Gaussian Processes

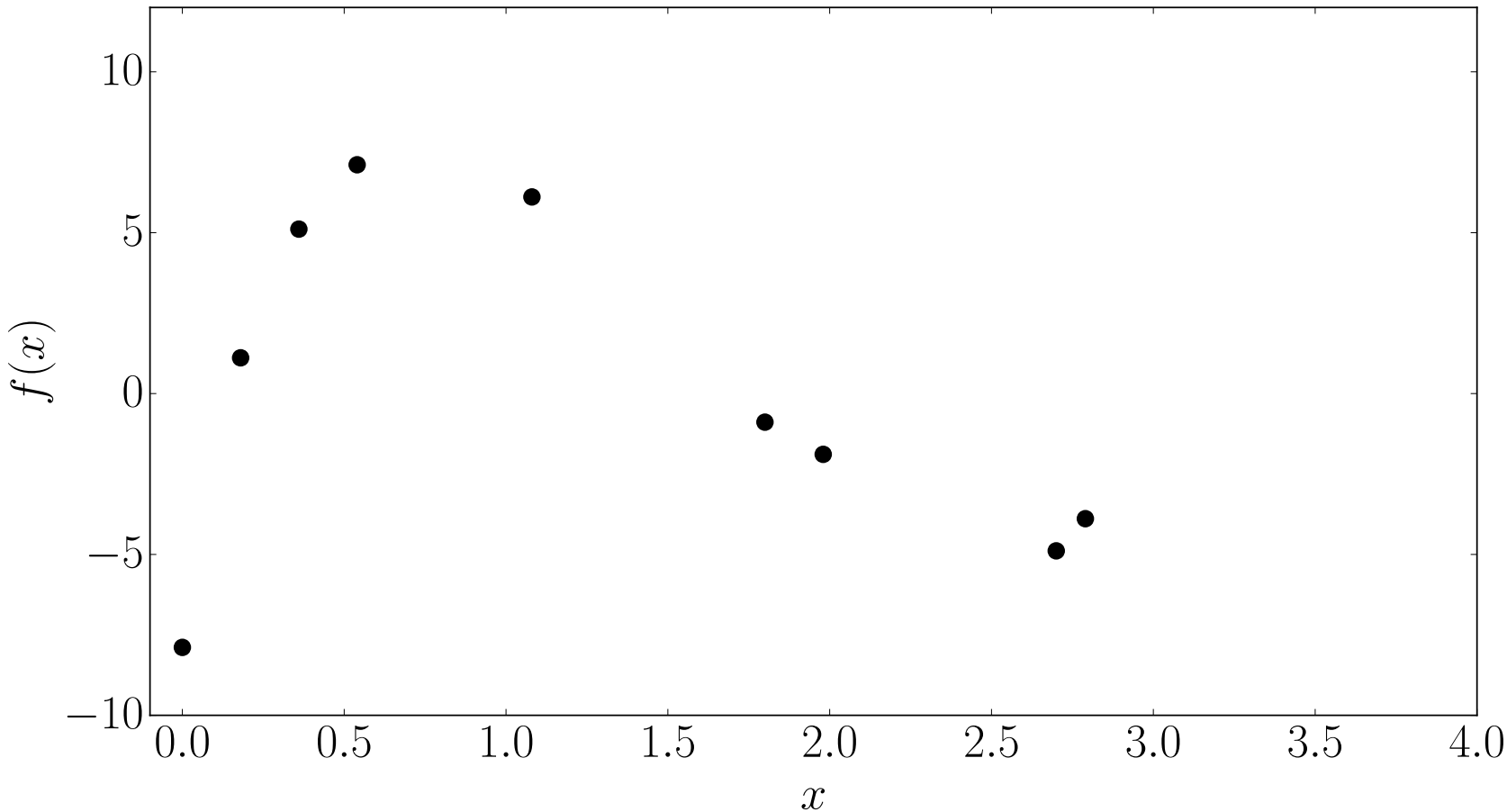
Wiebke Köpp

Technische Universität München

Reading Material:
"Gaussian Processes for Machine Learning" by Rasmussen, Williams [ch. 1, 2]

