

RWTH AACHEN  
UNIVERSITY

# Advanced Machine Learning Lecture 5

## Gaussian Processes 2

07.11.2016

Bastian Leibe  
RWTH Aachen  
<http://www.vision.rwth-aachen.de/>  
leibe@vision.rwth-aachen.de

Advanced Machine Learning Winter'16

RWTH AACHEN  
UNIVERSITY

## This Lecture: Advanced Machine Learning

- Regression Approaches
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Kernels (Kernel Ridge Regression)
  - Gaussian Processes
- Learning with Latent Variables
  - EM and Generalizations
  - Approximate Inference
- Deep Learning
  - Neural Networks
  - CNNs, RNNs, RBMs, etc.

B. Leibe

RWTH AACHEN  
UNIVERSITY

## Topics of This Lecture

- Kernels
  - Recap: Kernel trick
  - Constructing kernels
- Gaussian Processes
  - Recap: Definition
  - Prediction with noise-free observations
  - Prediction with noisy observations
  - GP Regression
  - Influence of hyperparameters
- Learning Gaussian Processes
  - Bayesian Model Selection
  - Model selection for Gaussian Processes
- Applications

B. Leibe

3

Advanced Machine Learning Winter'16

RWTH AACHEN  
UNIVERSITY

## Recap: Kernel Ridge Regression

- Dual definition
  - Instead of working with  $w$ , substitute  $w = \Phi^T a$  into  $J(w)$  and write the result using the **kernel matrix**  $K = \Phi \Phi^T$ :
 
$$J(a) = \frac{1}{2} a^T K K a - a^T K t + \frac{1}{2} t^T t + \frac{\lambda}{2} a^T K a$$
  - Solving for  $a$ , we obtain
 
$$a = (K + \lambda I_N)^{-1} t$$
- Prediction for a new input  $x$ :
  - Writing  $k(x)$  for the vector with elements  $k_n(x) = k(x_n, x)$ 

$$y(x) = w^T \phi(x) = a^T \Phi \phi(x) = k(x)^T (K + \lambda I_N)^{-1} t$$

$\Rightarrow$  The dual formulation allows the solution to be entirely expressed in terms of the kernel function  $k(x, x')$ .

B. Leibe

4

Advanced Machine Learning Winter'16

RWTH AACHEN  
UNIVERSITY

## Recap: Properties of Kernels

- Theorem
  - Let  $k: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  be a **positive definite kernel function**. Then there exists a **Hilbert Space**  $\mathcal{H}$  and a mapping  $\phi: \mathcal{X} \rightarrow \mathcal{H}$  such that
 
$$k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}$$
  - where  $\langle \cdot, \cdot \rangle_{\mathcal{H}}$  is the inner product in  $\mathcal{H}$ .
- Translation
  - Take **any** set  $\mathcal{X}$  and **any** function  $k: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ .
  - If  $k$  is a positive definite kernel, then we can use  $k$  to learn a classifier for the elements in  $\mathcal{X}$ !
- Note
  - $\mathcal{X}$  can be any set, e.g.  $\mathcal{X} =$  "all videos on YouTube" or  $\mathcal{X} =$  "all permutations of  $\{1, \dots, k\}$ ", or  $\mathcal{X} =$  "the internet".

B. Leibe

5

Advanced Machine Learning Winter'16

RWTH AACHEN  
UNIVERSITY

## Recap: The "Kernel Trick"

Any algorithm that uses data only in the form of inner products can be **kernelized**.

- How to kernelize an algorithm
  - Write the algorithm only in terms of inner products.
  - Replace all inner products by kernel function evaluations.

$\Rightarrow$  The resulting algorithm will do the same as the linear version, but in the (hidden) feature space  $\mathcal{H}$ .

- Caveat: working in  $\mathcal{H}$  is not a guarantee for better performance. A good choice of  $k$  and model selection are important!

