University of Amsterdam
Informatics Institute

AMLAB
Amsterdam
Machine Learning Lab

Machine Learning 1

Lecture 12 - Kernel Methods
Gaussian Processes

Erik Bekkers

Image credit: Kirillm | Getty Images

University of Amsterdam
Informatics Institute

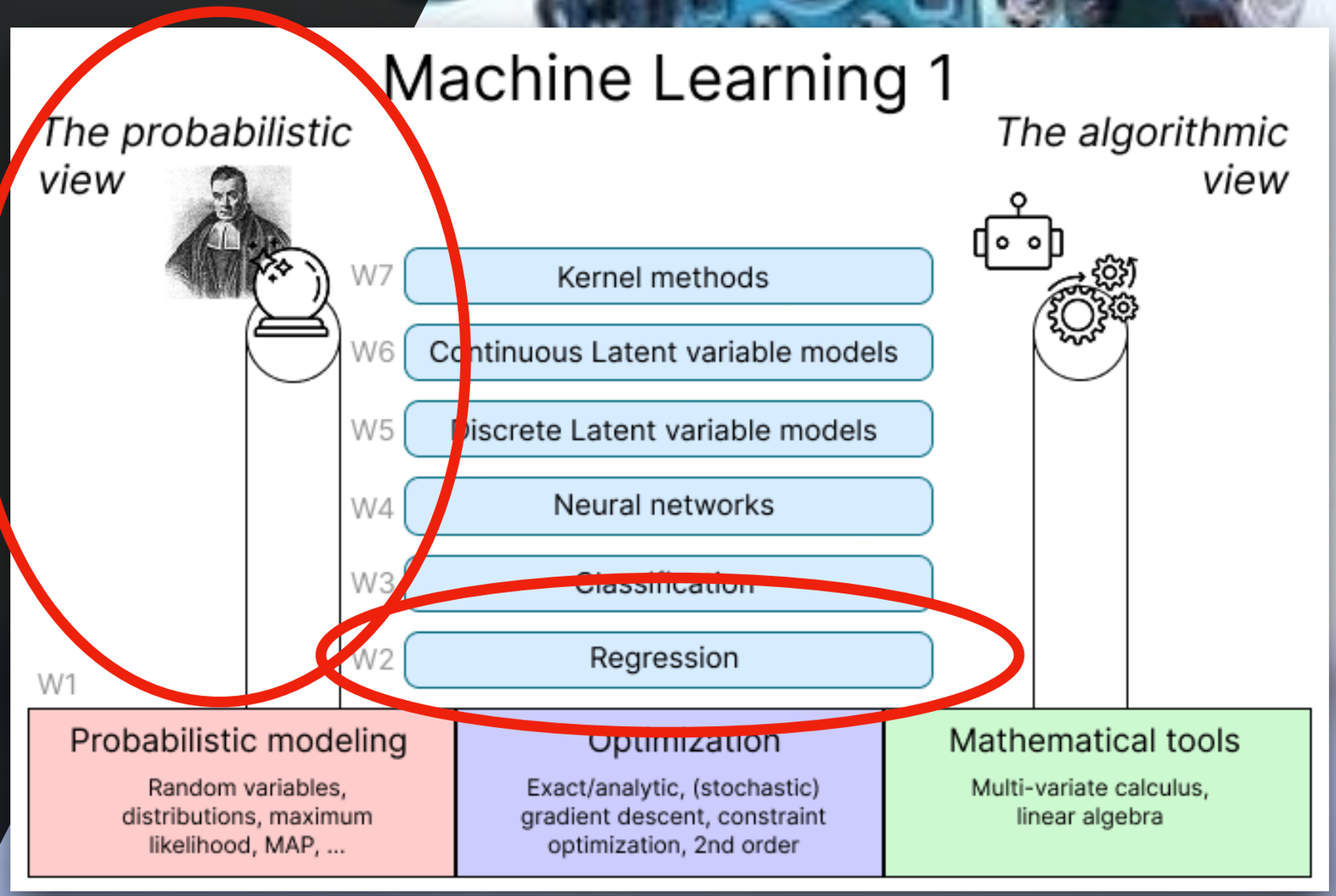
AMLAB
Amsterdam
Machine Learning Lab

Machine Learning 1

Lecture 5.1 - Supervised Learning
Bayesian Linear Regression - The Equivalent Kernel

Erik Bekkers

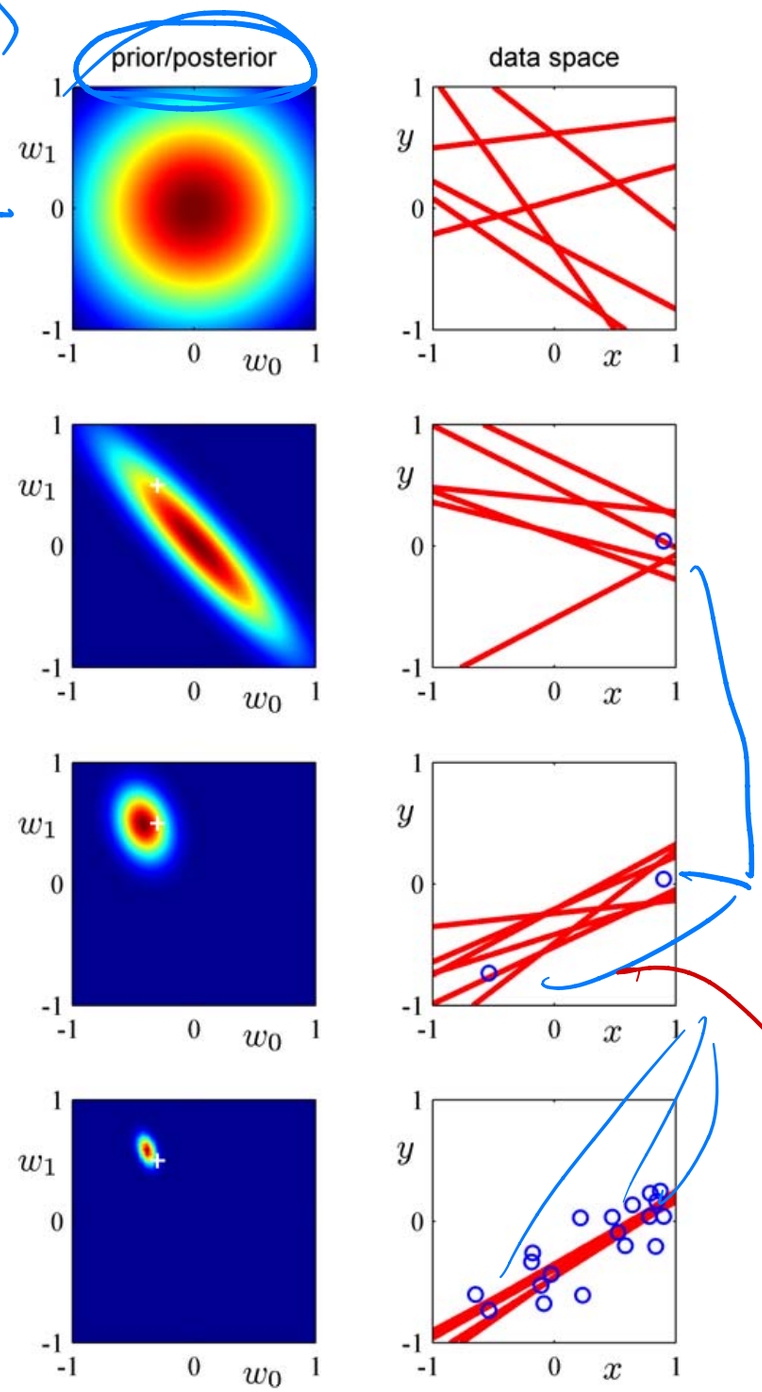
(Bishop 3.3.3)



Slide credits: Patrick Forré and Rianne van den Berg

Image credit: Kirillm | Getty Images

Bayesian Linear Regression



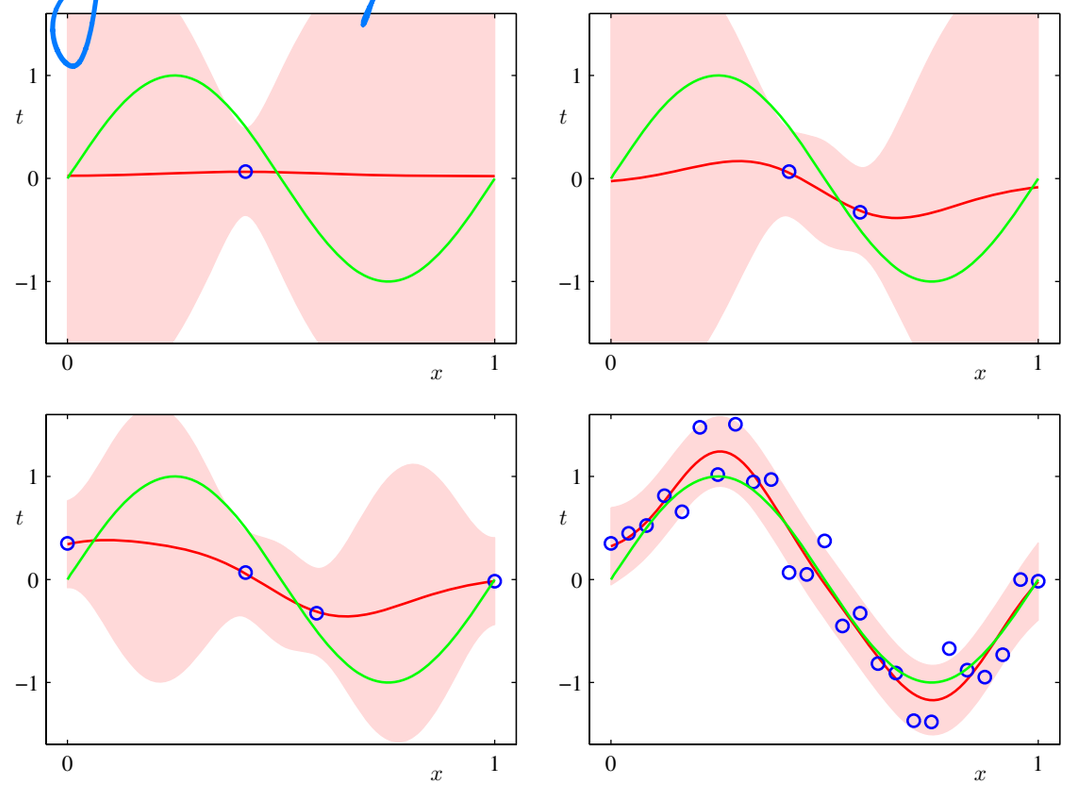
$p(\underline{w})$

$p(\underline{w} \mid D)$

Data D

Continuous Random Functions (Gaussian Processes)

Bayesian pred. dist.

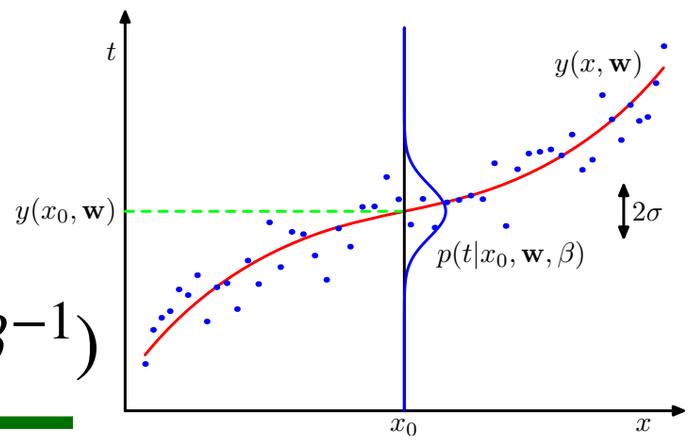


1. $\underline{w} \sim p(\underline{w} \mid D)$
2. plot $f_{\underline{w}}(x) = \underline{w}^T \underline{\phi}(x)$

$p(t \mid x, \underline{w}) = \mathcal{N}(t \mid f_{\underline{w}}(x), \beta^{-1})$

Bayesian Linear Regression

- Regression problem with:
  - Data: $\mathbf{X} = (\mathbf{x}_1, \ldots \mathbf{x}_N)^T$, $\mathbf{t} = (t_1, \ldots, t_N)^T$
  - **Predictive distribution** $p(t' \,|\, \mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t' \,|\, \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}'), \beta^{-1})$



- Probabilistic model with Gaussians:
  - Likelihood: $p(\mathbf{t} \,|\, \mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n \,|\, \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}) = \mathcal{N}(\mathbf{t} \mid \boldsymbol{\Phi}\mathbf{w}, \beta \mathbf{I})$
  - Conjugate prior: $p(\mathbf{w}) = \mathcal{N}(\mathbf{w} \,|\, \mathbf{m}_0, \mathbf{S}_0)$

Gaussian times Gaussian is a Gaussian

  - **Posterior**: $p(\mathbf{w} \,|\, \mathbf{t}, \mathbf{X}) = \dfrac{p(\mathbf{t} \,|\, \mathbf{X}, \mathbf{w}, \beta) p(\mathbf{w})}{p(\mathbf{t} \,|\, \mathbf{X}, \beta)} = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N)$
- Maximum A Posteriori estimate:
  - $\mathbf{w}_{\text{MAP}} = \underset{\mathbf{w}}{\text{argmax}}\, p(\mathbf{w} \mid \mathbf{t}, \mathbf{X}) = \mathbf{m}_N$

Bishop Ch 2.3, Eq. 2.116

$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$$
$$\mathbf{m}_N = \mathbf{S}_N (\mathbf{S}_0^{-1} \mathbf{m}_0 + \beta \boldsymbol{\Phi}^T \mathbf{t})$$

Predictive Distribution

- Observed dataset with inputs $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^T$ and targets $\mathbf{t} = (t_1, \ldots, t_N)^T$
- **Gaussian Posterior** distribution (from Gaussian prior and Gaussian likelihood)
  - $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N)$ with $\mathbf{m}_N = \beta\, \mathbf{S}_N\, \mathbf{\Phi}^T\, \mathbf{t}$

$$\mathbf{S}_N^{-1} = \alpha\, \mathbf{I} + \beta \mathbf{\Phi}^T \mathbf{\Phi}$$

- Parametrized Gaussian **predictive distribution**:
  - $p(t' \mid \mathbf{x}', \mathbf{w}, \beta) = \mathcal{N}(t \mid \boldsymbol{\phi}(\mathbf{x}')^T \mathbf{w}, \beta^{-1})$
- Analytic Gaussian **Bayesian predictive distribution** for new input
  - $p(t' \mid x', \mathbf{X}, \mathbf{t}, \alpha, \beta) = \int \mathcal{N}(t' \mid \boldsymbol{\phi}(\mathbf{x}')^T \mathbf{w}, \beta^{-1}) \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N) \mathrm{d}\mathbf{w}$

$$= \mathcal{N}(t' \mid \mathbf{x}', \boldsymbol{\phi}(\mathbf{x}')^T \mathbf{m}_N, \sigma_N^2(\mathbf{x}'))$$

- With $\sigma_N^2(\mathbf{x}') = \beta^{-1} + \boldsymbol{\phi}(\mathbf{x}')^T \mathbf{S}_N \boldsymbol{\phi}(\mathbf{x}')$

$p(t', \mathbf{w} \mid x', \mathbf{X}, \mathbf{t}, \alpha, \beta)$ → marginalization "weighted sum of pred.-distr."