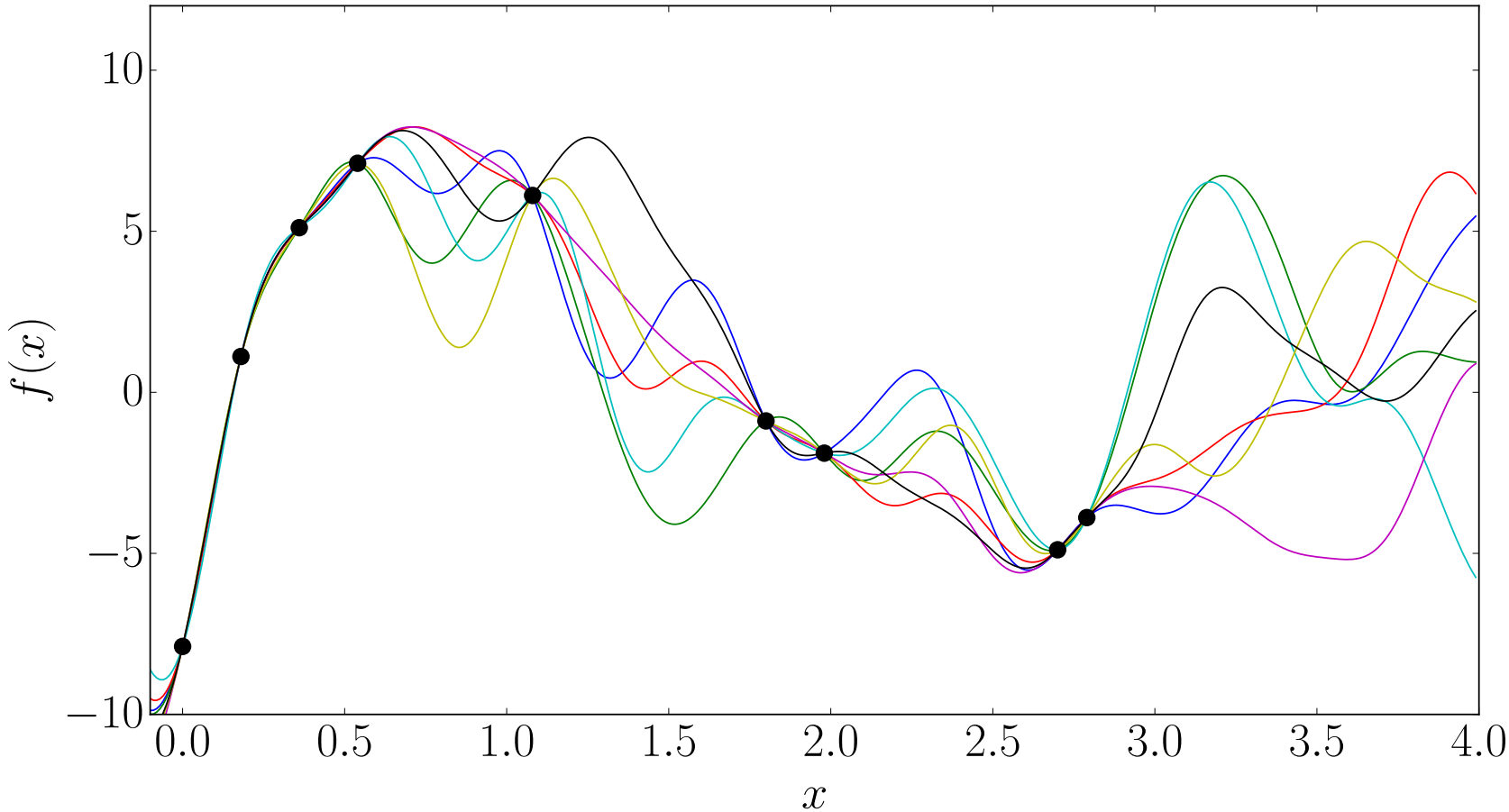
Note: These slides are adapted from slides originally by Daniala Korhammer

Some unknown process



What values can we assume $f(1.8)$ and $f(3.5)$ to be?
How certain are we about this?