B. Leibe

6

Advanced Machine Learning Winter'16

RWTH AACHEN UNIVERSITY

## Topics of This Lecture

- **Kernels**
  - Recap: Kernel trick
  - Constructing kernels
- **Gaussian Processes**
  - Recap: Definition
  - Prediction with noise-free observations
  - Prediction with noisy observations
  - GP Regression
  - Influence of hyperparameters
- **Learning Gaussian Processes**
  - Bayesian Model Selection
  - Model selection for Gaussian Processes
- **Applications**

13

RWTH AACHEN UNIVERSITY

## Recap: Gaussian Process

- **Gaussian distribution**
  - Probability distribution over scalars / vectors.
- **Gaussian Process (generalization of Gaussian distrib.)**
  - Describes properties of functions.
  - **Function:** Think of a function as a long vector where each entry specifies the function value  $f(x_i)$  at a particular point  $x_i$ .
  - **Issue:** How to deal with infinite number of points?
    - If you ask only for properties of the function at a finite number of points...
    - Then inference in Gaussian Process gives you the same answer if you ignore the infinitely many other points.
- **Definition**
  - A **Gaussian Process (GP)** is a collection of random variables any finite number of which has a joint Gaussian distribution.

14

RWTH AACHEN UNIVERSITY

## Recap: Gaussian Process

- **A Gaussian Process is completely defined by**
  - **Mean function**  $m(x)$  and
$$m(x) = \mathbb{E}[f(x)]$$
  - **Covariance function**  $k(x, x')$ 

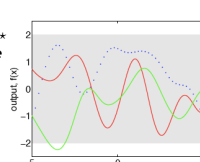
$$k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x')))]$$
  - We write the Gaussian Process (GP)
$$f(x) \sim \mathcal{GP}(m(x), k(x, x'))$$

15

RWTH AACHEN UNIVERSITY

## Recap: GPs Define Prior over Functions

- **Distribution over functions:**
  - Specification of covariance function implies distribution over functions.
  - I.e. we can draw samples from the distribution of functions evaluated at a (finite) number of points.
  - **Procedure**
    - We choose a number of input points  $X_*$
    - We write the corresponding covariance matrix (e.g. using SE) element-wise:
$$K(X_*, X_*)$$
    - Then we generate a random Gaussian vector with this covariance matrix:
$$f_* \sim \mathcal{N}(0, K(X_*, X_*))$$



16

RWTH AACHEN UNIVERSITY

## Topics of This Lecture

- **Kernels**
  - Recap: Kernel trick
  - Constructing kernels
- **Gaussian Processes**
  - Recap: Definition
  - Prediction with noise-free observations
  - Prediction with noisy observations
  - GP Regression
  - Influence of hyperparameters
- **Learning Gaussian Processes**
  - Bayesian Model Selection
  - Model selection for Gaussian Processes
- **Applications**

17

RWTH AACHEN UNIVERSITY

## Prediction with Noise-free Observations

- **Assume our observations are noise-free:**

$$\{(x_n, f_n) \mid n = 1, \dots, N\}$$
- **Joint distribution of the training outputs  $f$  and test outputs  $f_*$  according to the prior:**

$$\begin{bmatrix} f \\ f_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right)$$
  - $K(X, X_*)$  contains covariances for all pairs of training and test points.
- **To get the posterior (after including the observations)**
  - We need to restrict the above prior to contain only those functions which agree with the observed values.
  - Think of generating functions from the prior and rejecting those that disagree with the observations (obviously prohibitive).

19

RWTH AACHEN UNIVERSITY

## Prediction with Noise-free Observations

- Calculation of posterior: simple in GP framework
  - Corresponds to conditioning the joint Gaussian prior distribution on the observations:
 
$$\mathbf{f}_* | X_*, X, \mathbf{f} \sim \mathcal{N}(\bar{\mathbf{f}}_*, \text{cov}[\mathbf{f}_*]) \quad \bar{\mathbf{f}}_* = \mathbb{E}[\mathbf{f}_* | X, X_*, \mathbf{f}]$$
- with:
 
$$\bar{\mathbf{f}}_* = K(X_*, X)K(X, X)^{-1}\mathbf{f}$$

$$\text{cov}[\mathbf{f}_*] = K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X_*)$$
- This uses the general property of Gaussians that
 
$$\boldsymbol{\mu} = \begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \boldsymbol{\Sigma} = \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix} \Rightarrow \begin{aligned} \mu_{a|b} &= \mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(\mathbf{x}_b - \mu_b) \\ \Sigma_{a|b} &= \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba} \end{aligned}$$

Slide credit: Bernt Schiele B. Leibe 20

RWTH AACHEN UNIVERSITY

## Prediction with Noise-free Observations

- Example:

