}\,|\,\mathbf{X}, \mathbf{w}) = \mathcal{N}\left(\mathbf{t}\,|\,\boldsymbol{\Phi}\mathbf{w}, \beta^{-1}\mathbf{I}\right)$$

- We imposed a prior on $\mathbf{w}$:

$$p(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}\,|\,\mathbf{0}, \boldsymbol{\Sigma}_p\right)$$

- After observing X and t, we obtained posterior

$$p(\mathbf{w}\,|\,\mathbf{X}, \mathbf{t}) = \frac{p(\mathbf{t}\,|\,\mathbf{X}, \mathbf{w})p(\mathbf{w})}{p(\mathbf{t}\,|\,\mathbf{X})} = \mathcal{N}\left(\mathbf{w}\,|\,\mathbf{m}_N, \boldsymbol{S}_N\right)$$

with $\mathbf{m}_N = \beta\mathbf{S}_N^{-1}\boldsymbol{\Phi}^T\mathbf{t}\,, \qquad \mathbf{S}_N^{-1} = \beta\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \boldsymbol{\Sigma}_p^{-1}$

Revisit Bayesian linear regression

- Predictions for new point $\mathbf{x}^*$, after observing N data points

$$p(t^* \mid \mathbf{x}^*, \mathbf{X}, \mathbf{t}) = \int p(t^* \mid \mathbf{x}^*, \mathbf{w}) p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) \, \mathrm{d}\mathbf{w}$$

$$= \mathcal{N}\left(t^* \mid \mu_N(\mathbf{x}^*), \sigma_N^2(\mathbf{x}^*)\right)$$

with

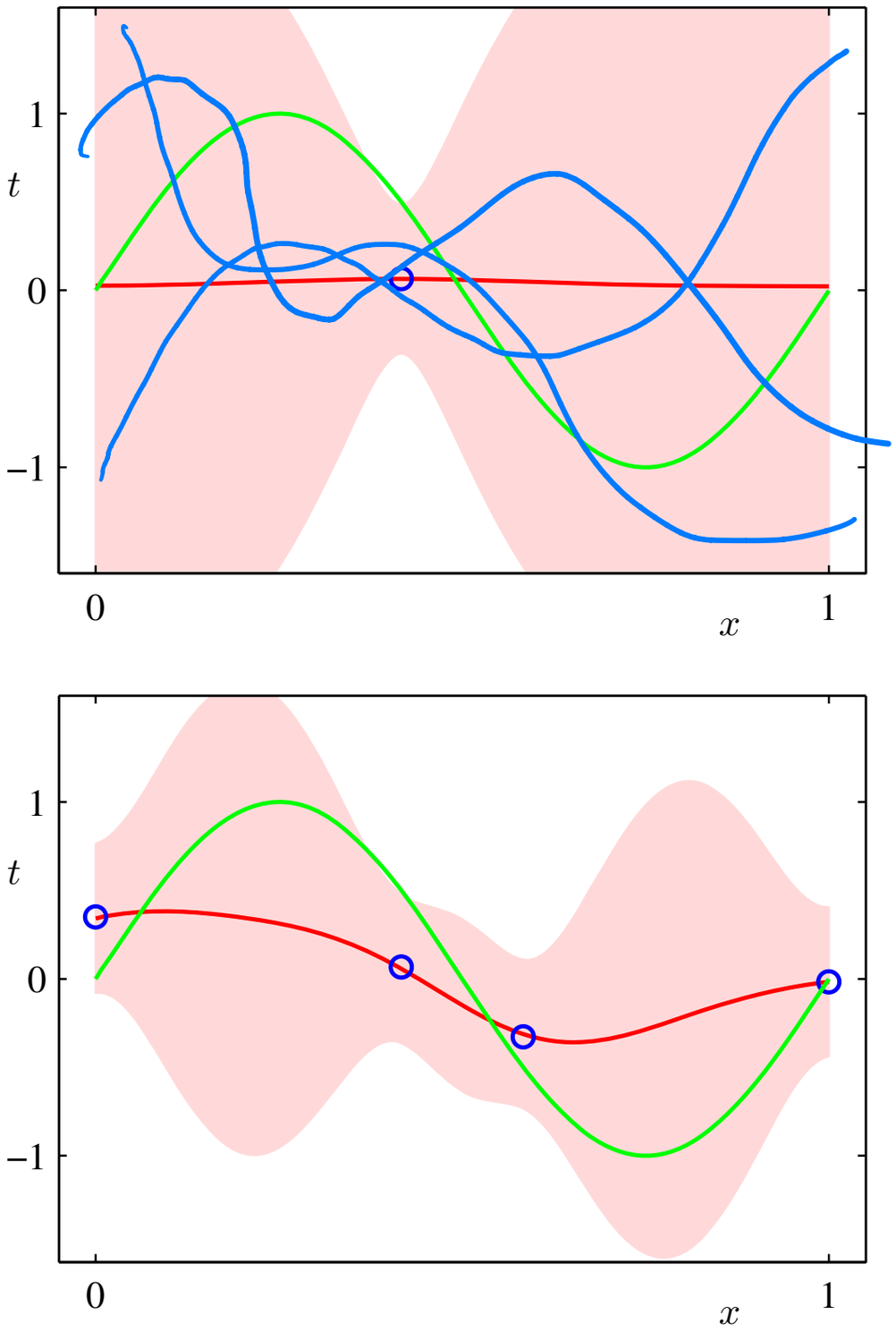
$$\mu_N(\mathbf{x}^*) = \boldsymbol{\phi}(\mathbf{x}^*)^T \mathbf{m}_N = \beta \boldsymbol{\phi}(\mathbf{x}^*)^T \mathbf{S}_N^{-1} \boldsymbol{\Phi}^T \mathbf{t} = \sum_{n=1}^{N} \beta \boldsymbol{\phi}(\mathbf{x}^*)^T \mathbf{S}_N^{-1} \boldsymbol{\phi}(\mathbf{x}_n) t_n$$