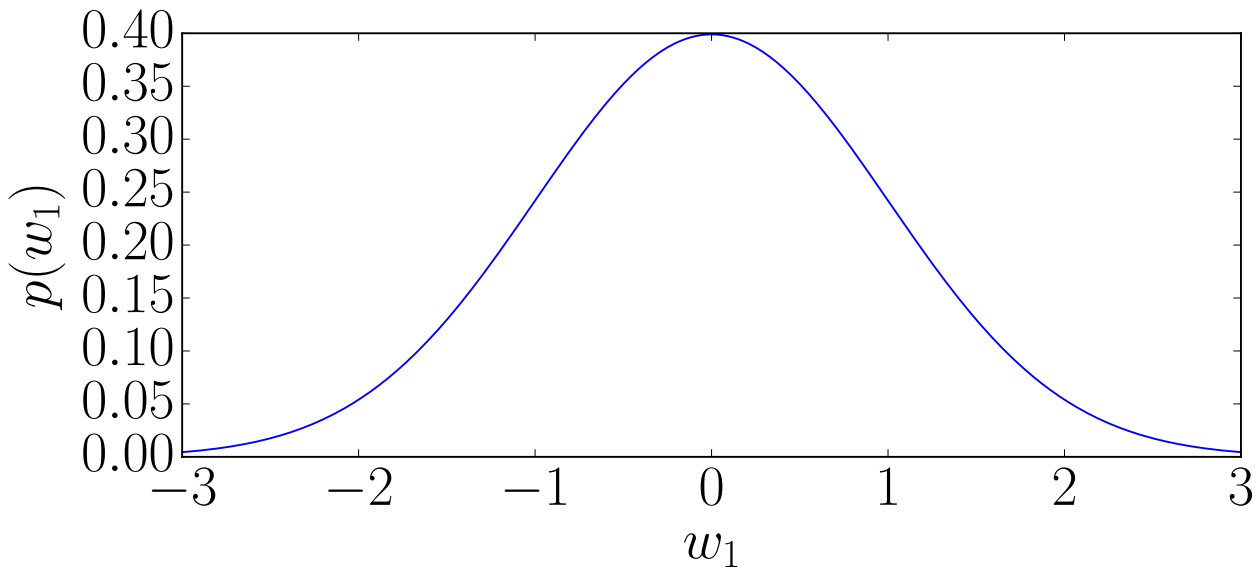
Some functions that are consistent with our data



We're quite certain about $f(1.8)$ because we know the function values of some data points that are close to it. But we don't know anything about $f(3.5)$.

Gaussian for the distribution over vectors

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



Definition of a Gaussian process

Similarly, we can use a Gaussian process to describe a **distribution over functions**:

$$\boldsymbol{f} \sim \mathcal{GP}(m, K)$$

where $m : \mathcal{X} \to \mathbb{R}$ is the **mean function**

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

and $K : \mathcal{X}^2 \to \mathbb{R}$ is the **covariance function**

$$K(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}[(f(\boldsymbol{x}) - m(\boldsymbol{x}))(f(\boldsymbol{x}') - m(\boldsymbol{x}')].$$

For consistency with the kernel lecture, we denote the covariance function K. In the literature you will also find κ or k.

GPs define multivariate Gaussian distributions

We have data points $\boldsymbol{X} = [\boldsymbol{x}_1^T, \cdots, \boldsymbol{x}_n^T]^T$ and are interested in their function values $\boldsymbol{f}(\boldsymbol{X}) = (f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n))^T$.

> A Gaussian process is a collection of random variables (RV), any finite number of which have **joint Gaussian distribution**.

$\boldsymbol{f}$ is one such subset of RV and has (prior) joint Gaussian distribution:

$$\boldsymbol{f}(\boldsymbol{X}) \sim \mathcal{N}(\boldsymbol{m}(\boldsymbol{X}), \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})),$$

Idea of the Gaussian Process (informally!)