Prior

Posterior using 5 noise-free observations

Slide credit: Bernt Schiele B. Leibe Image source: Rasmussen & Williams, 2006 21

RWTH AACHEN UNIVERSITY

## Topics of This Lecture

- Kernels
  - Recap: Kernel trick
  - Constructing kernels
- Gaussian Processes
  - Recap: Definition
  - Prediction with noise-free observations
  - Prediction with noisy observations
  - GP Regression
  - Influence of hyperparameters
- Learning Gaussian Processes
  - Bayesian Model Selection
  - Model selection for Gaussian Processes
- Applications

Slide credit: Bernt Schiele B. Leibe 22

RWTH AACHEN UNIVERSITY

## Prediction with Noisy Observations

- Typically, we assume noise in the observations
 
$$t = f(\mathbf{x}) + \epsilon \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$
- The prior on the noisy observations becomes
 
$$\text{cov}[y_p, y_q] = k(\mathbf{x}_p, \mathbf{x}_q) + \sigma_n^2 \delta_{pq}$$
  - Written in compact form:
 
$$\text{cov}[\mathbf{y}] = K(X, X) + \sigma_n^2 I$$
- Joint distribution of the observed values and the test locations under the prior is then:
 
$$\begin{bmatrix} \mathbf{t} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right)$$

Slide credit: Bernt Schiele B. Leibe 24

RWTH AACHEN UNIVERSITY

## Prediction with Noisy Observations

- Calculation of posterior:
  - Corresponds to conditioning the joint Gaussian prior distribution on the observations:
 
$$\mathbf{f}_* | X_*, X, \mathbf{t} \sim \mathcal{N}(\bar{\mathbf{f}}_*, \text{cov}[\mathbf{f}_*]) \quad \bar{\mathbf{f}}_* = \mathbb{E}[\mathbf{f}_* | X, X_*, \mathbf{t}]$$
- with:
 
$$\bar{\mathbf{f}}_* = K(X_*, X) (K(X, X) + \sigma_n^2 I)^{-1} \mathbf{t}$$

$$\text{cov}[\mathbf{f}_*] = K(X_*, X_*) - K(X_*, X) (K(X, X) + \sigma_n^2 I)^{-1} K(X, X_*)$$
- ⇒ This is the key result that defines Gaussian process regression!
  - The predictive distribution is a Gaussian whose mean and variance depend on the test points  $X_*$  and on the kernel  $k(\mathbf{x}, \mathbf{x}')$ , evaluated on the training data  $X$ .

Slide credit: Bernt Schiele B. Leibe 25

RWTH AACHEN UNIVERSITY

## Gaussian Process Regression

- Example

Slide credit: Bernt Schiele B. Leibe 26

RWTH AACHEN UNIVERSITY

## Gaussian Process Regression

Slide credit: Bernt Schiele      B. Leibe      27

RWTH AACHEN UNIVERSITY

## Discussion

- Key result:**  $f_* | X_*, X, t \sim \mathcal{N}(\bar{f}_*, \text{cov}[f_*])$  with
 
$$\bar{f}_* = K(X_*, X) (K(X, X) + \sigma_n^2 I)^{-1} t$$

$$\text{cov}[f_*] = K(X_*, X_*) - K(X_*, X) (K(X, X) + \sigma_n^2 I)^{-1} K(X, X_*)$$
- Observations**
  - The mean can be written in linear form
 
$$\bar{f}(x_*) = k(x_*, X) \underbrace{[K(X, X) + \sigma_n^2 I]^{-1} t}_{\alpha} = \sum_{n=1}^N \alpha_n k(x_*, x_n).$$
 - This form is commonly encountered in the kernel literature ( $\rightarrow$  SVM)
  - The variance is the difference between two terms
 
$$V(x_*) = \underbrace{k(x_*, x_*)}_{\text{Prior variance}} - \underbrace{k(x_*, X) [K(X, X) + \sigma_n^2 I]^{-1} k(X, x_*)}_{\text{Explanation of data } X}$$