Bishop Eq. 2.115

Integral / Convolution with two Gaussian is again a Gaussian

Predictive Distribution

- Datasets:
  - $t = \sin(2\pi x) + \epsilon$
  - $\epsilon \sim \mathcal{N}(0, \beta^{-1})$
- Dataset sizes:
  - $N = 1,2,4,25$
- Model:
  - $y(x, \mathbf{w}) = \boldsymbol{\phi}(x)^T \mathbf{w}$
  - $\phi_j(x)$: Gaussian basis functions

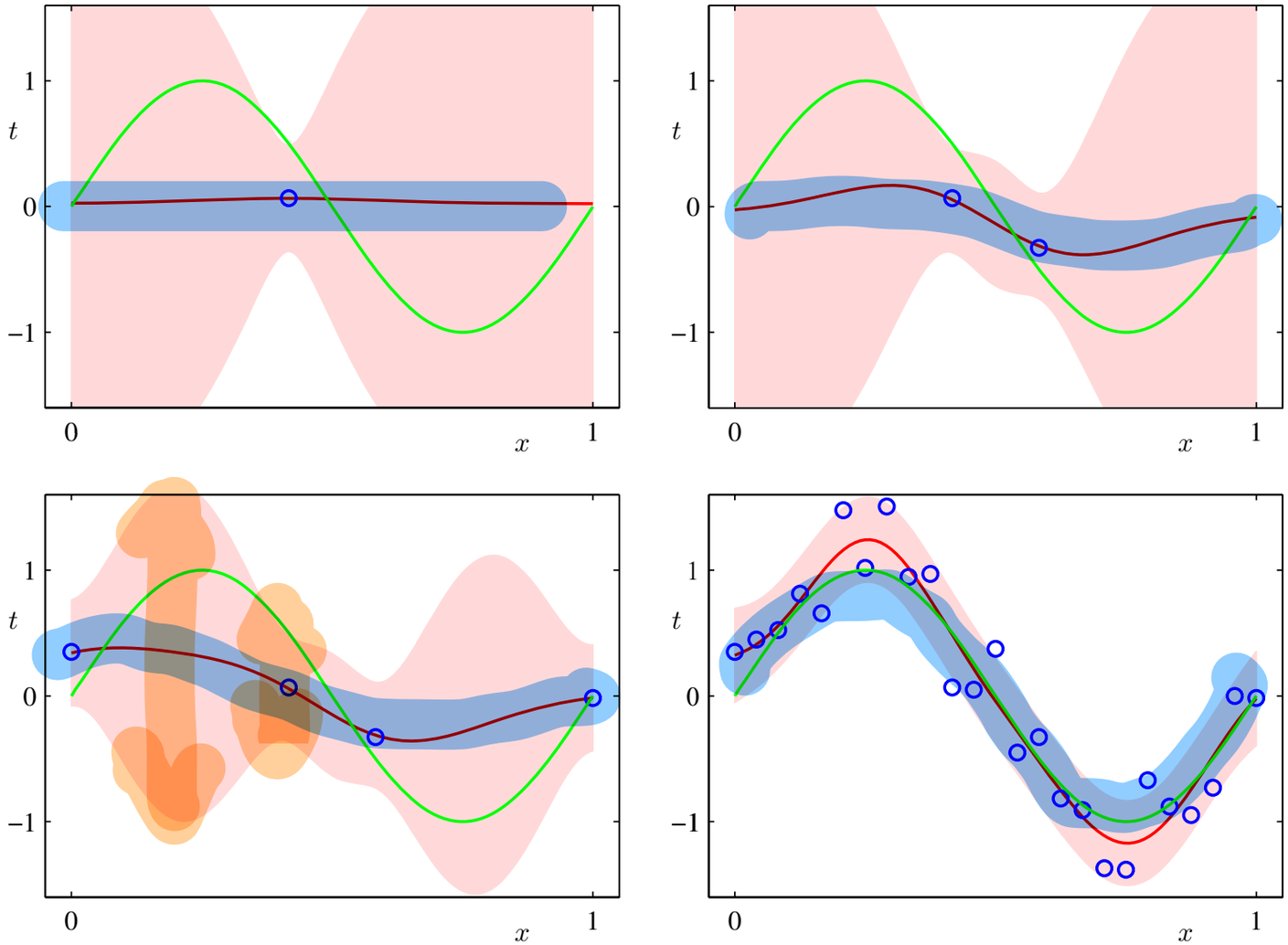
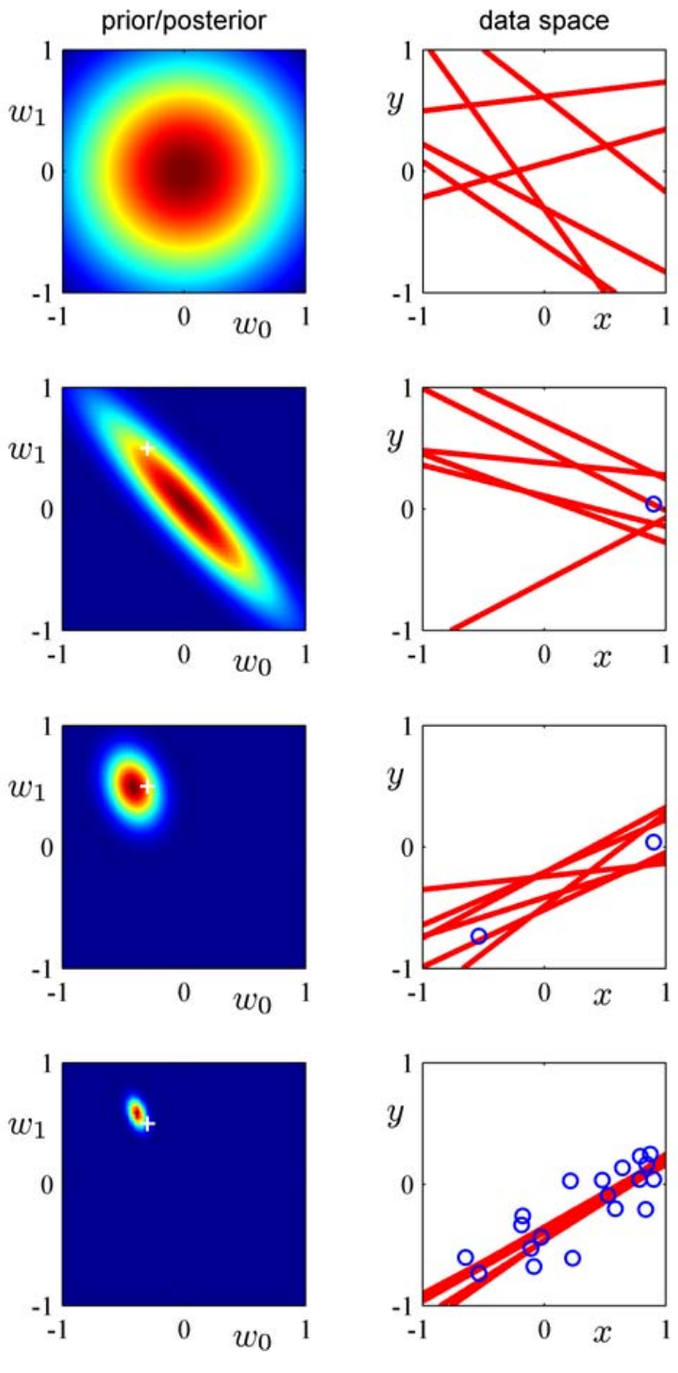


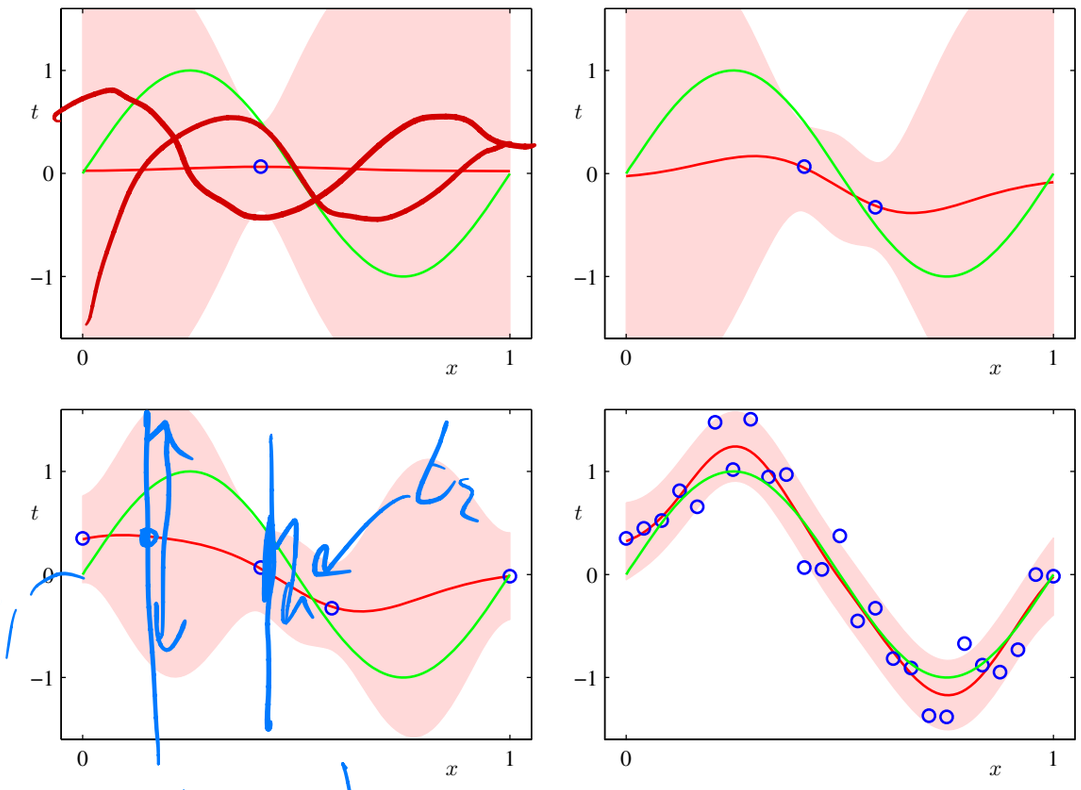
Figure: Predictive distribution (Bishop 3.8)

- **Bayesian predictive distribution**:
  - $p(t' \mid x', \mathbf{X}, \mathbf{t}, \alpha, \beta) = = \mathcal{N}(t' \mid \mathbf{x}', \boldsymbol{\phi}(\mathbf{x}')^T \mathbf{m}_N, \sigma_N^2(\mathbf{x}'))$
  - $\sigma_N^2(\mathbf{x}') = \beta^{-1} + \boldsymbol{\phi}(\mathbf{x}')^T \mathbf{S}_N \boldsymbol{\phi}(\mathbf{x}'), \quad \mathbf{m}_N = \beta\, \mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t}, \quad \mathbf{S}_N^{-1} = \alpha \mathbf{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$

Bayesian Linear Regression



Continuous Random Functions / Bayesian predictive distributions / Gaussian Processes



$$\mathrm{cov}[t_1', t_2' \mid x_1', x_2']$$

Kernel method!

(Predictions based on proximity to training data)

Equivalent Kernel Formulation

- **Bayesian predictive distribution**

$$p(t'|x', \mathbf{X}, \mathbf{t}, \alpha, \beta) = \int p(t'|x', \mathbf{w}, \beta) p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) \mathrm{d}\mathbf{w}$$
$$= \mathcal{N}(t'|\mathbf{m}_N^T \boldsymbol{\phi}(x'), \sigma_N^2(x'))$$

$$\mathbf{m}_N = \beta \mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t} \qquad \sigma_N^2(x') = \frac{1}{\beta} + \boldsymbol{\phi}(x')^T \mathbf{S}_N \boldsymbol{\phi}(x') \qquad \mathbf{S}_N^{-1} = \alpha \mathbb{1} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$$

- predictive mean:

parametric "primal" form

chosen parameters

$$y(x', \mathbf{m}_N) = \boldsymbol{\phi}(x')^T \mathbf{m}_N = \beta \boldsymbol{\phi}(x') \mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t} = \beta \boldsymbol{\phi}(x')^T \mathbf{S}_n \sum_{n=1}^{N} (\boldsymbol{\Phi}^T)_{:,n} t_n$$

$$= \sum_{n=1}^{N} \beta \boldsymbol{\phi}(x')^T \mathbf{S}_N \boldsymbol{\phi}(x_n) t_n$$

$$= \sum_{n=1}^{N} k(x', x_n) t_n$$

non-parametric form

"dual" form

Equivalent kernel for Gaussian Basis Functions

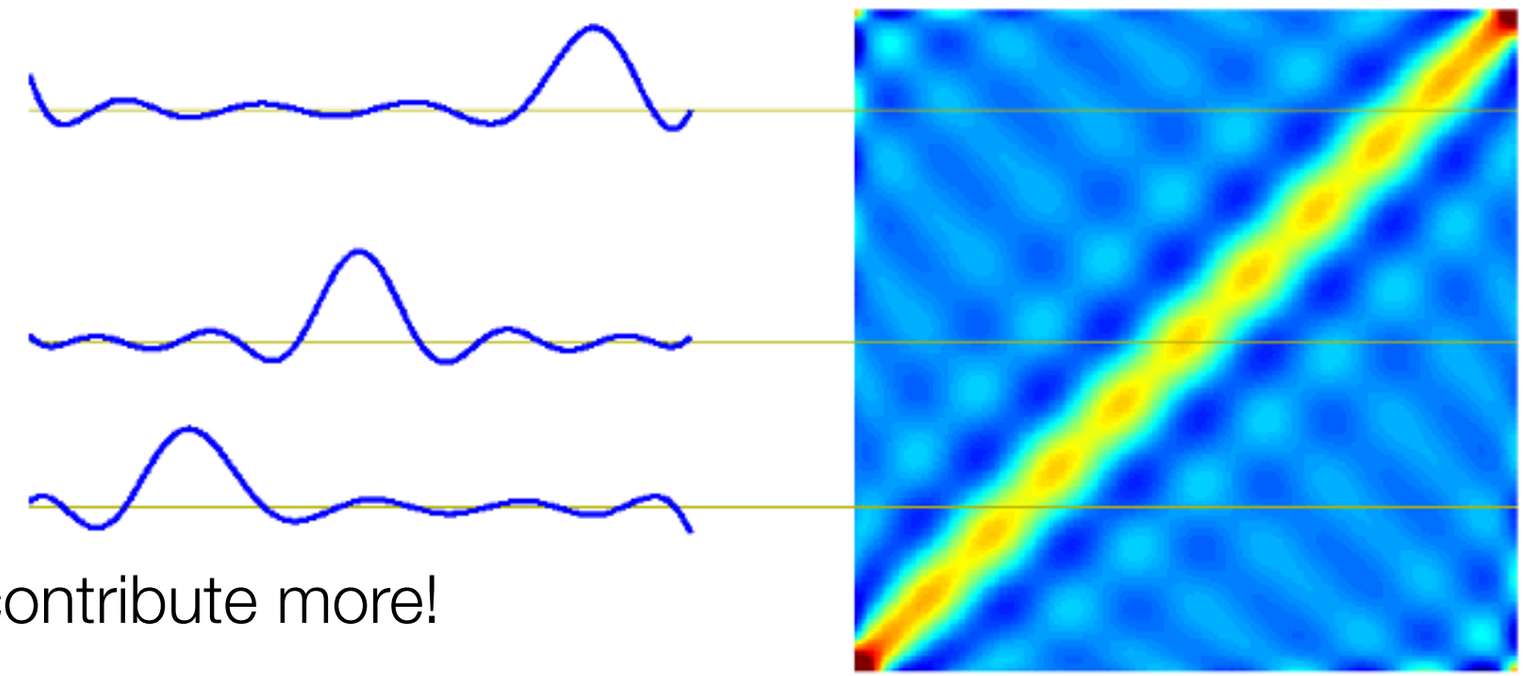
Figure: Equivalent kernel k(x', x) (Bishop 3.10)

- Localized kernel

$$k(x', x) = \beta \boldsymbol{\phi}(x')^T \mathbf{S}_N \boldsymbol{\phi}(x)$$

- predictive mean

$$y(x', \mathbf{m}_N) = \sum_{n=1}^{N} k(x', x_n) t_n$$



- Training points x_n close to x' contribute more!

- Covariance of between predictions:

$$\mathrm{cov}[t_1, t_2 | x_1, x_2] = \mathrm{cov}_{\mathbf{w}}[y(x_1, \mathbf{w}), y(x_2, \mathbf{w})] = \mathrm{cov}_{\mathbf{w}}[\boldsymbol{\phi}(x_1)^T \mathbf{w}, \mathbf{w}^T \boldsymbol{\phi}(x_2)]$$

$$= \mathbb{E}_{\mathbf{w}}[\boldsymbol{\phi}(x_1)^T \mathbf{w}\, \mathbf{w}^T \boldsymbol{\phi}(x_2)] - \mathbb{E}_{\mathbf{w}}[\boldsymbol{\phi}(x_1)^T \mathbf{w}] \mathbb{E}_{\mathbf{w}}[\mathbf{w}^T \boldsymbol{\phi}(x_2)]$$

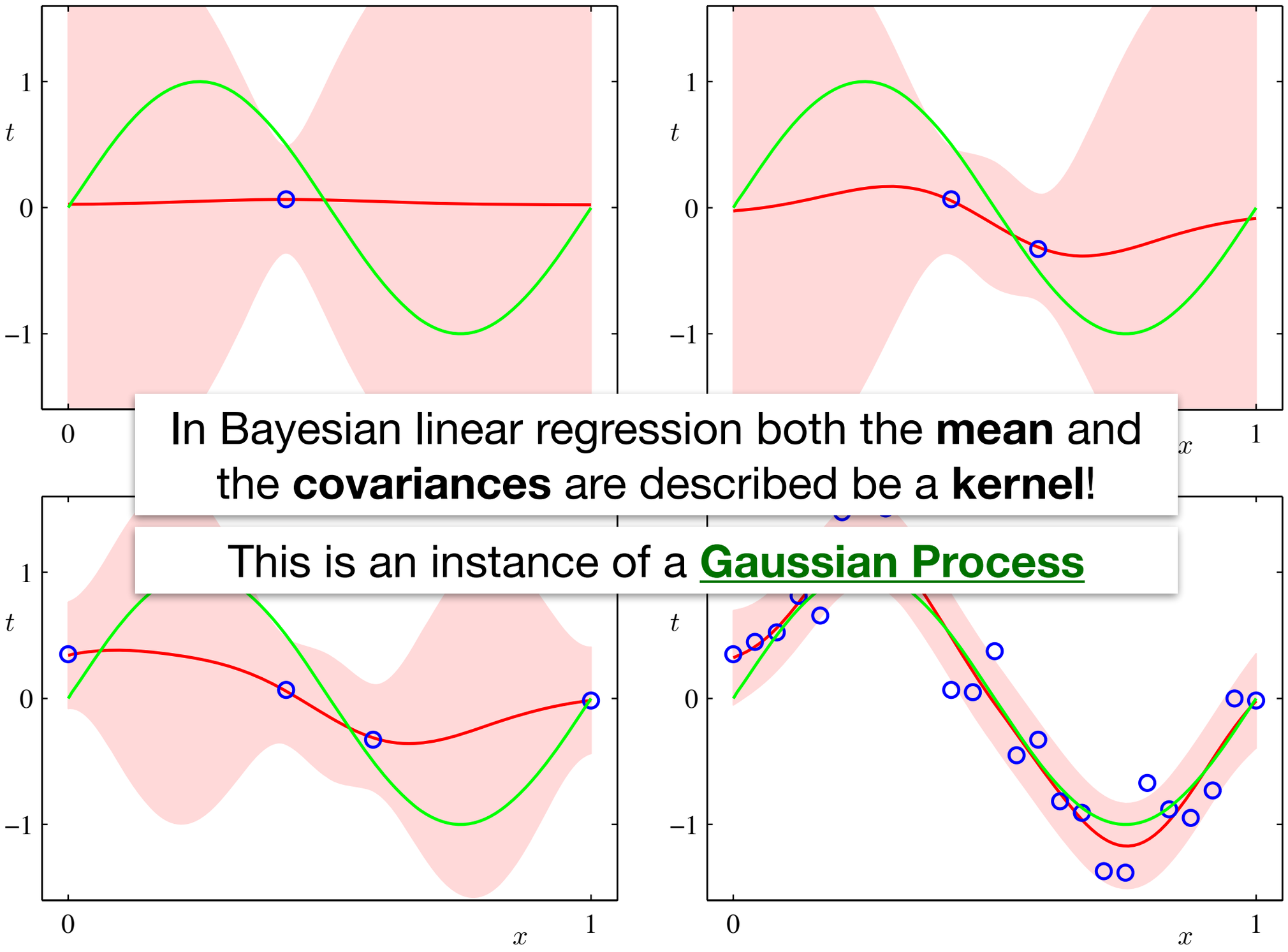
$$= \boldsymbol{\phi}(x_1)^T (\mathbb{E}_{\mathbf{w}}[\mathbf{w}\mathbf{w}^T] - \mathbb{E}[\mathbf{w}]\mathbb{E}[\mathbf{w}]^T) \boldsymbol{\phi}(x_2) = \boldsymbol{\phi}(x_1)^T \mathrm{cov}[\mathbf{w}, \mathbf{w}] \boldsymbol{\phi}(x_2) = \boldsymbol{\phi}(x_1)^T \mathbf{S}_N \boldsymbol{\phi}(x_2)$$

Kernel describes how prediction at two location are correlated!!!