Our basic assumption is that if vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ are similar, then $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$ should be similar, too.

The covariance function $K(\boldsymbol{x}, \boldsymbol{x}')$ returns a measure of the similarity of $\boldsymbol{x}$ and $\boldsymbol{x}'$ that also encodes how similar $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$ should be.

The mean function $m(\boldsymbol{x})$ encodes the a priori expectation of the (unknown) function.

For inference we condition the unknown function values on the known ones. If there are no "similar" known values, then the mean function dominates the result.

Setting the mean function

In most cases we simply use

$$\mathbb{E}[f(\boldsymbol{x})] = m(\boldsymbol{x}) = 0,$$

which makes sense especially if we normalize the output to zero mean.

Formal properties of the covariance function

The covariance function $K(\boldsymbol{x}, \boldsymbol{x}')$ needs to be a measure of similarity between $\boldsymbol{x}$ and $\boldsymbol{x}'$. This is the basic assumption which makes inference possible, since we assume that similar data points have similar function values.

K needs to be symmetric

$$K(\boldsymbol{x}, \boldsymbol{x}') = K(\boldsymbol{x}', \boldsymbol{x})$$

and positive semidefinite

$$\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} K(\boldsymbol{x}, \boldsymbol{x}') g(\boldsymbol{x}) g(\boldsymbol{x}') \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{x}' \geq 0,$$

for all $g \in L_2$ (Mercer's theorem).

Setting the covariance function

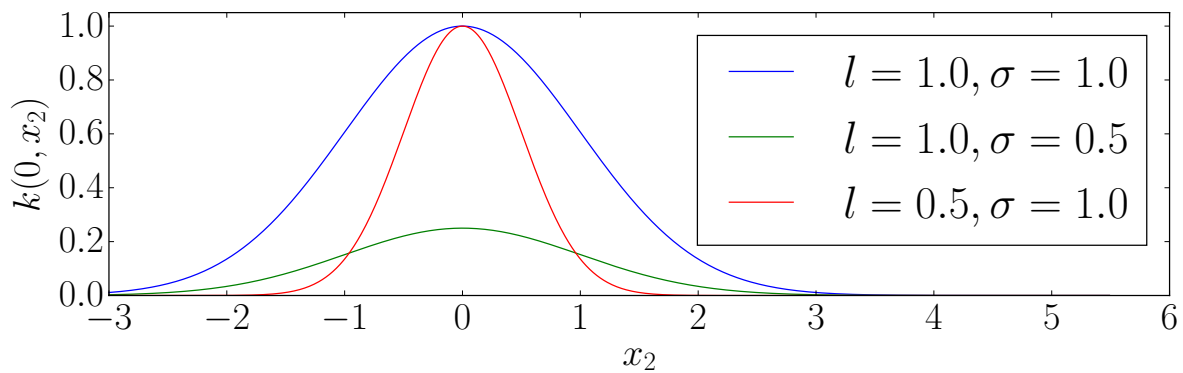
The "default" covariance function is the squared exponential kernel

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{(\boldsymbol{x} - \boldsymbol{x}')^T(\boldsymbol{x} - \boldsymbol{x}')}{2l^2}\right), \tag{1}$$

where l varies the length (or width) and σ the height of the kernel.

Note that $\boldsymbol{x}$ can be in any domain, if K defines a good measure of the similarity of two vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$.

The covariance function is what drives the behavior of a GP!



What is encoded in the covariance matrix?