Slide adapted from Carl Rasmussen      B. Leibe      28

RWTH AACHEN UNIVERSITY

## Computational Complexity

- Computational complexity**
  - Central operation in using GPs involves **inverting a matrix of size  $N \times N$**  (the kernel matrix  $K(X, X)$ ):
 
$$\bar{f}_* = K(X_*, X) (K(X, X) + \sigma_n^2 I)^{-1} t$$

$$\text{cov}[f_*] = K(X_*, X_*) - K(X_*, X) (K(X, X) + \sigma_n^2 I)^{-1} K(X, X_*)$$
 $\Rightarrow$  Effort in  $\mathcal{O}(N^3)$  for  $N$  data points!
  - Compare this with the basis function model ( $\rightarrow$  Lecture 3)
 
$$p(f_* | x_*, X, t) \sim \mathcal{N}\left(\frac{1}{\sigma_n^2} \phi(x_*)^T S^{-1} \Phi(X) t, \phi(x_*)^T S^{-1} \phi(x_*)\right)$$

$$S = \frac{1}{\sigma_n^2} \Phi(X) \Phi(X)^T + \Sigma_p^{-1}$$
 $\Rightarrow$  Effort in  $\mathcal{O}(M^3)$  for  $M$  basis functions.

B. Leibe      30

RWTH AACHEN UNIVERSITY

## Computational Complexity

- Complexity of GP model**
  - Training effort:  $\mathcal{O}(N^3)$  through matrix inversion
  - Test effort:  $\mathcal{O}(N^2)$  through vector-matrix multiplication
- Complexity of basis function model**
  - Training effort:  $\mathcal{O}(M^3)$
  - Test effort:  $\mathcal{O}(M^2)$
- Discussion**
  - If the number of basis functions  $M$  is smaller than the number of data points  $N$ , then the basis function model is more efficient.
  - However, advantage of GP viewpoint is that we can consider covariance functions that can only be expressed by an **infinite number of basis functions**.
  - Still, exact GP methods become infeasible for large training sets.

B. Leibe      31

RWTH AACHEN UNIVERSITY

## GP Regression Algorithm

- Very simple algorithm!**

input:  $X$  (inputs),  $y$  (targets),  $k$  (covariance function),  $\sigma_n^2$  (noise level),  $x_*$  (test input)

2:  $L := \text{cholesky}(K + \sigma_n^2 I)$

$\alpha := L^{-T} (L \backslash y)$  } predictive mean eq. (2.25)

4:  $\bar{f}_* := k_*^T \alpha$

$v := L \backslash k_*$  } predictive variance eq. (2.26)

6:  $\mathbb{V}[f_*] := k(x_*, x_*) - v^T v$

$\log p(y|X) := -\frac{1}{2} y^T \alpha - \sum_i \log L_{ii} - \frac{n}{2} \log 2\pi$  eq. (2.30)

8: **return:**  $\bar{f}_*$  (mean),  $\mathbb{V}[f_*]$  (variance),  $\log p(y|X)$  (log marginal likelihood)

  - Based on the following equations (Matrix inv.  $\leftrightarrow$  Cholesky fact.)
 
$$\bar{f}_* = k_*^T (K + \sigma_n^2 I)^{-1} t$$

$$\text{cov}[f_*] = k(x_*, x_*) - k_*^T (K + \sigma_n^2 I)^{-1} k_*$$

$$\log p(t|X) = -\frac{1}{2} t^T (K + \sigma_n^2 I)^{-1} t - \frac{1}{2} \log |K + \sigma_n^2 I| - \frac{N}{2} \log 2\pi$$

B. Leibe      Image source: Rasmussen & Williams, 2006      32

RWTH AACHEN UNIVERSITY

## Influence of Hyperparameters

- Most covariance functions have some free parameters.**
  - Example:**

$$k_y(x_p, x_q) = \sigma_f^2 \exp\left\{-\frac{(x_p - x_q)^2}{2 \cdot l^2}\right\} + \sigma_n^2 \delta_{pq}$$
  - Parameters:**  $(l, \sigma_f, \sigma_n)$ 
    - Signal variance:**  $\sigma_f^2$
    - Range of neighbor influence** (called "length scale"):  $l$
    - Observation noise:**  $\sigma_n^2$

Slide credit: Bernt Schiele      B. Leibe      33

RWTH AACHEN UNIVERSITY

## Influence of Hyperparameters

$$k_y(\mathbf{x}_p, \mathbf{x}_q) = \sigma_f^2 \exp \left\{ -\frac{(\mathbf{x}_p - \mathbf{x}_q)^2}{2 \cdot l^2} \right\} + \sigma_n^2 \delta_{pq}$$