$\beta^{-1} k(x_1, x_2)$

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N) \qquad \mathbb{E}[t'|x', \mathbf{w}] = y(x', \mathbf{w})$$

Revisit Bayesian linear regression



In Bayesian linear regression both the **mean** and the **covariances** are described be a **kernel**!

This is an instance of a **Gaussian Process**

University of Amsterdam
Informatics Institute

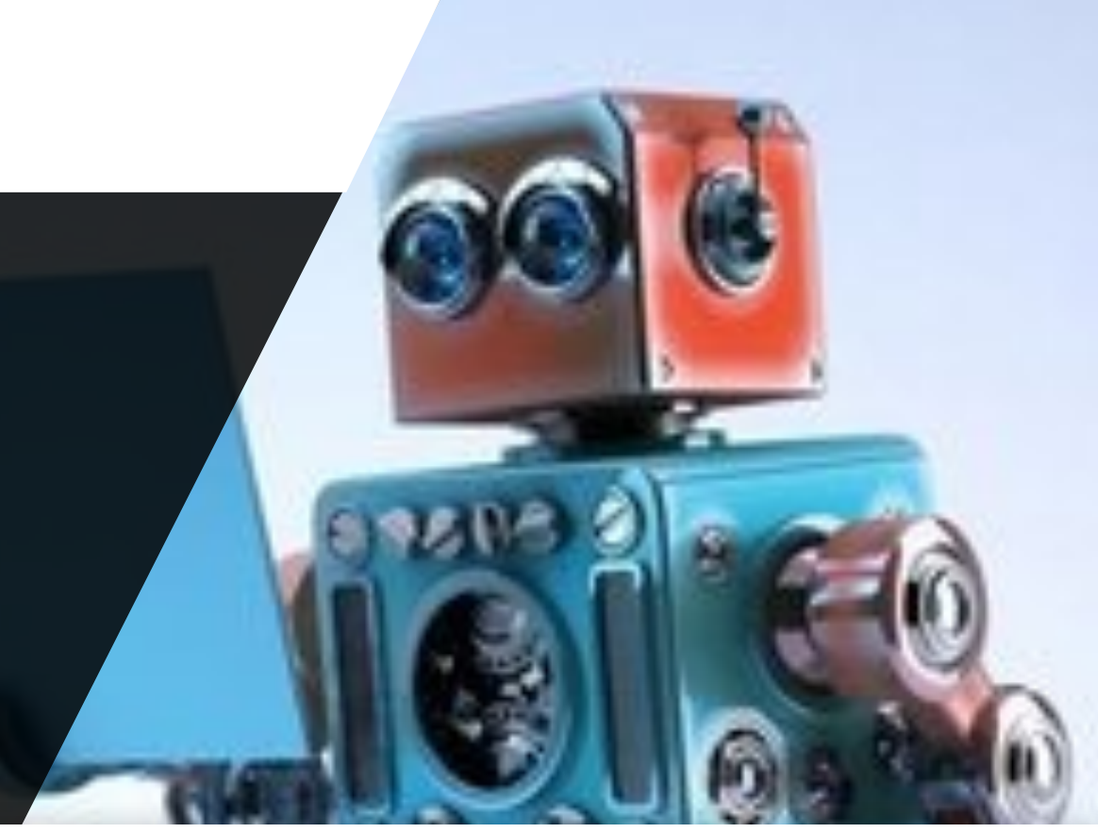
AMLAB
Amsterdam
Machine Learning Lab

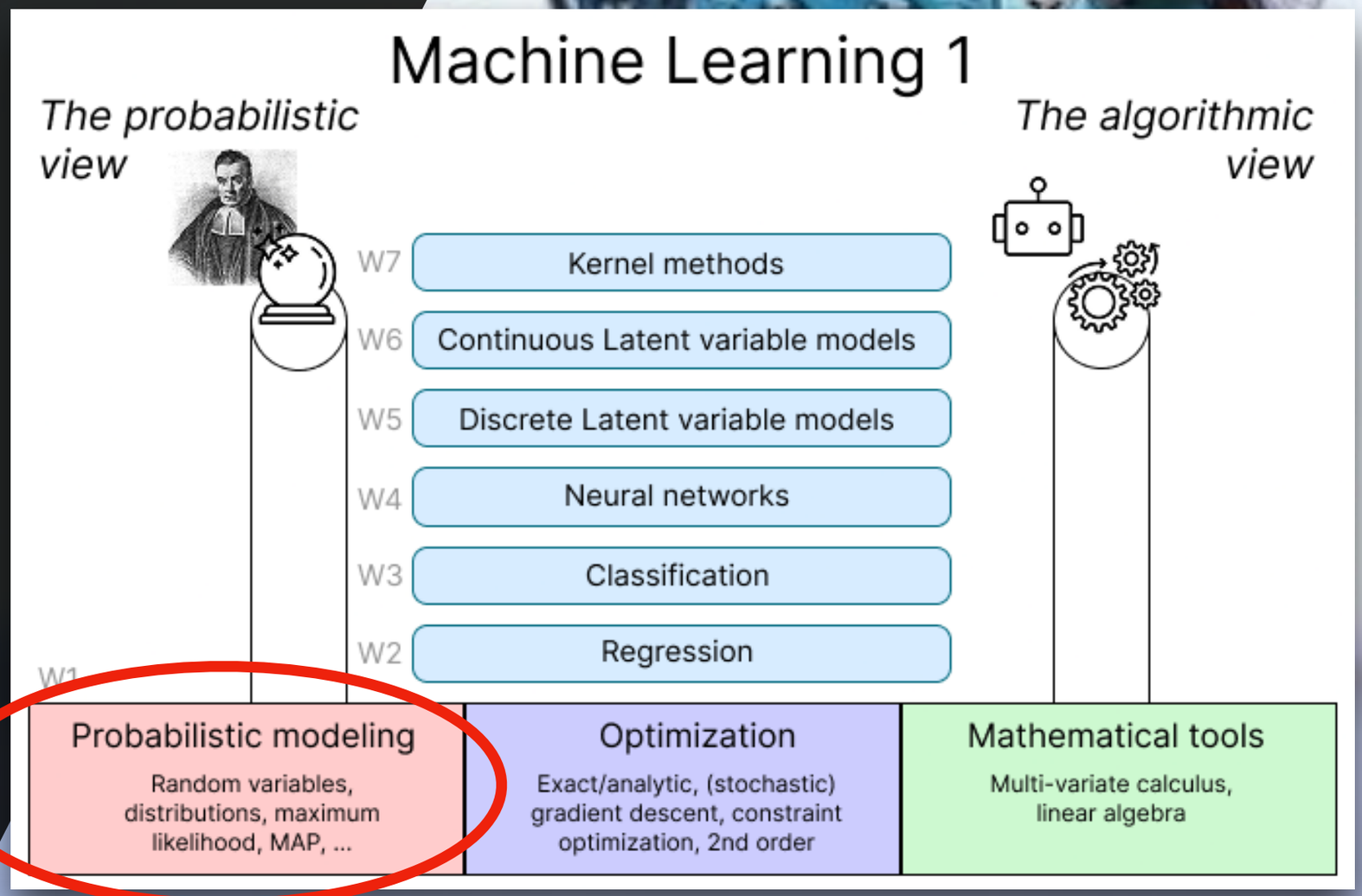
Machine Learning 1

Lecture 12.1 - Kernel Methods
Gaussian Processes - Properties of Gaussian Random Variables

Erik Bekkers

(Bishop 2.3.1, 2.3.2)





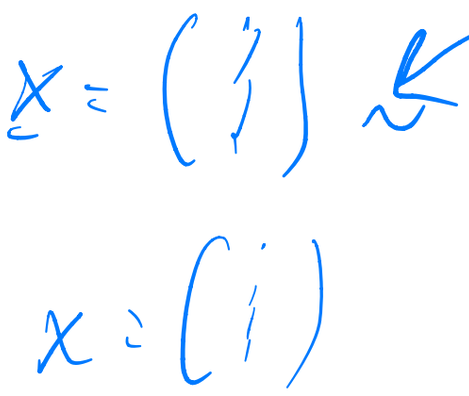
Slide credits: Patrick Forré and Rianne van den Berg

Image credit: Kirillm | Getty Images

Gaussian Processes are Gaussian **distributions for continuous functions**

Instead of sampling (finite dimensional) vectors

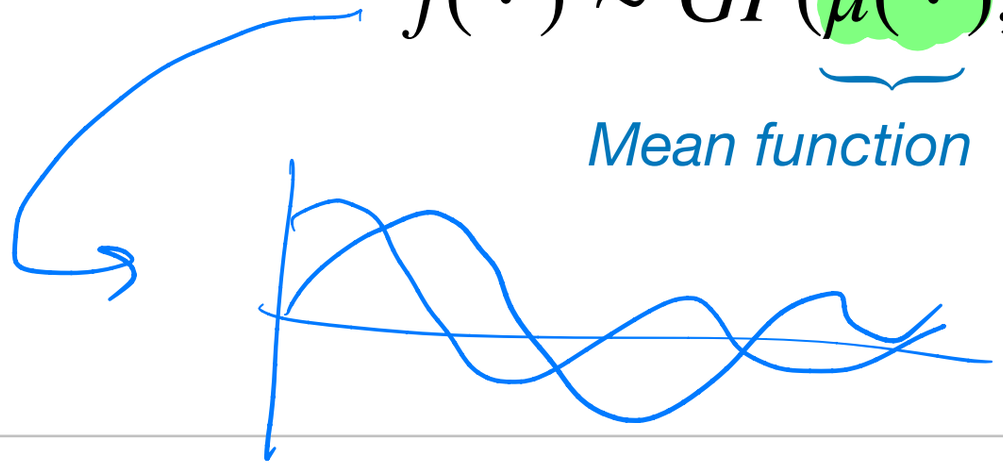
$$\mathbf{x} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$



With GPs we sample (infinite dimensional) functions

$$f(\cdot) \sim GP(\mu(\cdot), k(\cdot,\cdot))$$

Mean function *Kernel*



Gaussian Processes are
Gaussian **distributions for continuous functions**

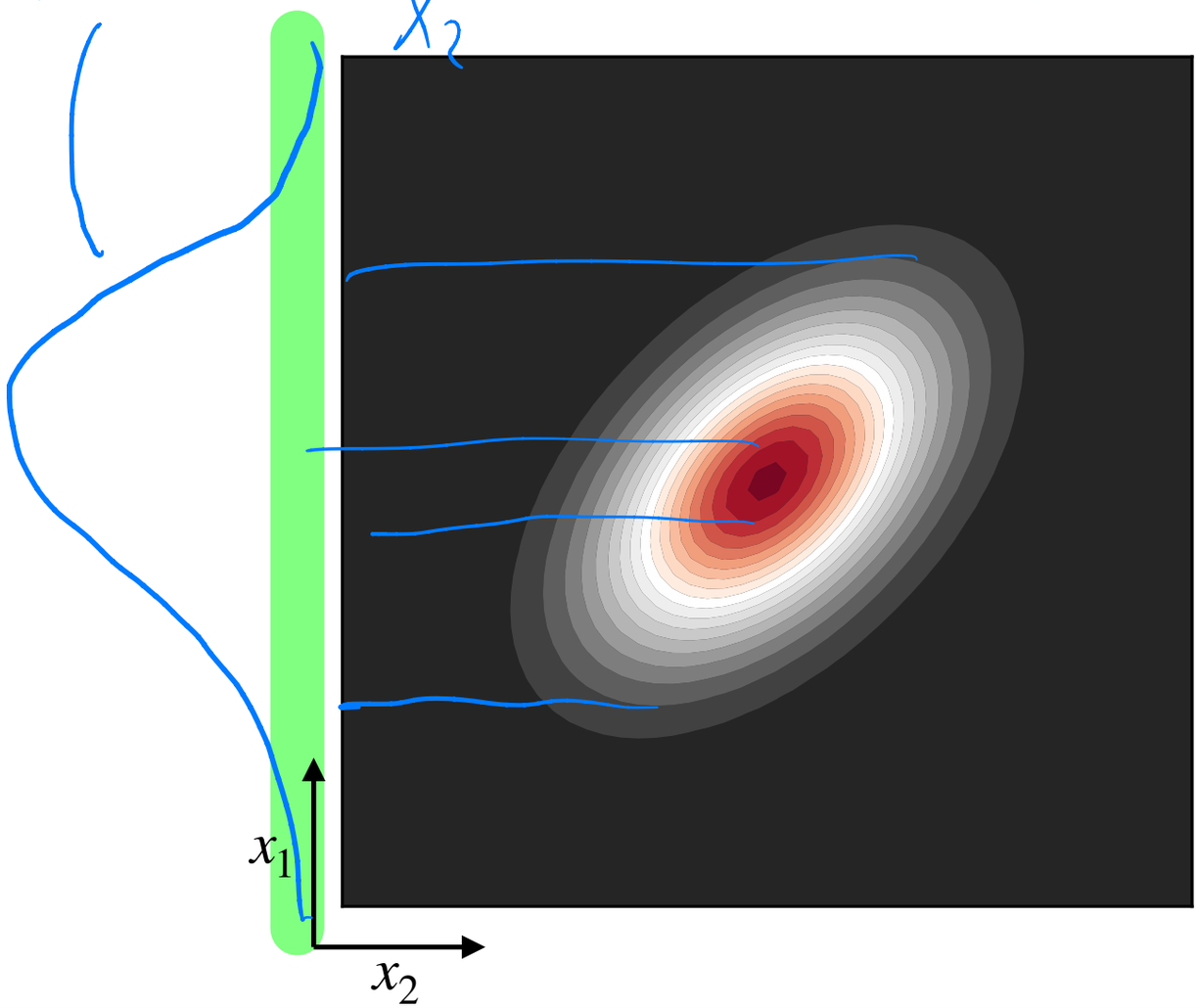
*In order to **sample from GPs** and in order to **fit** them we need to revisit some **basic properties of multi-variate Gaussians***

Gaussians: Marginalization property

- Take two random variables x_1 and x_2, that are jointly Gaussian distributed:

$$p(x_1, x_2) = \mathcal{N}\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \middle| \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right)$$

$$p(x_1) = \int_{x_2} p(x_1, x_2)\, dx_2$$



- Then the marginals are given by

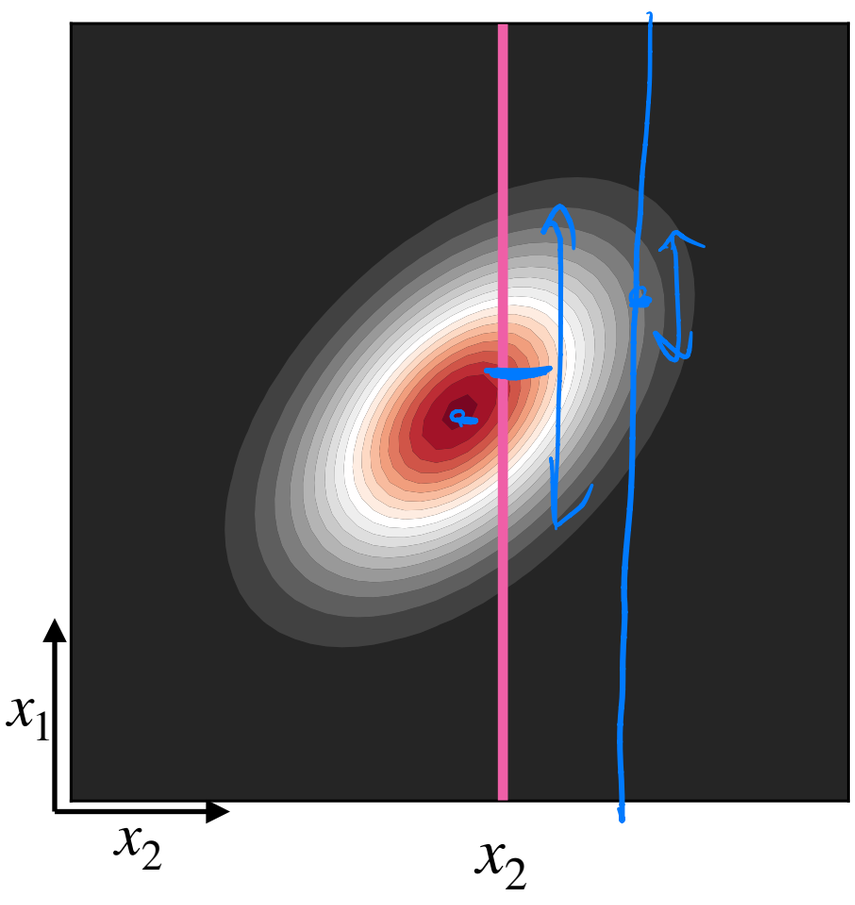
$$p(x_1) = \mathcal{N}(x_1 \mid \mu_1, \Sigma_{11})$$