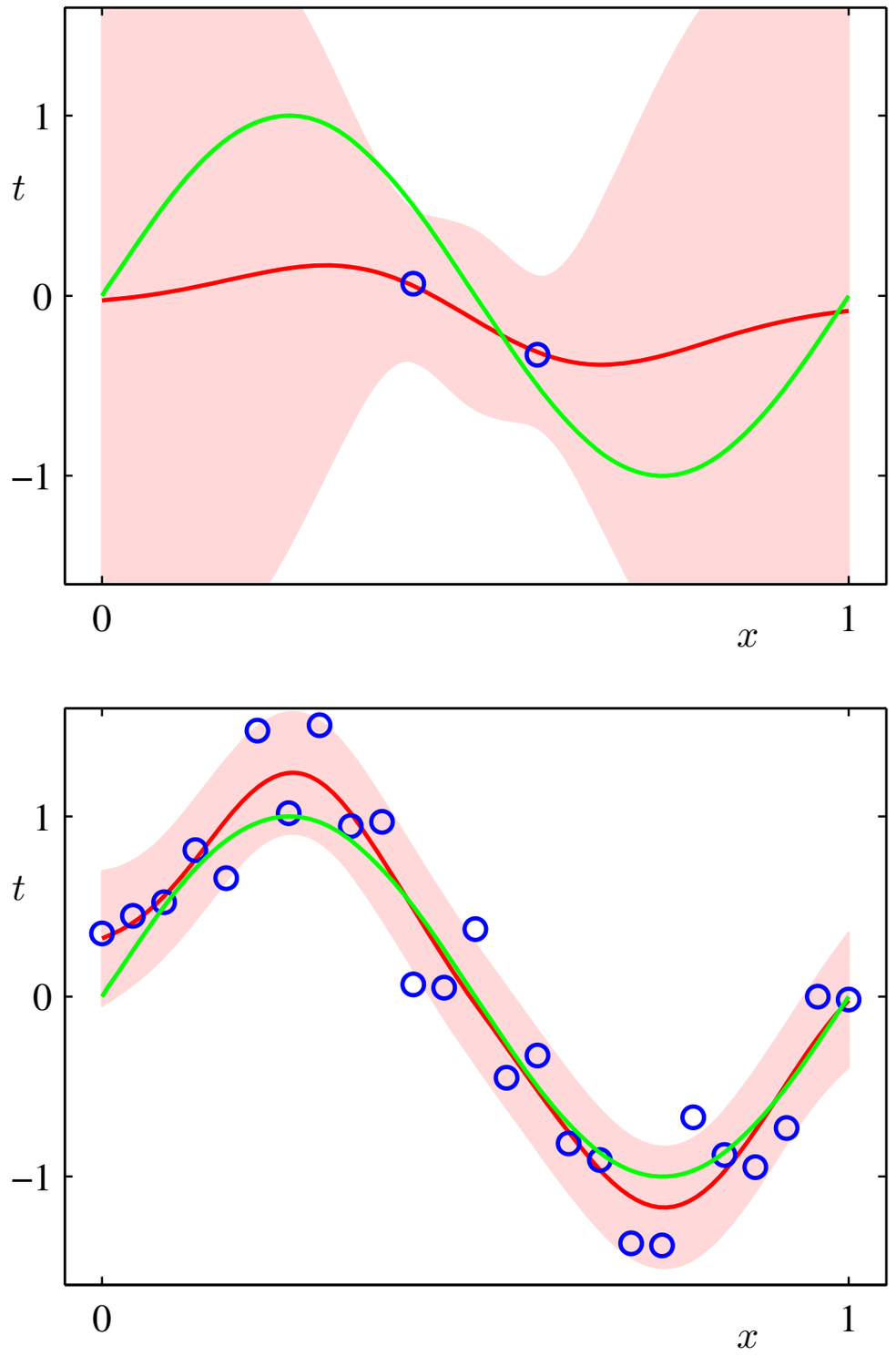
$$= \sum_{n=1}^{N} k(\mathbf{x}^*, \mathbf{x}_n) t_n$$

$$\sigma_N^2(\mathbf{x}^*) = \beta^{-1} + \boldsymbol{\phi}(\mathbf{x}^*)^T \mathbf{S}_N \boldsymbol{\phi}(\mathbf{x}^*)$$

- Mean predictions are linear combinations of kernels!

Revisit Bayesian linear regression





Revisit Bayesian linear regression

- Linear Bayesian regression suffers from limited expressiveness (linear in $\mathbf{w}$)
- Can become more expressive for larger M, where $\boldsymbol{\phi}(\mathbf{x}) \in \mathbb{R}^M$
- However, this also requires more parameters! ($\mathbf{w} \in \mathbb{R}^M$)
- It's expensive to train more expressive models! (requires inverting $\mathbf{S}_N$ which is of size $M \times M$)
- We also need to choose our basis functions at 'good' locations
- But we saw that the final mean prediction can be written in equivalent kernel form…