- Examples for different settings of the length scale
  - $(1, \sigma_f, \sigma_n) =$  ( $\sigma$  parameters set by optimizing the marginal likelihood)
  - $(0.3, 1.08, 0.00005)$
  - $(1, 1, 0.1)$
  - $(3.0, 1.16, 0.89)$

34

Slide credit: Bernt Schiele      B. Leibe      Image source: Rasmussen & Williams, 2006

RWTH AACHEN UNIVERSITY

## Topics of This Lecture

- Kernels
  - Recap: Kernel trick
  - Constructing kernels
- Gaussian Processes
  - Recap: Definition
  - Prediction with noise-free observations
  - Prediction with noisy observations
  - GP Regression
  - Influence of hyperparameters
- Learning Gaussian Processes
  - Bayesian Model Selection
  - Model selection for Gaussian Processes
- Applications

35

B. Leibe

RWTH AACHEN UNIVERSITY

## Learning Kernel Parameters

- Can we determine the length scale and noise levels from training data?

36

Slide credit: Bernt Schiele      B. Leibe

RWTH AACHEN UNIVERSITY

## Bayesian Model Selection

- Goal
  - Determine/learn different parameters of Gaussian Processes
- Hierarchy of parameters
  - Lowest level
    - w - e.g. parameters of a linear model.
  - Mid-level (hyperparameters)
    - $\theta$  - e.g. controlling prior distribution of w.
  - Top level
    - Typically discrete set of model structures  $\mathcal{H}_i$ .
- Approach
  - Inference takes place one level at a time.

37

Slide credit: Bernt Schiele      B. Leibe

RWTH AACHEN UNIVERSITY

## Model Selection at Lowest Level

- Posterior of the parameters  $\mathbf{w}$  is given by Bayes' rule
 
$$p(\mathbf{w} | \mathbf{t}, X, \theta, \mathcal{H}_i) = \frac{p(\mathbf{t} | X, \mathbf{w}, \theta, \mathcal{H}_i) p(\mathbf{w} | \theta, X, \mathcal{H}_i)}{p(\mathbf{t} | X, \theta, \mathcal{H}_i)}$$

$$= \frac{p(\mathbf{t} | X, \mathbf{w}, \mathcal{H}_i) p(\mathbf{w} | \theta, \mathcal{H}_i)}{p(\mathbf{t} | X, \theta, \mathcal{H}_i)}$$
- with
  - $p(\mathbf{t} | X, \mathbf{w}, \mathcal{H}_i)$  likelihood and
  - $p(\mathbf{w} | \theta, \mathcal{H}_i)$  prior parameters  $\mathbf{w}$ ,
  - Denominator (normalizing constant) is independent of the parameters and is called **marginal likelihood**.
$$p(\mathbf{t} | X, \theta, \mathcal{H}_i) = \int p(\mathbf{t} | X, \mathbf{w}, \mathcal{H}_i) p(\mathbf{w} | \theta, \mathcal{H}_i) d\mathbf{w}$$

38

Slide credit: Bernt Schiele      B. Leibe

RWTH AACHEN UNIVERSITY

## Model Selection at Mid Level

- Posterior of parameters  $\theta$  is again given by Bayes' rule
 
$$p(\theta | \mathbf{t}, X, \mathcal{H}_i) = \frac{p(\mathbf{t} | X, \theta, \mathcal{H}_i) p(\theta | X, \mathcal{H}_i)}{p(\mathbf{t} | X, \mathcal{H}_i)}$$

$$= \frac{p(\mathbf{t} | X, \theta, \mathcal{H}_i) p(\theta | \mathcal{H}_i)}{p(\mathbf{t} | X, \mathcal{H}_i)}$$
- where
  - The marginal likelihood of the previous level  $p(\mathbf{t} | X, \theta, \mathcal{H}_i)$  plays the role of the likelihood of this level.
  - $p(\theta | \mathcal{H}_i)$  is the **hyperprior** (prior of the hyperparameters)
  - Denominator (normalizing constant) is given by:
 
$$p(\mathbf{t} | X, \mathcal{H}_i) = \int p(\mathbf{t} | X, \theta, \mathcal{H}_i) p(\theta | \mathcal{H}_i) d\theta$$
 which is again a **marginal likelihood** (at the mid level).