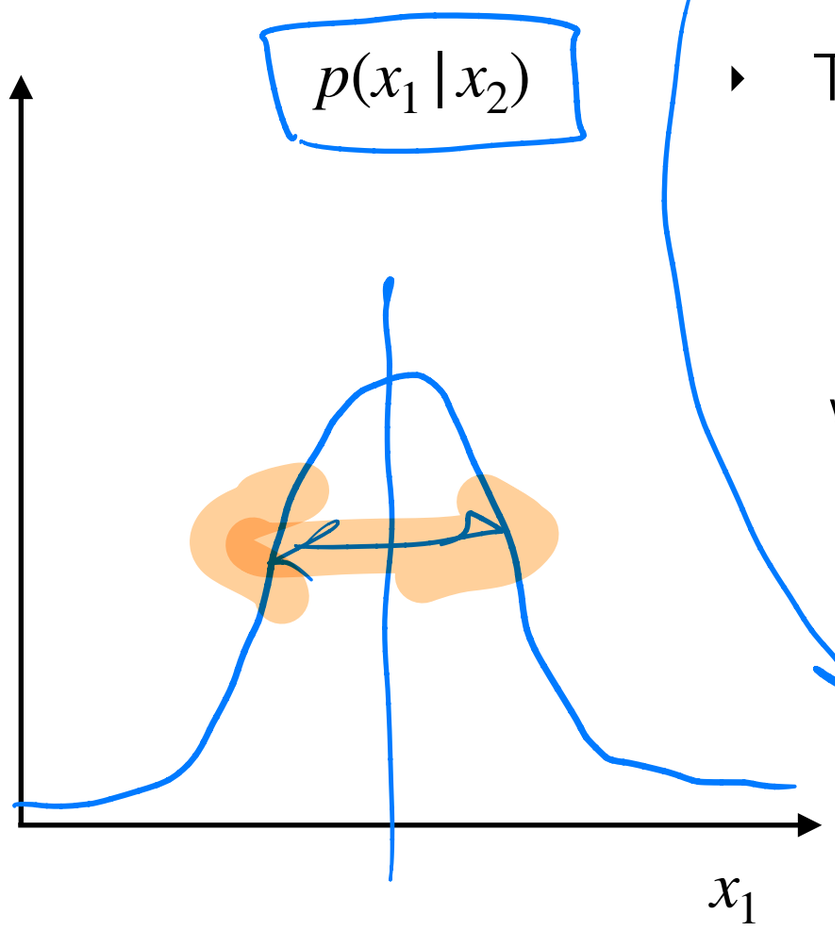
$$p(x_2) = \mathcal{N}(x_2 \mid \mu_2, \Sigma_{22})$$

Gaussians: Conditioning Property

- Take two random variables x_1 and x_2, that are jointly Gaussian distributed:

$$p(x_1, x_2) = \mathcal{N}\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \middle| \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right)$$





- Then the conditional is:

$$p(x_1 \mid x_2) = \mathcal{N}(\mu_{1|2}, \Sigma_{1|2})$$

with

$$\mu_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2)$$

$$\Sigma_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$$

Summing Random Variables

- The sum of two independent Gaussian random variables is also a Gaussian random variable:
  - If

$$x \sim \mathcal{N}(\mu, \Sigma)$$

$$y \sim \mathcal{N}(\mu', \Sigma')$$

  - Then

$$z = x + y \qquad \rightarrow \qquad z \sim \mathcal{N}(\mu + \mu', \Sigma + \Sigma')$$

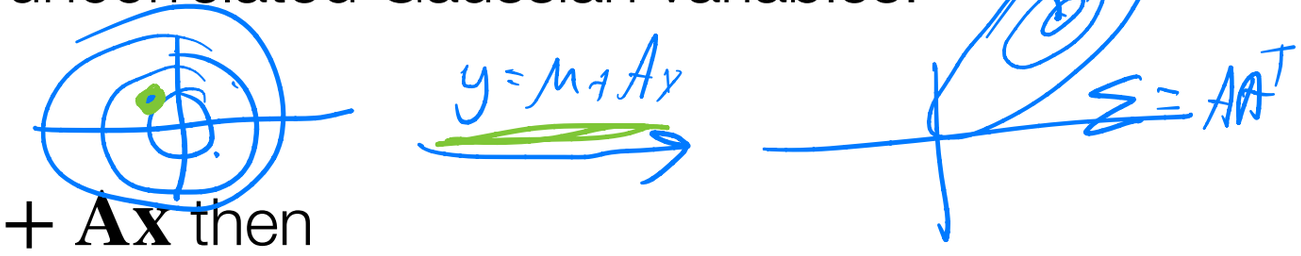
$\mu + \mu' = \mathbb{E}[z]$

$\mathbb{E}[z] = \mathbb{E}[x + y] = \mathbb{E}[x] + \mathbb{E}[y] = \mu + \mu'$

Sampling Correlated Gaussian Variables

- If we have sampled a vector $\mathbf{x}$ of uncorrelated Gaussian variables:

$$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$



If $\mathbf{y}$ is obtained from $\mathbf{x}$ via $\mathbf{y} = \boldsymbol{\mu} + \mathbf{A}\mathbf{x}$ then

$$\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{A}\mathbf{A}^T) \qquad \text{with} \qquad \boldsymbol{\Sigma} = \mathbf{A}\mathbf{A}^T$$

- **Reparametrization trick**: You want to sample correlated samples $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ from a Gaussian with some covariance $\boldsymbol{\Sigma}$:
  1. Determine $\mathbf{A}$ by e.g. eigendecomposition s.t. $\boldsymbol{\Sigma} = \mathbf{A}\mathbf{A}^T$
  2. Simply sample from a Gaussian $\mathbf{x} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
  3. Obtain $\mathbf{y}$ via $\mathbf{A}\mathbf{x} + \boldsymbol{\mu}$

- *Tips for computing $\mathbf{A}$*
  - *For a given $\boldsymbol{\Sigma}$, you can compute $\boldsymbol{\Sigma} = \mathbf{A}\mathbf{A}^T$ with a Cholesky decomposition such that $\mathbf{A}$ is lower triangular*
  - *Or you compute the eigendecomposition $\boldsymbol{\Sigma} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$ and take $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}^{1/2}$*

multi-variate case

Summary

- Consider $\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}$ and $p(\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \right)$

$\mathbf{x} \in \mathbb{R}^5$, $\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \in \mathbb{R}^3$, $\boldsymbol{\mu}_1 \in \mathbb{R}^2$, $\boldsymbol{\Sigma}_{11} \in \mathbb{R}^{2\times 2}$, $\boldsymbol{\Sigma}_{12} \in \mathbb{R}^{2\times 3}$

- Then the marginals are given by

$$p(\mathbf{x}_1) = \mathcal{N}\left(\mathbf{x}_1 | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}\right) \quad \text{and} \quad p(\mathbf{x}_2) = \mathcal{N}\left(\mathbf{x}_2 | \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22}\right)$$

- And the conditional is given by

$$p(\mathbf{x}_1 | \mathbf{x}_2) = \mathcal{N}\left(\mathbf{x}_1 | \boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2}\right) \quad \text{with} \quad \boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{x}_2 - \boldsymbol{\mu}_2)$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

- If $\mathbf{x}$ is an uncorrelated Gaussian random variable (i.e., $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$) then $\mathbf{y} = \boldsymbol{\mu} + \mathbf{A}\mathbf{x}$ is correlated $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{A}\mathbf{A}^T)$ with $\boldsymbol{\Sigma} = \mathbf{A}\mathbf{A}^T$

University of Amsterdam
Informatics Institute

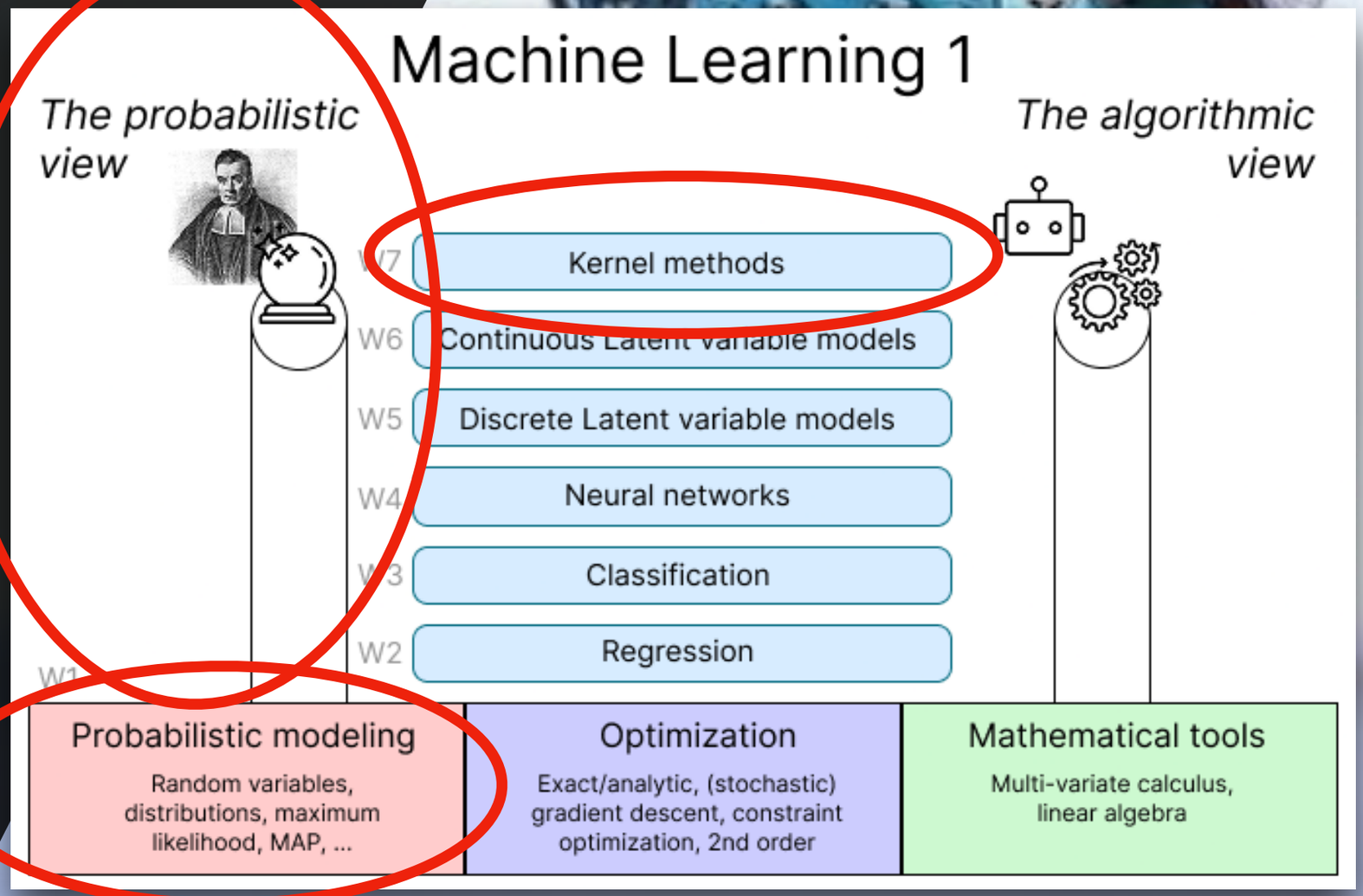
AMLAB
Amsterdam
Machine Learning Lab

Machine Learning 1

Lecture 12.3 - Kernel Methods
Gaussian Processes - Definition

Erik Bekkers

(Bishop 6.4.1)



Slide credits: Patrick Forré and Rianne van den Berg

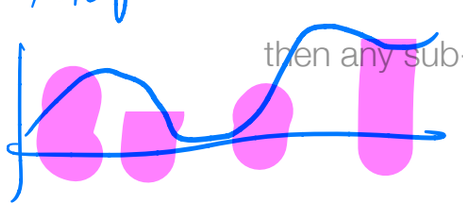
Image credit: Kirillm | Getty Images

Gaussian Processes

Recall property of Gaussians: If $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} \sim N(\mu_{1-5}, \Sigma_{1-5})$,

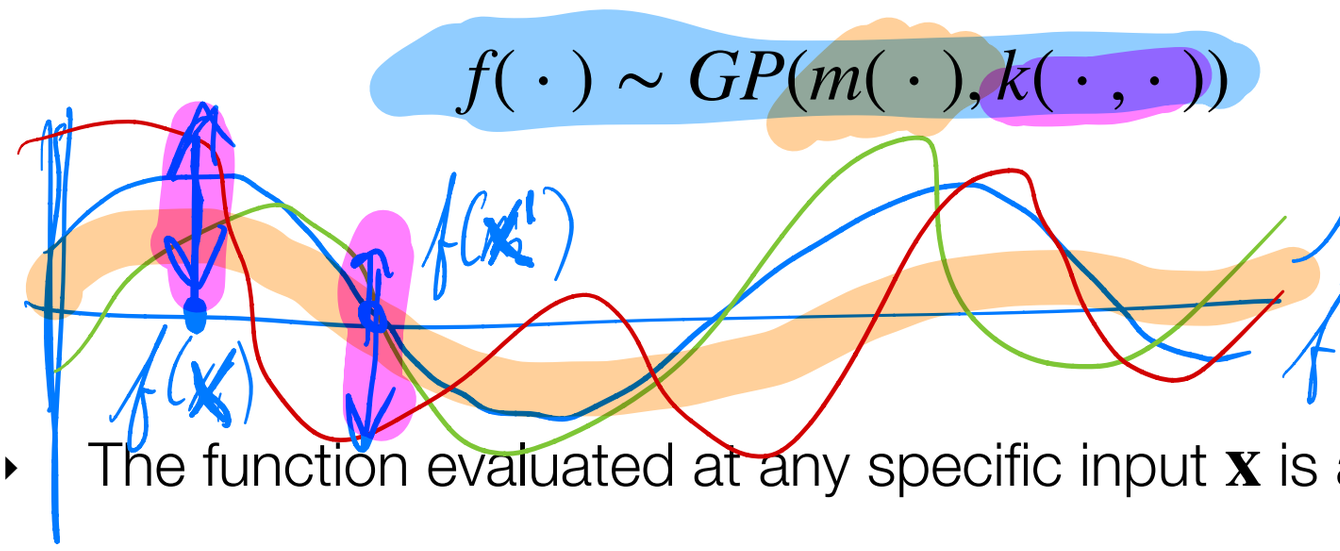
then any sub-vector is also Gaussian. E.g. $\begin{pmatrix} x_2 \\ x_4 \\ x_6 \end{pmatrix} \sim N(\mu_{2,4,5}, \Sigma_{2,4,5})$

- **Definition (Gaussian Process)**:



A Gaussian process is a collection of random variables indexed with time or space, any finite number of which is jointly Gaussian distributed

A Gaussian process is a distribution for random functions



- The function evaluated at any specific input $\mathbf{x}$ is a random variable $f(\mathbf{x})$, with

$$\mathbb{E}[f(\mathbf{x})] = m(\mathbf{x})$$

$$\text{cov}(f(\mathbf{x}), f(\mathbf{x}')) = \mathbb{E}[\left(f(\mathbf{x}) - m(\mathbf{x})\right)\left(f(\mathbf{x}') - m(\mathbf{x}')\right)] = k(\mathbf{x}, \mathbf{x}')$$

Weather forecasting

temperature f as a function ~~of location $x \in S^2$~~ of location & time $x \in S^2 \times \mathbb{R}$

$f : S^2 \times \mathbb{R} \to \mathbb{R}$

one measurement

$\to f(x) \sim N(m(x), k(x,x))$ normal distr

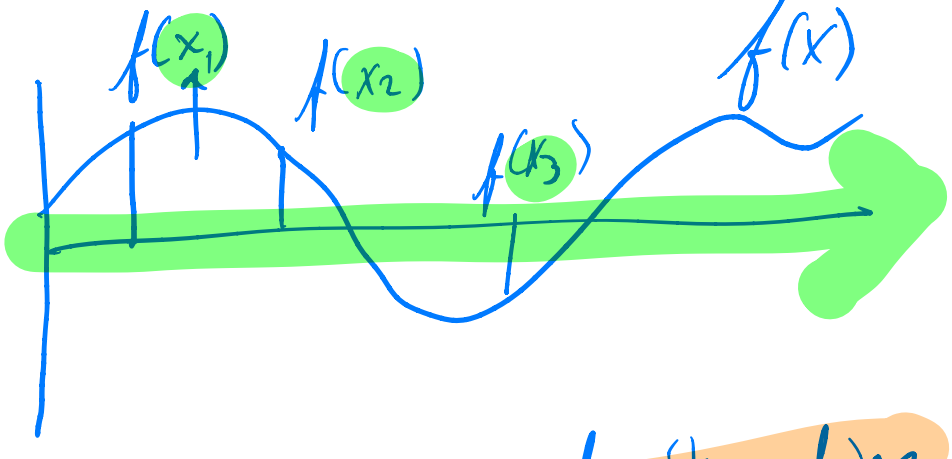
temperature measured at fixed location x. One measurement is random. But actually the entire func is random

$f(\cdot) \sim GP(m(\cdot), k(\cdot,\cdot))$

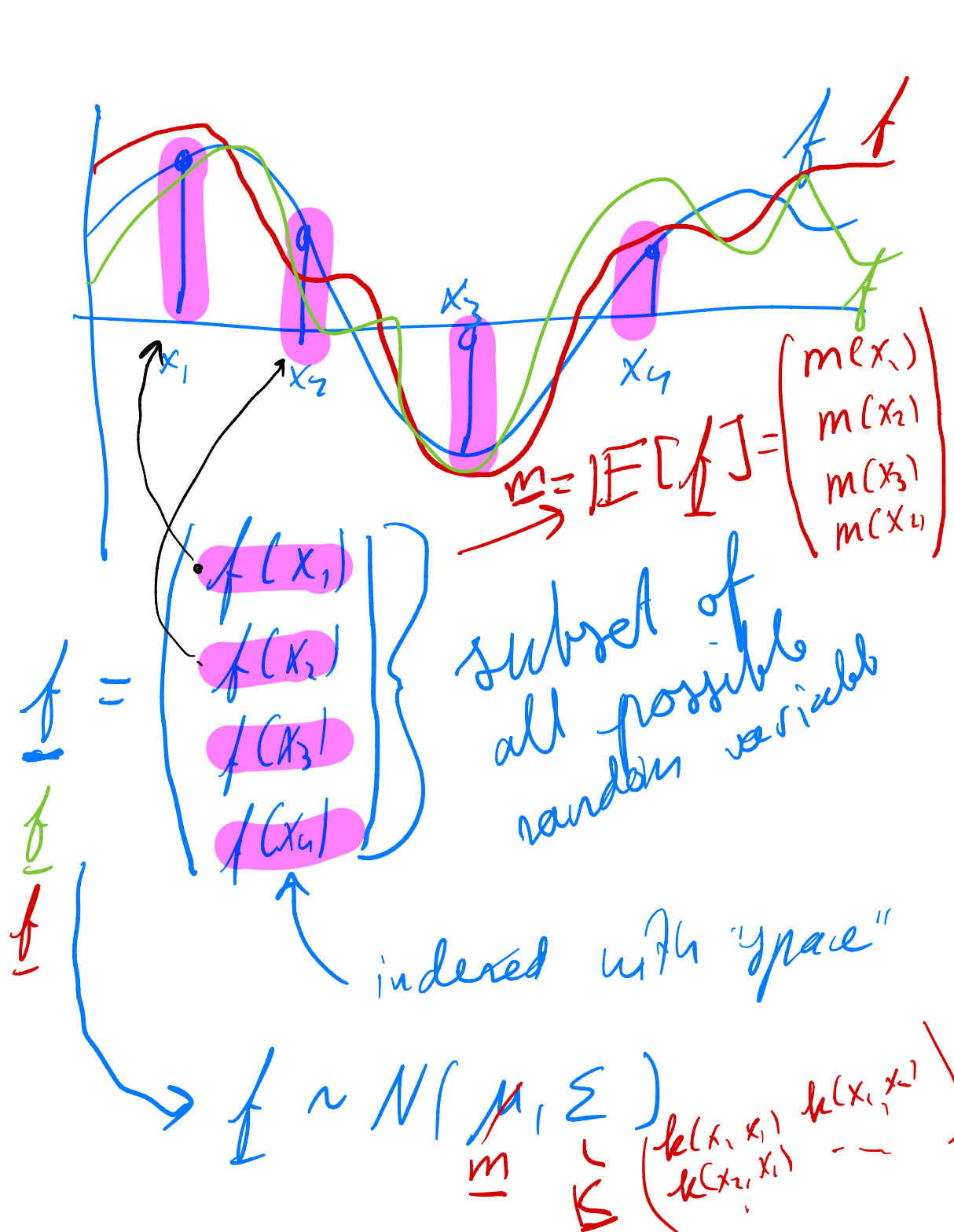
$$\underline{f} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{pmatrix} \in \mathbb{R}^N$$

finite dim vector space

vector elements are indexed with integer index



functions are infinite dim. vectors. It's value are continuously indexed with indices over the real line $\mathbb{R}$


x_1
x_2
x_3
x_4
$\underline{m} = \mathbb{E}[\underline{f}] = \begin{pmatrix} m(x_1) \\ m(x_2) \\ m(x_3) \\ m(x_4) \end{pmatrix}$
$\underline{f} = \begin{pmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \\ f(x_4) \end{pmatrix}$
subset of all possible random variables
indexed with "space"
$\underline{f}$
$\underline{f} \sim N(\underline{m}, \underline{K})$
$\underline{K} = \begin{pmatrix} k(x_1,x_1) & k(x_1,x_2) & \cdots \\ k(x_2,x_1) & \cdots & \end{pmatrix}$

Gaussian Processes

directly defines distr. on functions

- Think of functions as infinite dimensional vectors

$\rightarrow f(\cdot) \sim GP(m(\cdot), k(\cdot,\cdot))$

Sampling function $f(\cdot)$

In contrast:
Sampling random weights $\underline{w} \sim p(\underline{w}) \rightarrow f(x) = \phi(x)^T \underline{w}$
gives indirectly a distribution of functions. This is what we did before in the parametric scenario.

- Sample **finite-dimensional** vectors $\mathbf{f} \in \mathbb{R}^N$ from **Gaussians** $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean vector $\boldsymbol{\mu} \in \mathbb{R}^N$, and covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{N \times N}$

Recall property of Gaussians: If $\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} \sim N(\mu, \Sigma)$, then finite subset is also Gaussian. E.g. $\begin{pmatrix} f_2 \\ f_4 \\ f_5 \end{pmatrix} \sim N(\mu_{2,4,5}, \Sigma_{2,4,5})$

- Sample infinite-dimensional vectors (functions) $f : \mathbb{R} \rightarrow \mathbb{R}$ from **Gaussian processes** $GP(m(\cdot), k(\cdot,\cdot))$, defined by mean function $m(\cdot)$ and covariance kernel $k(\cdot,\cdot)$

Gaussian process: If $f(\cdot) \sim GP(m(\cdot), k(\cdot,\cdot))$, then finite subset is also Gaussian. E.g. $\begin{pmatrix} f(x_1) \\ f(x_4) \\ f(x_5) \end{pmatrix} \sim N(\mu_{2,4,5}, \Sigma_{2,4,5})$

Functional Viewpoint, why is this a GP?

- Take any finite set $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ with corresponding random variables $\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)\}$ then

$$p\left(\begin{bmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_N) \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} m(\mathbf{x}_1) \\ \vdots \\ m(\mathbf{x}_N) \end{bmatrix}, \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \ldots & k(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \mathbf{x}_1) & \ldots & k(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}\right)$$

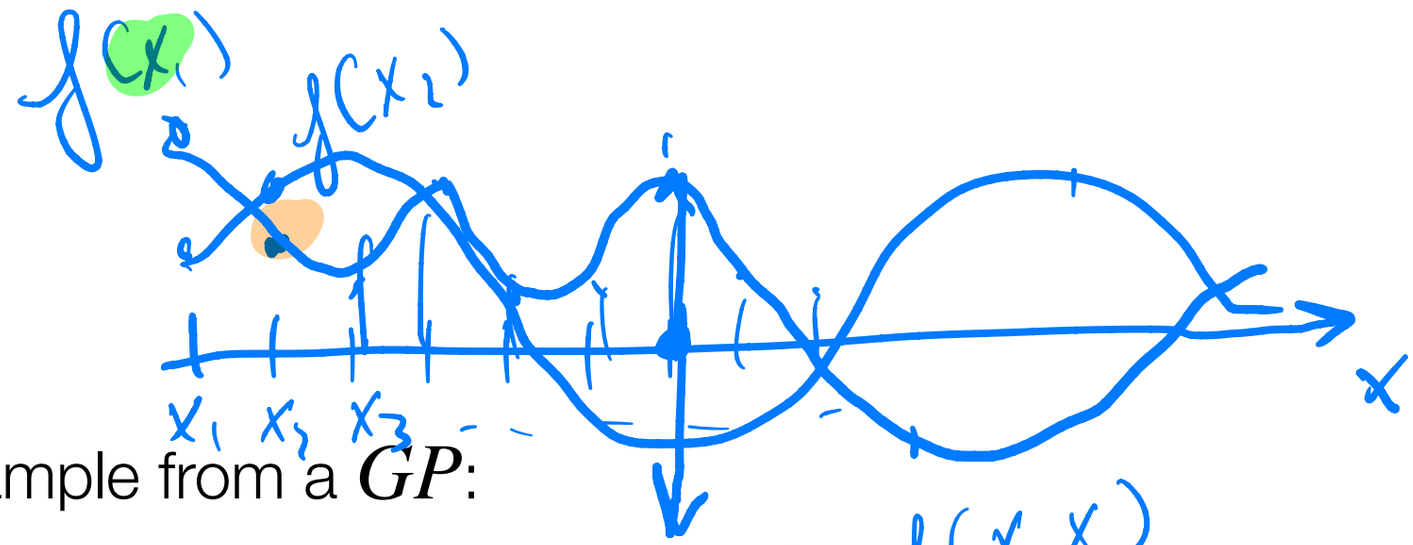
- **Consistency requirement**: any subset of $\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)\}$ should also be Gaussian distributed.

- This is true because of Gaussian marginalization property:

$$p\left(\begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{m}_1 \\ \boldsymbol{m}_2 \end{bmatrix}, \begin{bmatrix} \boldsymbol{K}_{11} & \boldsymbol{K}_{12} \\ \boldsymbol{K}_{21} & \boldsymbol{K}_{22} \end{bmatrix}\right) \rightarrow p(\mathbf{f}_1) = \mathcal{N}\left(\boldsymbol{m}_1, \boldsymbol{K}_{11}\right)$$

Functions as vectors

- Think of a function $f(\cdot)$ drawn from a GP as an extremely high-dimensional vector drawn from an extremely high-dimensional multivariate Gaussian distribution



- How do you sample from a GP:
  - Generate grid of points $\mathbf{x} \in \mathbb{R}^n$
  - Construct the Gram matrix $\mathbf{K}$ for all sampled $\mathbf{x}$.
  - Sample vector.

"matrix of inner products"

$$\begin{bmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_N) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(\mathbf{x}_1) \\ \vdots \\ m(\mathbf{x}_N) \end{bmatrix}, \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \dots & k(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \mathbf{x}_1) & \dots & k(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix} \right)$$

$m \in \mathbb{R}^N$

Example: Bayesian Linear Regression

- Bayesian linear models:

$$f(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x})^T \mathbf{w}$$

- Prior on $\mathbf{w}$:

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

- Then $f(\mathbf{x})$ is a Gaussian process

$$\mathbb{E}[f(\mathbf{x})] = \boldsymbol{\phi}(\mathbf{x})^T \mathbb{E}[\mathbf{w}] = \mathbf{0}$$

$$\text{cov}(f(\mathbf{x}), f(\mathbf{x}')) = \mathbb{E}[f(\mathbf{x}) f(\mathbf{x}')] = \boldsymbol{\phi}(\mathbf{x})^T \mathbb{E}[\mathbf{w}\mathbf{w}^T] \boldsymbol{\phi}(\mathbf{x}')$$

$$= \boldsymbol{\phi}(\mathbf{x})^T \boldsymbol{\Sigma}_p \boldsymbol{\phi}(\mathbf{x}')$$

- Thus $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)$ for any N are jointly Gaussian!

University of Amsterdam
Informatics Institute

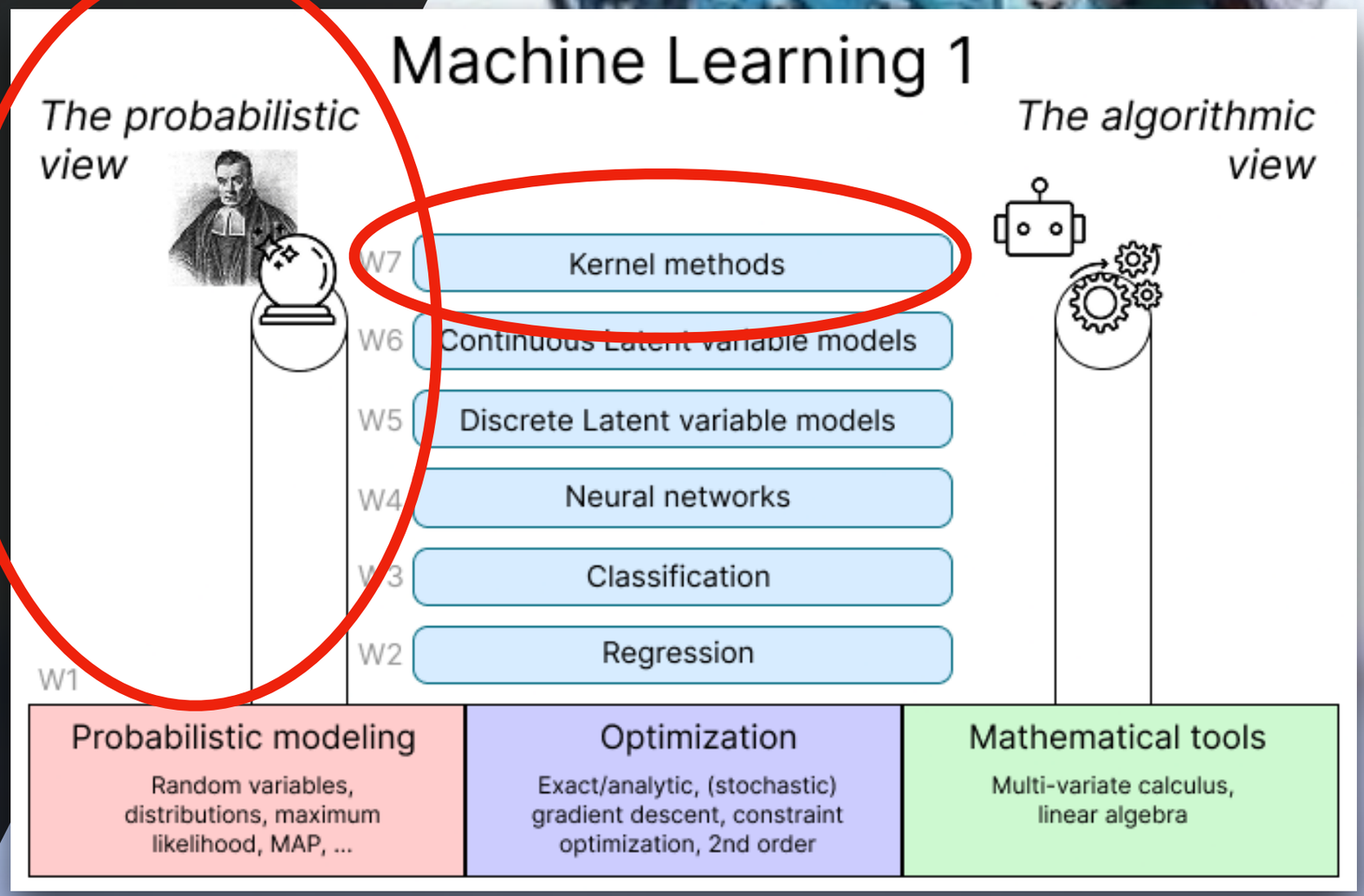
AMLAB
Amsterdam
Machine Learning Lab

Machine Learning 1

Lecture 12.4 - Kernel Methods
Gaussian Processes - With Exponential Kernels

Erik Bekkers

(Bishop 6.4.2)



Slide credits: Patrick Forré and Rianne van den Berg

Image credit: Kirillm | Getty Images

Drawing functions from GP's

- Specifying a kernel determines the characteristics over functions drawn from the GP

- For simplicity, let's take

$$\mathbf{f} = \begin{bmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_N) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \dots & k(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \mathbf{x}_1) & \dots & k(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}\right)$$

$m(\cdot) = 0$

- We consider this kernel

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

linear kernel

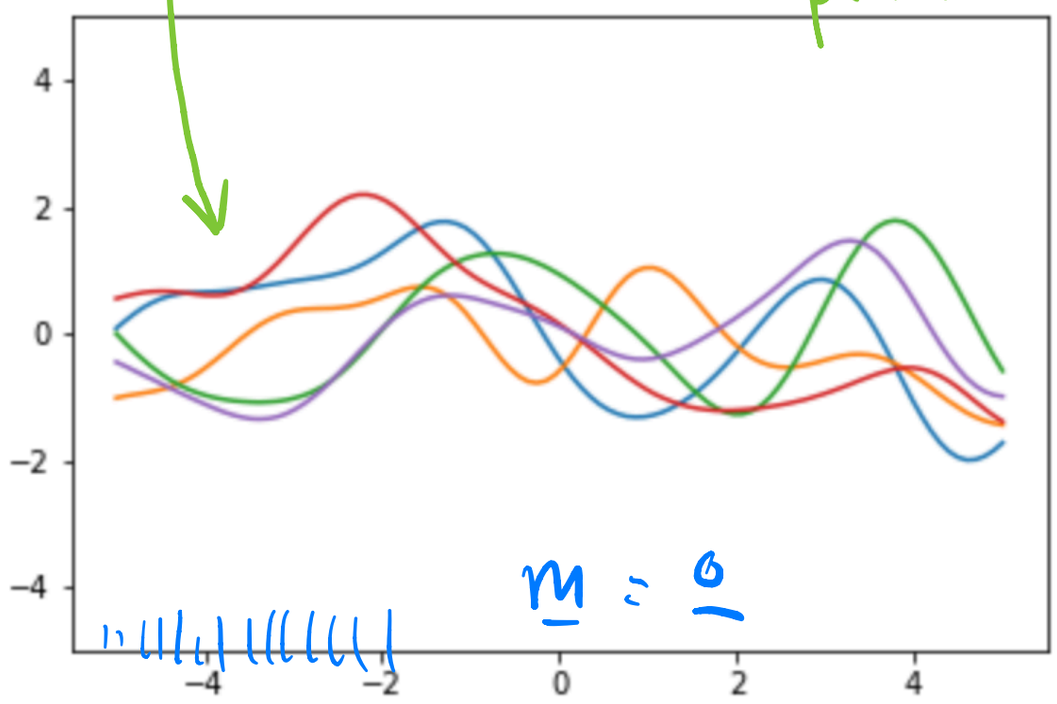
exp

radial basis func.