39

Slide credit: Bernt Schiele      B. Leibe

RWTH AACHEN UNIVERSITY

## Model Selection at Top Level

- At the top level, we calculate the posterior of the model
 
$$p(\mathcal{H}_i | \mathbf{t}, X) = \frac{p(\mathbf{t} | X, \mathcal{H}_i) p(\mathcal{H}_i)}{p(\mathbf{t} | X)}$$
- where
  - Again, the denominator of the previous level  $p(\mathbf{t} | X, \mathcal{H}_i)$  plays the role of the likelihood.
  - $p(\mathcal{H}_i)$  is the prior of the model structure.
  - Denominator (normalizing constant) is given by:
 
$$p(\mathbf{t} | X) = \sum_i p(\mathbf{t} | X, \mathcal{H}_i) p(\mathcal{H}_i)$$

Slide credit: Bernt Schiele B. Leibe 40

RWTH AACHEN UNIVERSITY

## Bayesian Model Selection

- Discussion
  - Marginal likelihood is main difference to non-Bayesian methods
  - It automatically incorporates a trade-off between the model fit and the model complexity:
    - A simple model can only account for a limited range of possible sets of target values - if a simple model fits well, it obtains a high posterior.
    - A complex model can account for a large range of possible sets of target values - therefore, it can never attain a very high posterior.

Slide credit: Bernt Schiele B. Leibe Image source: Rasmussen & Williams, 2004 41

RWTH AACHEN UNIVERSITY

## Bayesian Model Selection

- Computational issues
  - Requires the evaluation of several integrals, which may or may not be analytically tractable, depending on details of the models.
  - In general, one may have to resort to analytic approximations or MCMC methods. (→Lecture 7)
- Model selection for GP regression
  - GP regression models with Gaussian noise are an (important) exception:
    - Integrals over the parameters are analytically tractable and
    - At the same time, the models are flexible.