Drawing functions from GP's

- Sample fine grid of points $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \in [-5,5]$
- Compute $\mathbf{K}$ Gram matrix $\in \mathbb{R}^{N \times N}$
- Reparametrization trick for sampling:
  - Compute $\mathbf{K} = \mathbf{L}\mathbf{L}^T$
  - Sample random vector of size N: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$
  - Sample $\mathbf{f} = \begin{bmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_N) \end{bmatrix} \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$ by computing $\mathbf{f} = \mathbf{L}\mathbf{z}$

draw 5 samples and plot them



Using kernel:

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

$\theta_0 = 1$

$\theta_1 = 1$

$\theta_2 = 0$

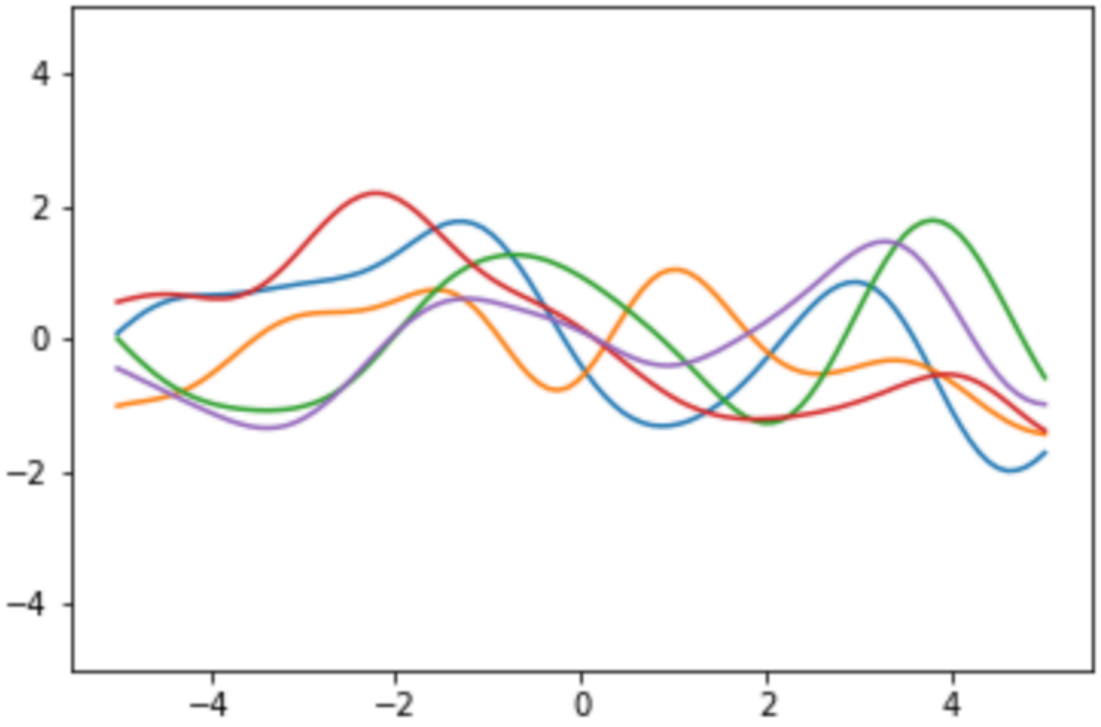
$\theta_3 = 0$

Varying the pre-factor θ_0

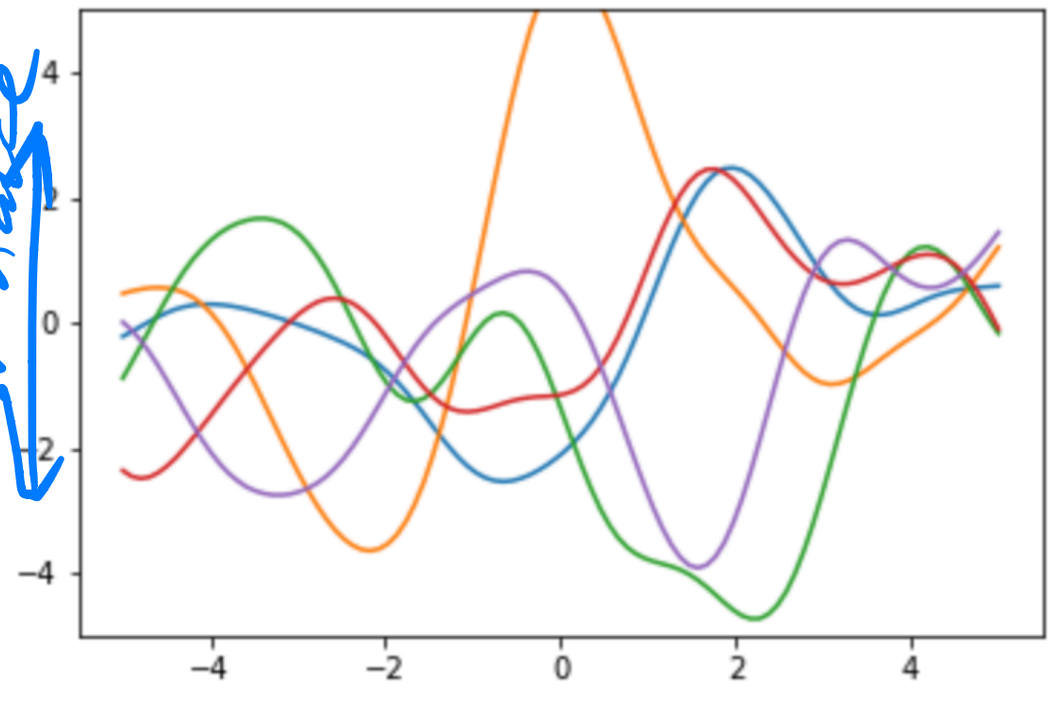
$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

variance regardless of $\underline{x}_n, \underline{x}_m$

$\theta_0 = 1$ $\theta_1 = 1$ $\theta_2 = 0$ $\theta_3 = 0$



$\theta_0 = 4$ $\theta_1 = 1$ $\theta_2 = 0$ $\theta_3 = 0$

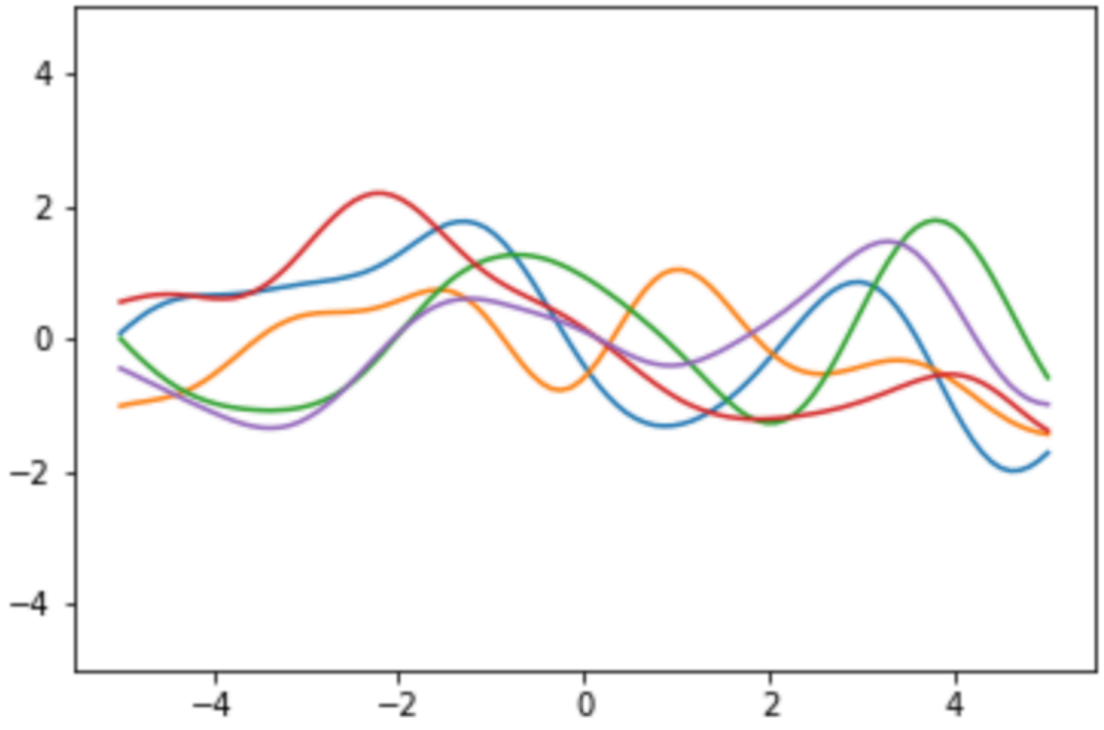


Varying the length scale θ_1

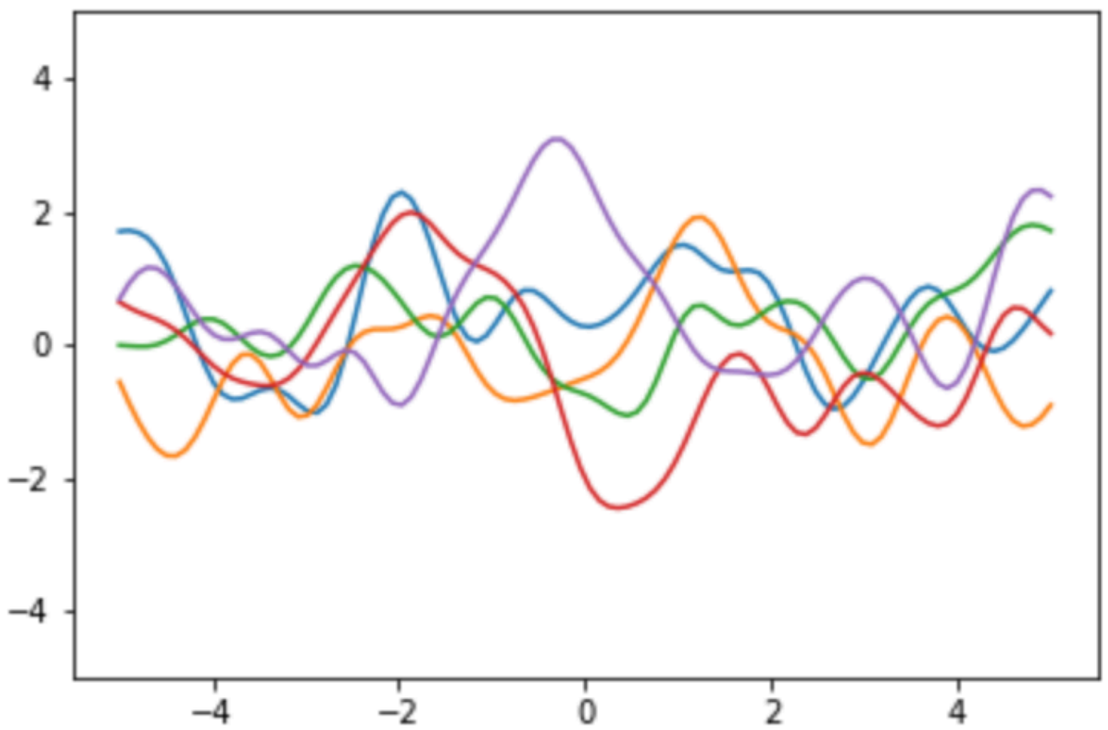
$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

parametrizes how quickly "similarity" decays with distance

$\theta_0 = 1$ $\theta_1 = 1$ $\theta_2 = 0$ $\theta_3 = 0$

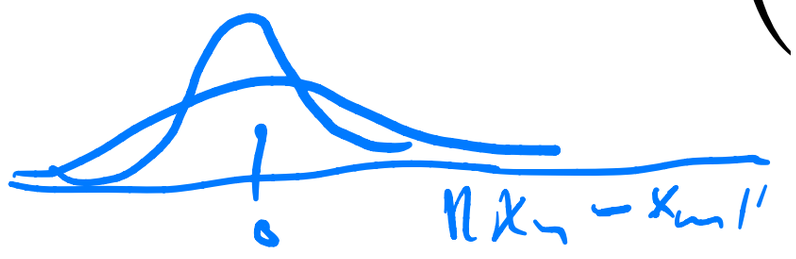


$\theta_0 = 1$ $\theta_1 = 0.5$ $\theta_2 = 0$ $\theta_3 = 0$



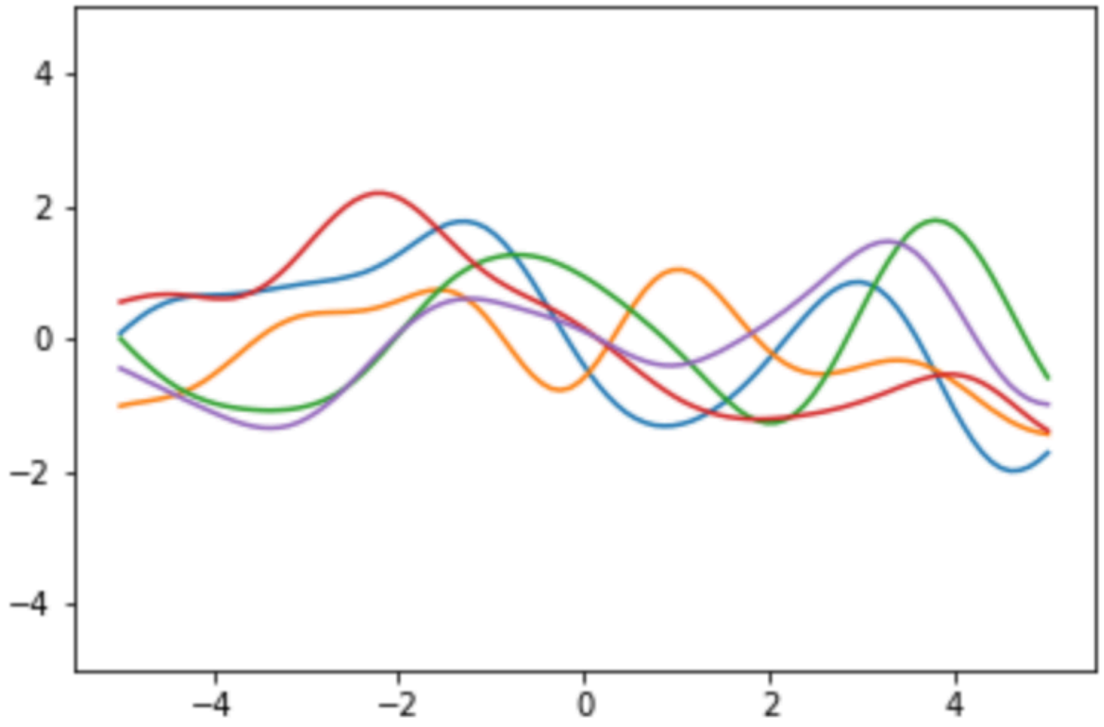
Varying the length scale θ_1

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

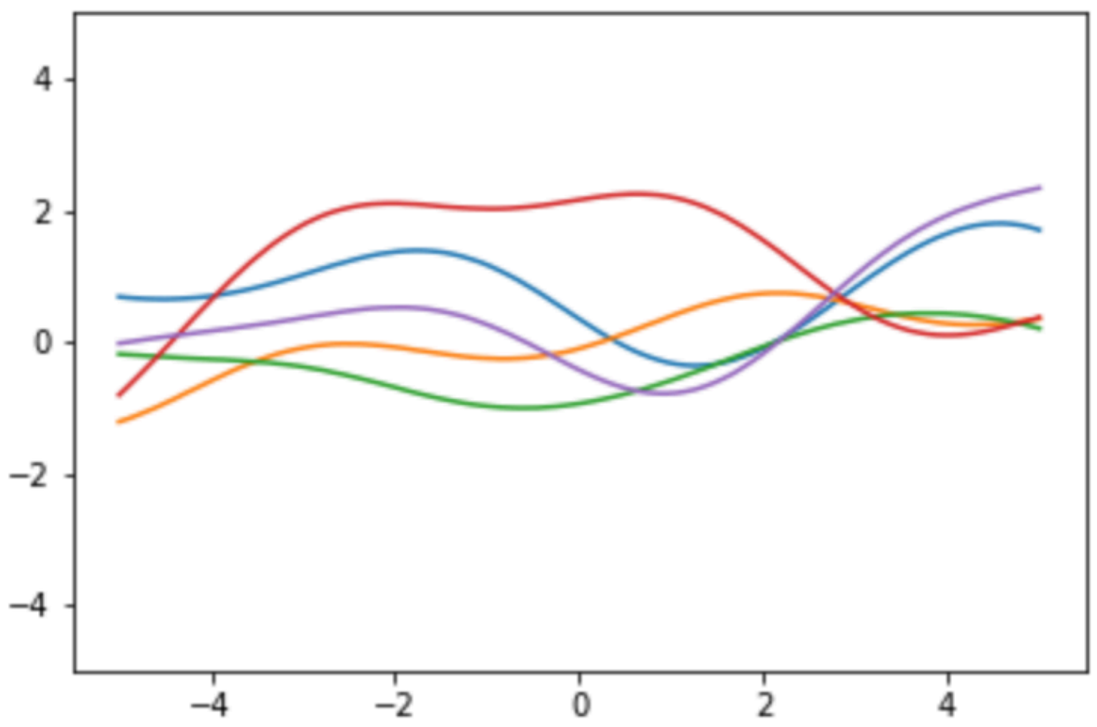


large θ_2 slow decay and thus even distant points vary together

$\theta_0 = 1$ $\theta_1 = 1$ $\theta_2 = 0$ $\theta_3 = 0$



$\theta_0 = 1$ $\theta_1 = 2$ $\theta_2 = 0$ $\theta_3 = 0$

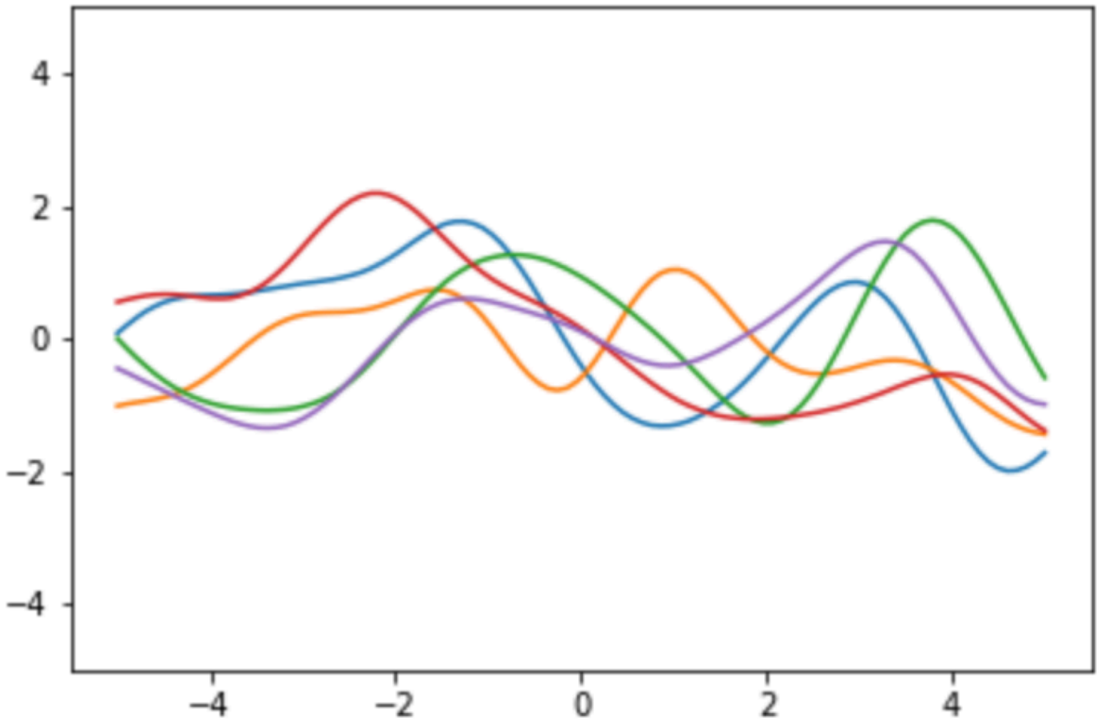


Varying the offset θ_2

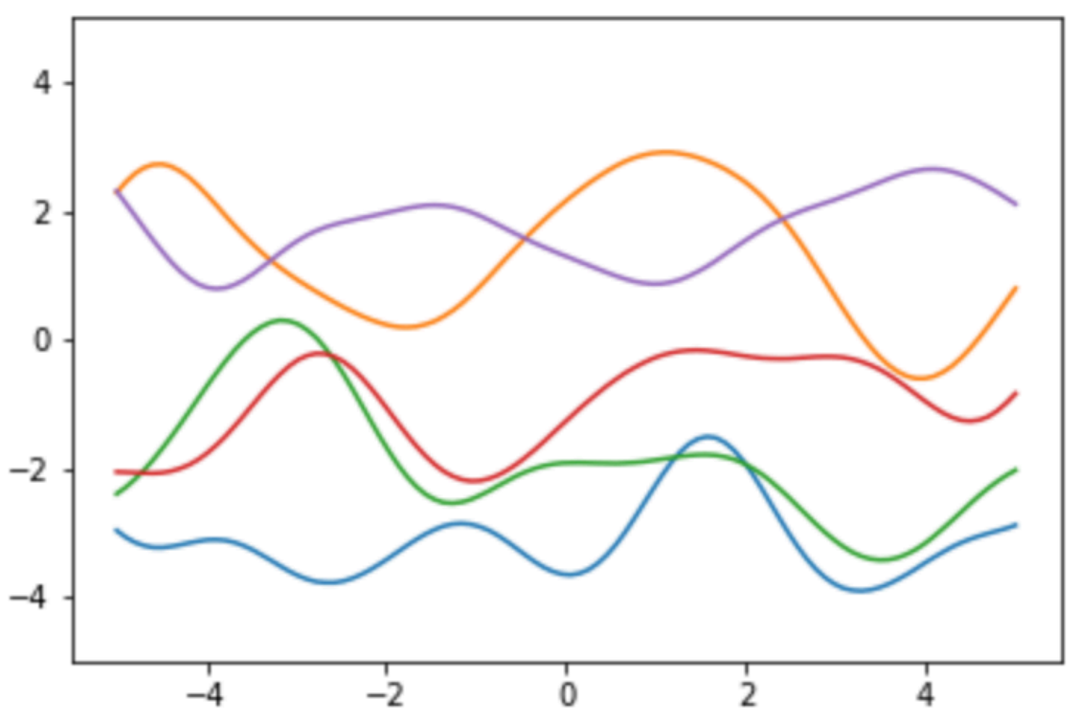
position independent
↓ correlation
not shown

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

$\theta_0 = 1$ $\theta_1 = 1$ $\theta_2 = 0$ $\theta_3 = 0$



$\theta_0 = 1$ $\theta_1 = 1$ $\theta_2 = 5$ $\theta_3 = 0$

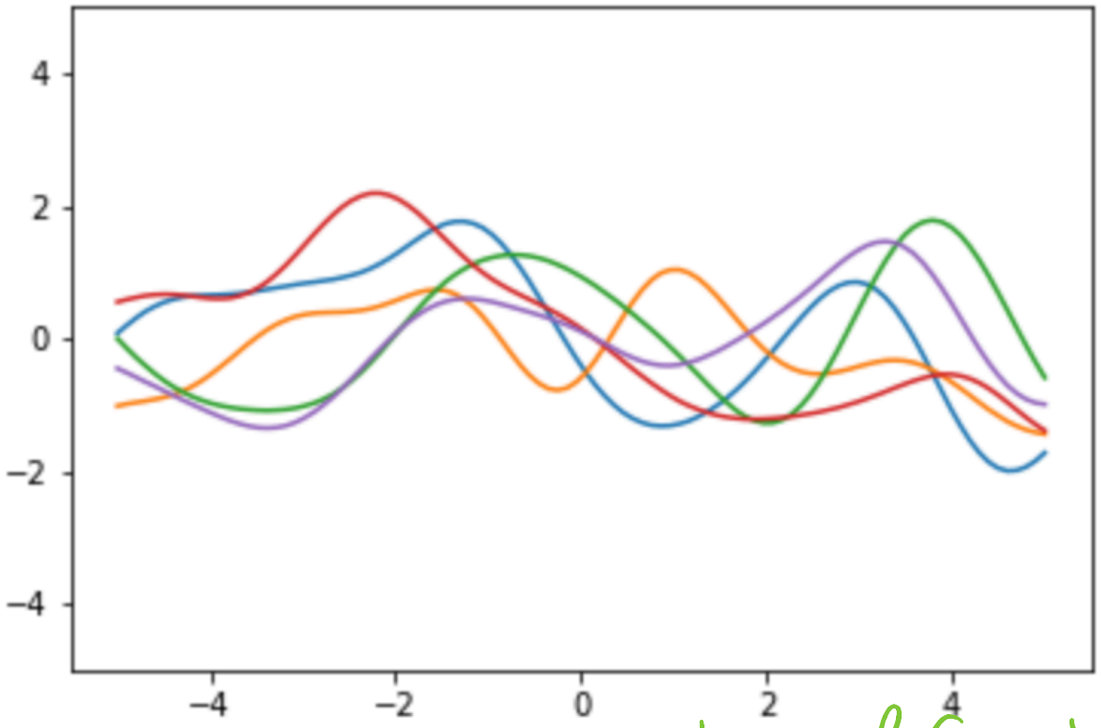


Varying the offset θ_2

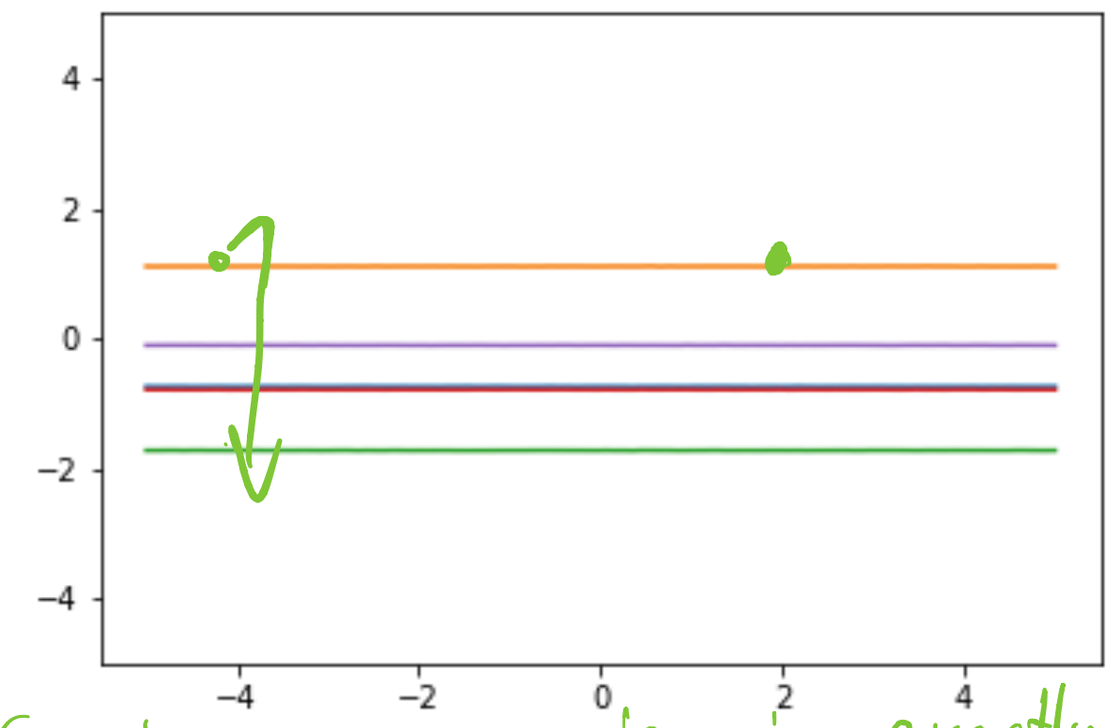
$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

perfect correlation as $\theta_2 \to \infty$

$\theta_0 = 1 \quad \theta_1 = 1 \quad \theta_2 = 0 \quad \theta_3 = 0$



$\theta_0 = 0 \quad \theta_1 = 0 \quad \theta_2 = 5 \quad \theta_3 = 0$



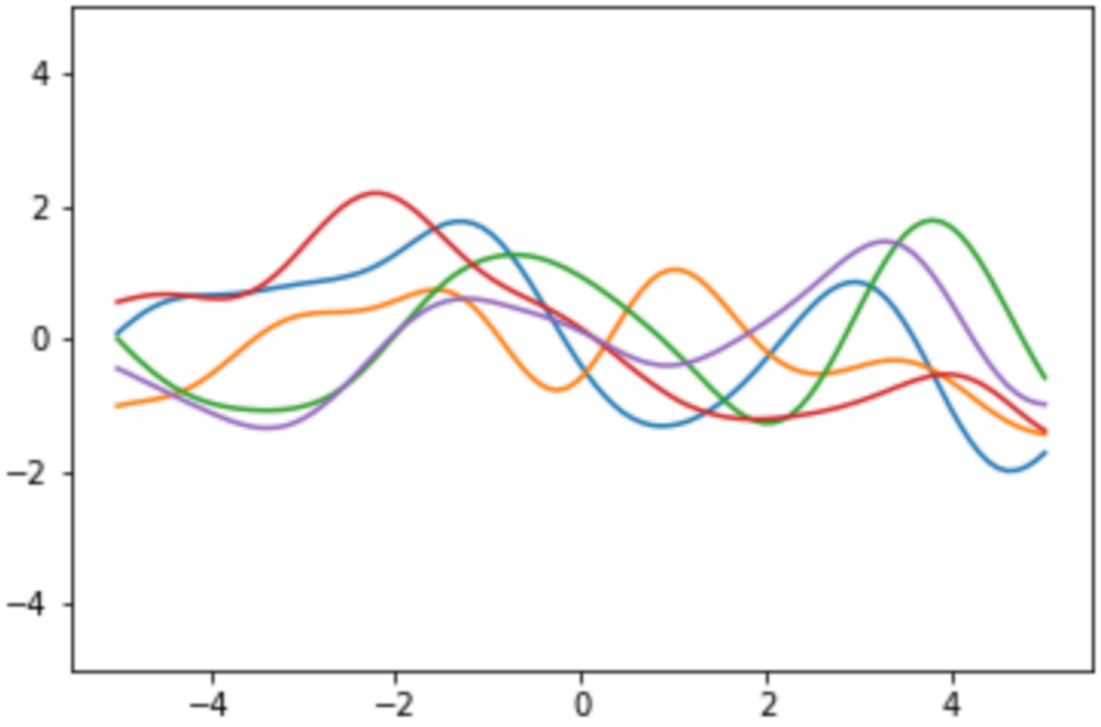
each $f(x_m), f(x_n)$ pair varies in exactly the same way

Varying the linear term θ_3

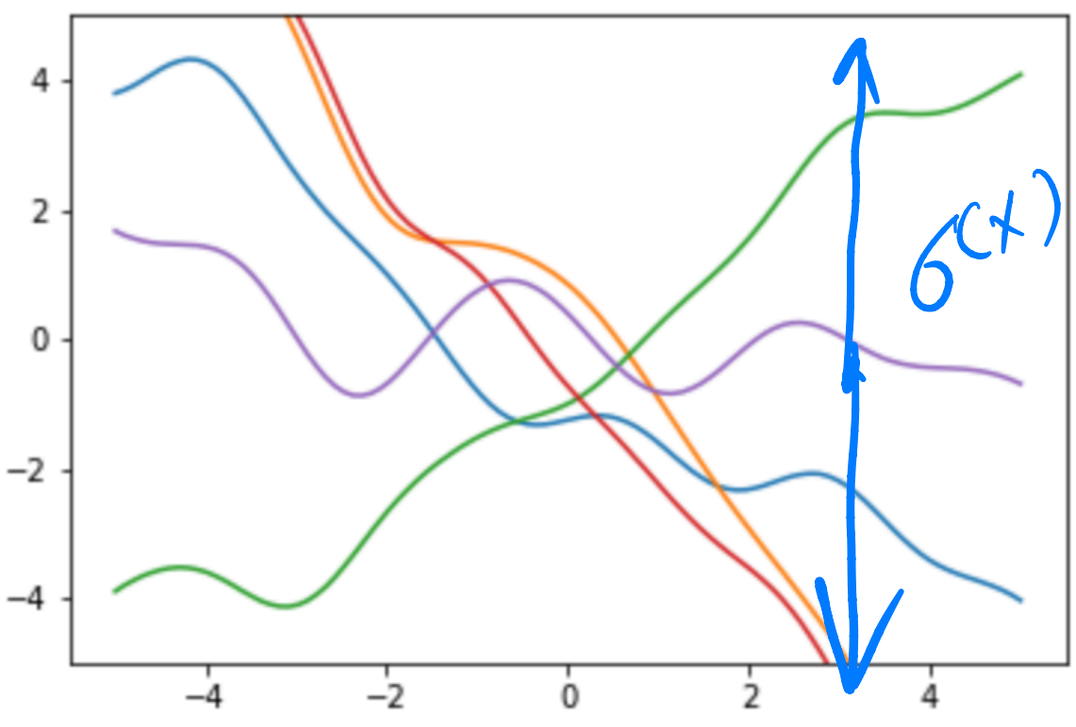
$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

sampled functions $f(\cdot)$ tend to look linear.