Slide credit: Bernt Schiele B. Leibe 42

RWTH AACHEN UNIVERSITY

## Example

Slide credit: Bernt Schiele B. Leibe 43

RWTH AACHEN UNIVERSITY

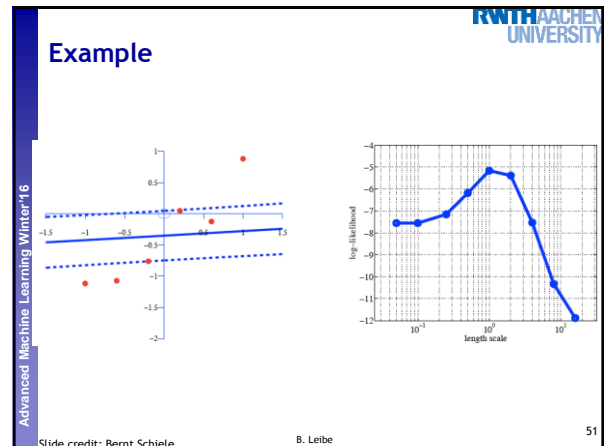
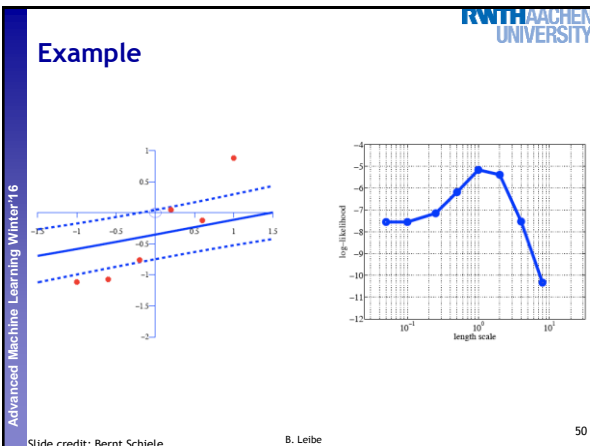
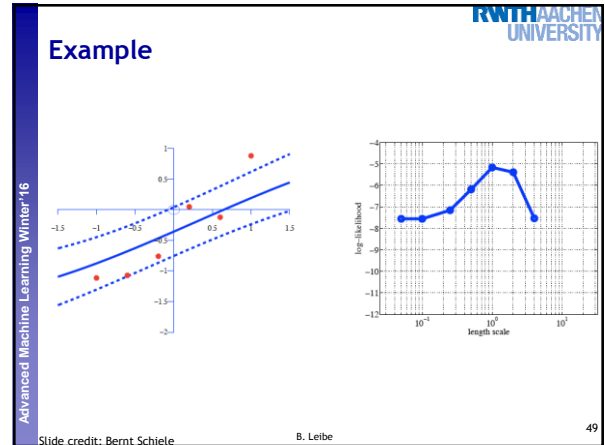
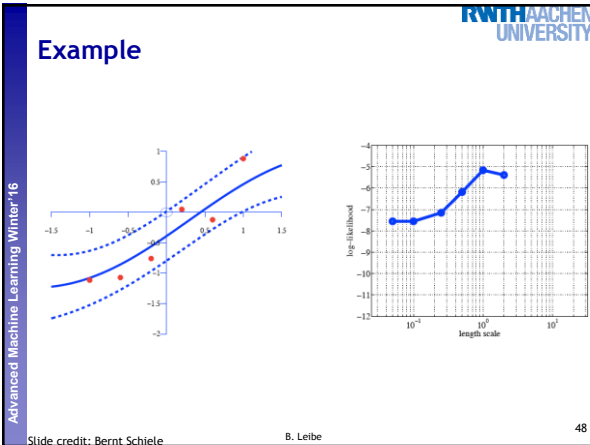
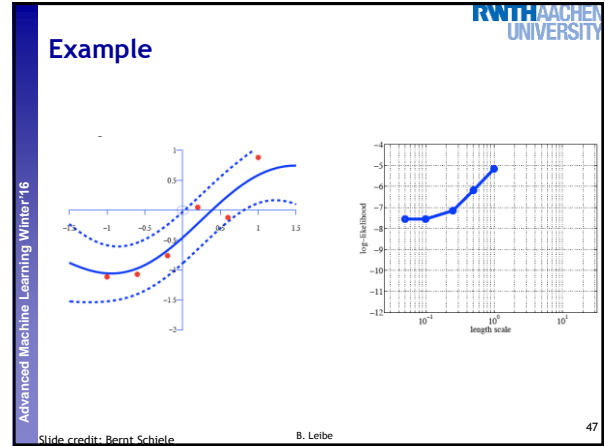
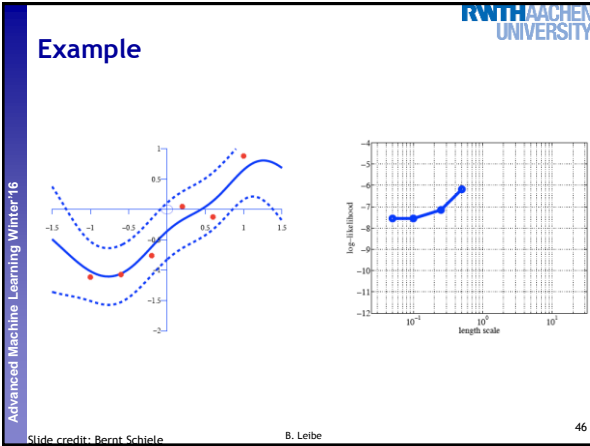
## Example

Slide credit: Bernt Schiele B. Leibe 44

RWTH AACHEN UNIVERSITY

## Example

Slide credit: Bernt Schiele B. Leibe 45



RWTH AACHEN UNIVERSITY

## Topics of This Lecture

- **Kernels**
  - Recap: Kernel trick
  - Constructing kernels
- **Gaussian Processes**
  - Recap: Definition
  - Prediction with noise-free observations
  - Prediction with noisy observations
  - GP Regression
  - Influence of hyperparameters
- **Learning Gaussian Processes**
  - Bayesian Model Selection
  - Model selection for Gaussian Processes
- **Applications**

65

RWTH AACHEN UNIVERSITY

## Application: Non-Linear Dimensionality Reduction

The slide illustrates non-linear dimensionality reduction. It shows a 2D manifold in 3D space (a curved surface) being mapped to a 2D space (a flat heatmap). Similarly, a 3D articulated body space (a 3D model of a person) is mapped to a 2D latent space (a 2D plot of body configurations).

66

RWTH AACHEN UNIVERSITY

## Gaussian Process Latent Variable Model

- At each time step  $t$ , we express our observations  $\mathbf{y}$  as a combination of basis functions  $\psi$  of latent variables  $\mathbf{x}$ .

$$\mathbf{y}_t = \sum_j b_j \psi_j(\mathbf{x}_t) + \delta_t$$

- This is modeled as a Gaussian process...