$\theta_0 = 1 \quad \theta_1 = 1 \quad \theta_2 = 0 \quad \theta_3 = 0$



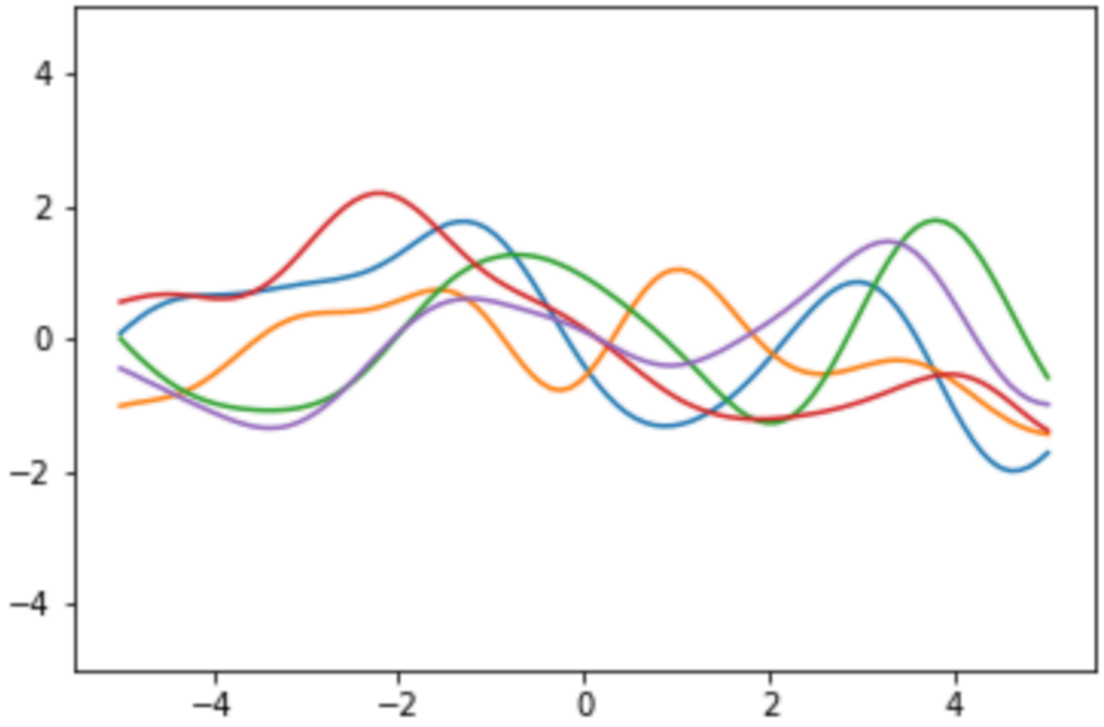
$\theta_0 = 1 \quad \theta_1 = 1 \quad \theta_2 = 0 \quad \theta_3 = 1$



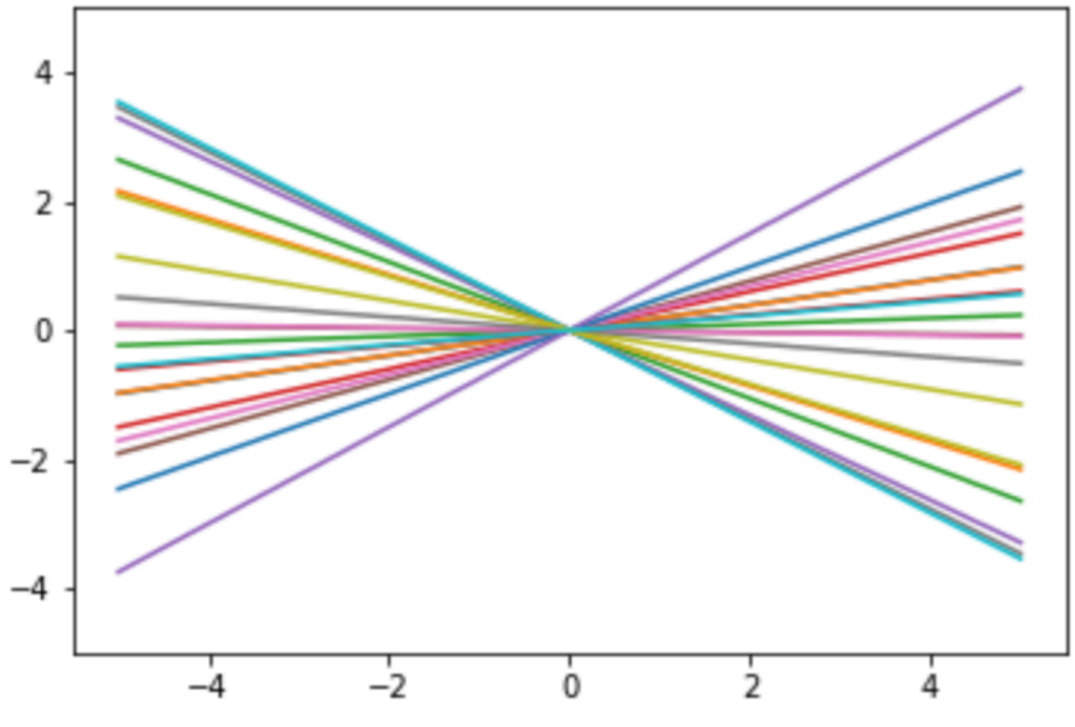
Varying the linear term θ_3

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left(-\frac{1}{2\theta_1^2}||\mathbf{x}_n - \mathbf{x}_m||^2\right) + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m$$

$\theta_0 = 1 \quad \theta_1 = 1 \quad \theta_2 = 0 \quad \theta_3 = 0$



$\theta_0 = 0 \quad \theta_1 = 0 \quad \theta_2 = 0 \quad \theta_3 = 0.2$



University of Amsterdam
Informatics Institute

AMLAB
Amsterdam
Machine Learning Lab

Machine Learning 1

Lecture 12.5 - Kernel Methods
Gaussian Processes - Bayesian Regression

Erik Bekkers

(Bishop 6.4.2, 6.4.3)

Slide credits: Patrick Forré and Rianne van den Berg

Image credit: Kirillm | Getty Images

Regression with GP's

- We have observed $\{(\mathbf{x}_i, f_i)\}_{i=1}^{N}$ where we assume:

no explicit parametrization, assume drawn from a GP

$$f_i = f(\mathbf{x}_i) = y(\mathbf{x}_i) + \varepsilon\,, \qquad \varepsilon \sim \mathcal{N}(0, \beta^{-1})$$

- Assume we have a GP for $y(\mathbf{x})$, so any vector of observations is a Gaussian random variable

Like a prior distribution

choice

$K \in \mathbb{R}^{N \times N}$

choice determin

class of func.

E.g. smooth functs are obtained with large θ_0

$$\mathbf{y} = \begin{bmatrix} y(\mathbf{x}_1) \\ \vdots \\ y(\mathbf{x}_N) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \dots & k(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \mathbf{x}_1) & \dots & k(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}\right)$$

- Then $f(\,\cdot\,)$ is also a GP since $\mathbf{f} = \mathbf{y} + \boldsymbol{\varepsilon}$; the sum of two independent random variables is also Gaussian distributed

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, K(\mathbf{X}, \mathbf{X}) + \beta^{-1}\boldsymbol{I})$$

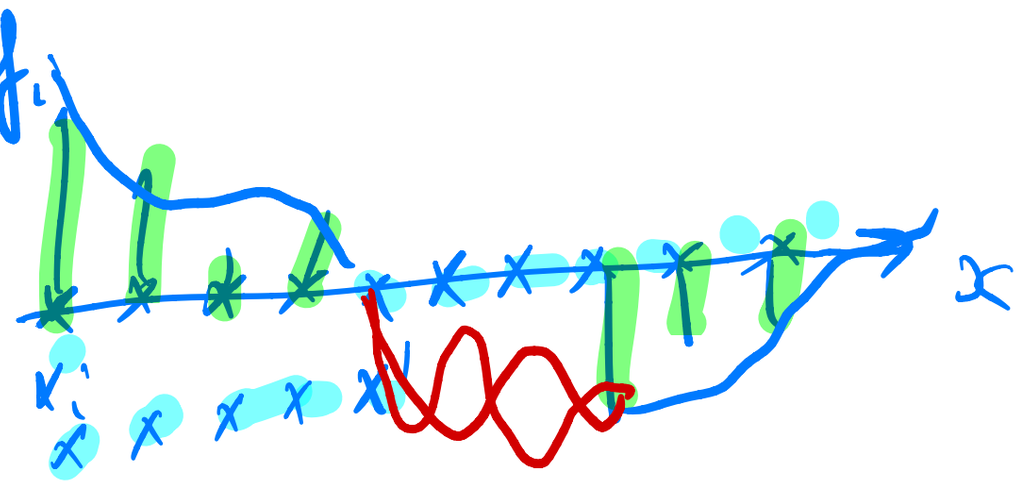
Predictions with GP's

- The joint distribution of test points $\mathbf{f}'$ (at $\mathbf{X}'$) and $\mathbf{f}$ (train points), according to our GP, is given by

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}' \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}(\mathbf{X},\mathbf{X}) + \beta^{-1}\mathbf{I} & \mathbf{K}(\mathbf{X},\mathbf{X}') \\ \mathbf{K}(\mathbf{X}',\mathbf{X}) & \mathbf{K}(\mathbf{X}',\mathbf{X}') + \beta^{-1}\mathbf{I} \end{bmatrix}\right)$$

- Then

$$p(\mathbf{f}' \mid \mathbf{X}', \mathbf{X}, \mathbf{f}) = \mathcal{N}\left(\boldsymbol{\mu}', \boldsymbol{\Sigma}'\right)$$



with (property of Gaussian conditionals, see beginning of lecture)

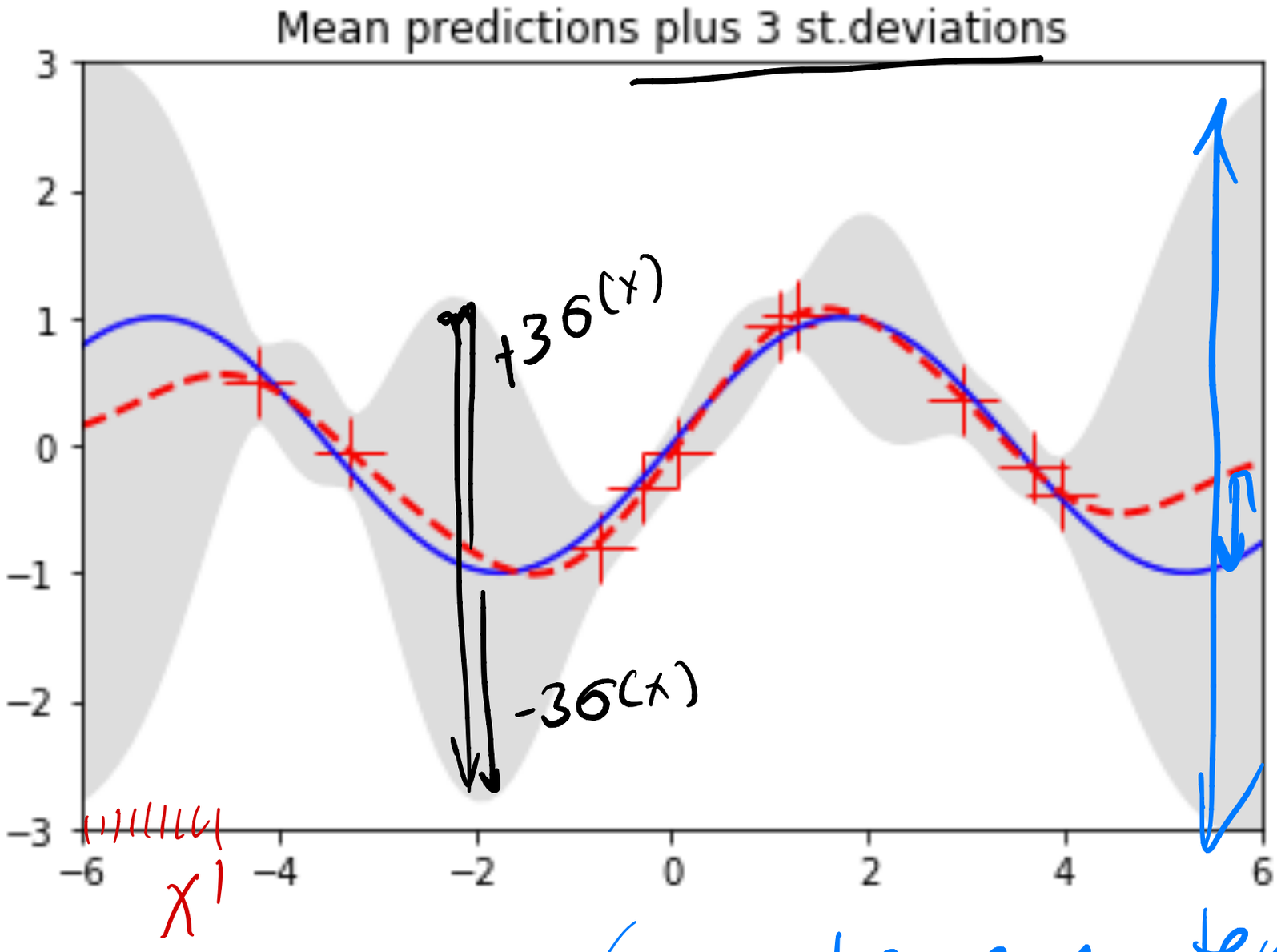
$$\boldsymbol{\mu}' = \mathbf{K}(\mathbf{X}',\mathbf{X})\left(\mathbf{K}(\mathbf{X},\mathbf{X}) + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{f}$$

$$\boldsymbol{\Sigma}' = \mathbf{K}(\mathbf{X}',\mathbf{X}') + \beta^{-1}\mathbf{I} - \mathbf{K}(\mathbf{X}',\mathbf{X})\left(\mathbf{K}(\mathbf{X},\mathbf{X}) + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{K}(\mathbf{X},\mathbf{X}')$$

Inverse required!

Predictions with GP's

active learning
- identify uncertain regions
- collect data in that region

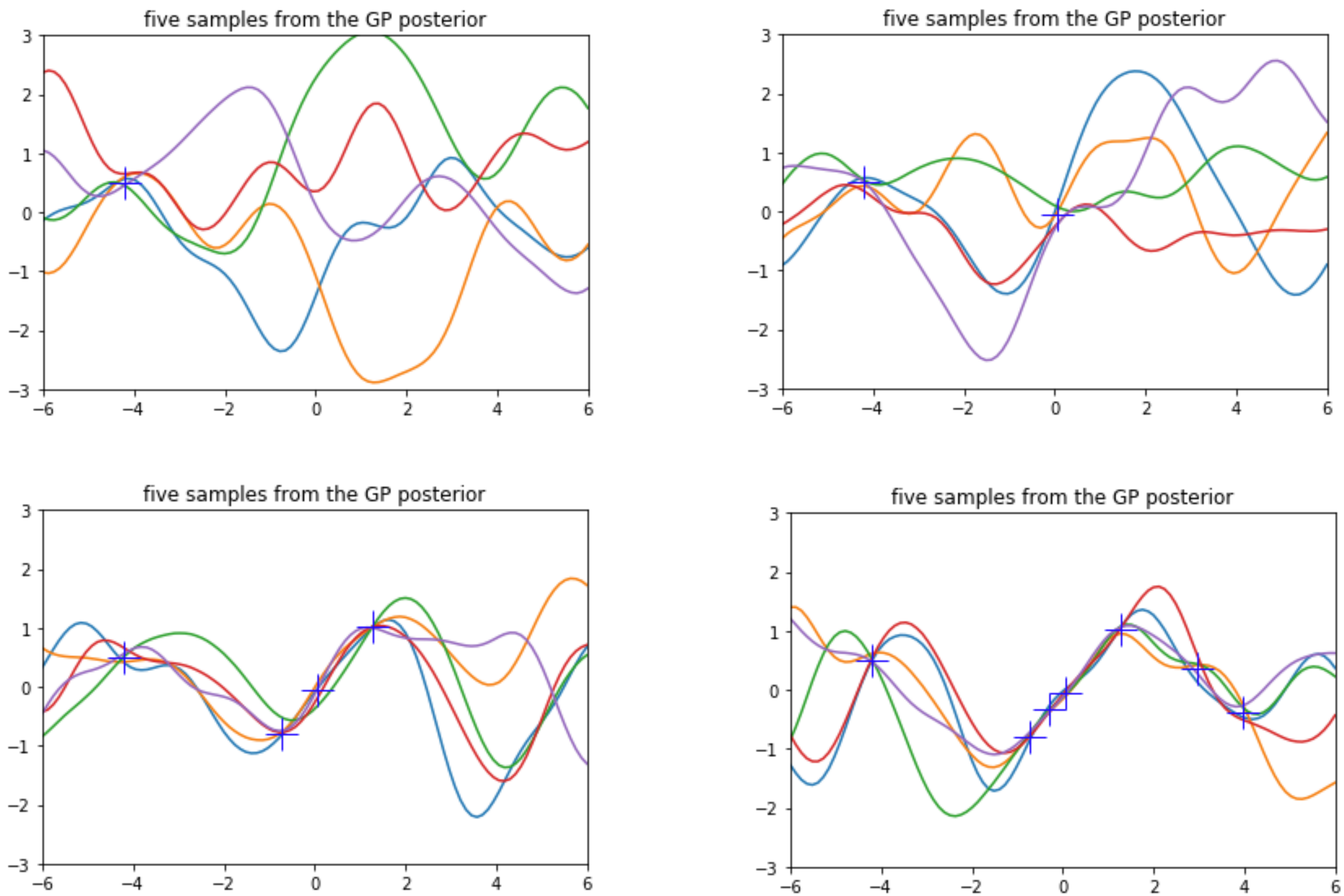


large uncert.

μ' errors true data

make e.g. temperature predicts at geo loc future.
Weather Stations

Drawing functions from GP posterior



How to choose kernel parameters?

- The kernel parameters $\theta_0, \theta_1, \theta_2, \theta_3$ are hyperparameters

- Simplest approach: take training observations, for which we know

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, C_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{X})) = \frac{1}{(2\pi)^{N/2} |\, C_{\boldsymbol{\theta}} \,|^{1/2}} \exp\left(-\frac{1}{2}\mathbf{f}^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} \mathbf{f}\right)$$

with $C_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{X}) = K_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{X}) + \beta^{-1}\boldsymbol{I}$

- Make a maximum likelihood estimate

$$\max_{\boldsymbol{\theta}} \ln p(\mathbf{f} \,|\, \mathbf{X}, \boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} -\frac{1}{2}\ln |\, \mathbf{C}_{\boldsymbol{\theta}} \,| - \frac{1}{2}\mathbf{f}^T \mathbf{C}_{\boldsymbol{\theta}}^{-1} \mathbf{f} - \frac{N}{2}\ln 2\pi$$

- Solve numerically for $\boldsymbol{\theta}$