67

RWTH AACHEN UNIVERSITY

## Example: Style-based Inverse Kinematics

The slide shows learned GPLVMs for three different actions: a walk, a jump shot, and a baseball pitch. Each action is represented by a sequence of images showing the progression of the movement, with a corresponding latent space trajectory (a path of red dots) that captures the underlying style of the movement.

68

RWTH AACHEN UNIVERSITY

## Application: Modeling Body Dynamics

- **Task:** estimate full body pose in  $m$  video frames.
  - High-dimensional  $\mathbf{Y}$ .
  - Model body dynamics using **hierarchical Gaussian process latent variable model (hGPLVM)** [Lawrence & Moore, ICML 2007].

The diagram shows a hierarchical model structure: Time (frame #)  $\mathbf{T}$  (top), Latent space  $\mathbf{Z}$  (middle), and Configuration  $\mathbf{Y}$  (bottom). The model is trained on data from time  $t=1$  to  $t=T$ .

$$p(\mathbf{Z}|\mathbf{T}, \theta) = \prod_{i=1}^T \mathcal{N}(\mathbf{z}_{:,i} | \mathbf{0}, \mathbf{K}_{\mathbf{T}})$$

$$p(\mathbf{Y}|\mathbf{Z}, \theta) = \prod_{i=1}^T \mathcal{N}(\mathbf{y}_{:,i} | \mathbf{0}, \mathbf{K}_{\mathbf{z}})$$

69

RWTH AACHEN UNIVERSITY

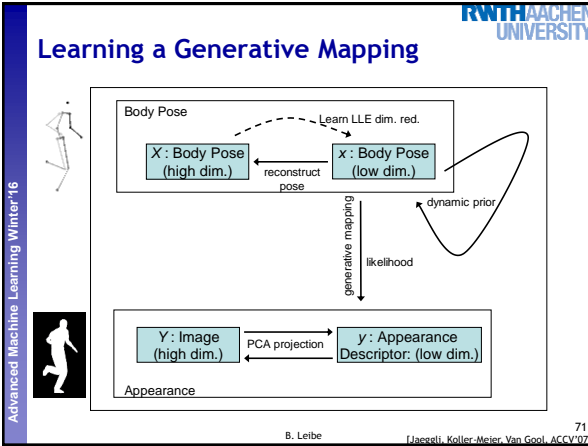
## Application: Mapping b/w Pose and Appearance

- **Appearance prediction**
  - Regression problem
  - High-dimensional data on both sides
  - ⇒ Low-dim, representation needed for learning!
- **Training with Motion-capture data possible**
  - Synthesized silhouettes for training
  - Background subtraction for test

The slide shows the mapping between pose and appearance. It includes 3D joint locations (a stick figure), 60-dim. data (a silhouette), and synthesized silhouettes (a sequence of images showing a person walking).

70





RWTH AACHEN UNIVERSITY

## Experimental Results

- Difficulties
  - Changing viewpoints
  - Low resolution (50 px)
  - Compression artifacts
  - Disturbing objects

Original video  
72  
[Jaegeli, Koller, Meier, Van Gool, ACCV'07]

RWTH AACHEN UNIVERSITY

## Articulated Motion in Latent Space (different work)

- Gaussian Process regression from latent space to
  - Pose [ $\rightarrow = p(\text{Pose} | z)$  to recover original pose from latent space]
  - Silhouette [ $\rightarrow = p(\text{Silhouette} | z)$  to do inference on silhouettes]

73  
B. Leibe [Gammeter, Ess, Leibe, Schindler, Van Gool, ECCV'08]

RWTH AACHEN UNIVERSITY

## Results

454 frames (~35 sec)  
23 Pedestrians  
20 detected by multi-body tracker  
74  
B. Leibe [Gammeter, Ess, Leibe, Schindler, Van Gool, ECCV'08]

RWTH AACHEN UNIVERSITY

## References and Further Reading

- Kernels and Gaussian Processes are (shortly) described in Chapters 6.1 and 6.4 of Bishop's book.

Christopher M. Bishop  
Pattern Recognition and Machine Learning  
Springer, 2006

Carl E. Rasmussen, Christopher K.I. Williams  
Gaussian Processes for Machine Learning  
MIT Press, 2006

- A better introduction can be found in Chapters 3 and 5 of the book by Rasmussen & Williams (also available online: <http://www.gaussianprocess.org/gpml/>)

75  
B. Leibe