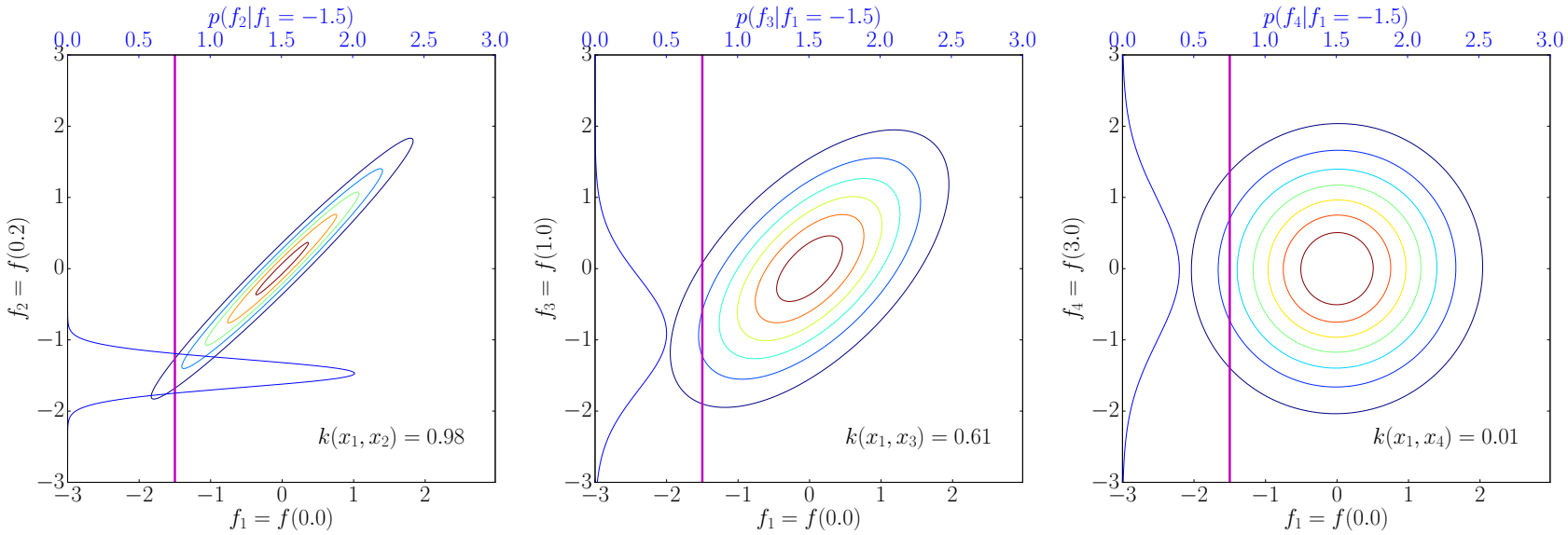
4-datapoint example with $\boldsymbol{X} = \{0, 0.2, 1, 3\}$ and a simple squared exponential kernel with $l = 1$ and $\sigma = 1$:

$$\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) = \begin{pmatrix} K(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1, \boldsymbol{x}_4) \\ \vdots & \ddots & \vdots \\ K(\boldsymbol{x}_4, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_4, \boldsymbol{x}_4) \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0.98 & 0.61 & 0.011 \\ 0.98 & 1 & 0.73, & 0.020 \\ 0.61 & 0.73 & 1 & 0.14 \\ 0.011 & 0.020 & 0.14 & 1 \end{pmatrix}$$

Remember: It is not $\boldsymbol{X}$, over which the Gaussian is defined, but $\boldsymbol{f}(\boldsymbol{X})$!

What is encoded in the covariance matrix?



If two points are similar (left plot), they covary strongly. Knowing about f_1, reveals a lot about f_2.

If two points are far apart (right plot), their covariance is small. Knowing the value of f_1 reveals little about f_4.

In the extreme case where the covariance is 0, the conditional and marginal distributions are the same.

Drawing samples from an MVN

Generate a D-dimensional vector $\boldsymbol{u}$ by drawing D samples (independently) from $\mathcal{N}(0, 1)$.

Perform the Cholesky decomposition $\boldsymbol{\Sigma} = \boldsymbol{L}\boldsymbol{L}^T$, where $\boldsymbol{L}$ is a lower triangular matrix.

Compute $\boldsymbol{y} = \boldsymbol{\mu} + \boldsymbol{L}\boldsymbol{u}$, where $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

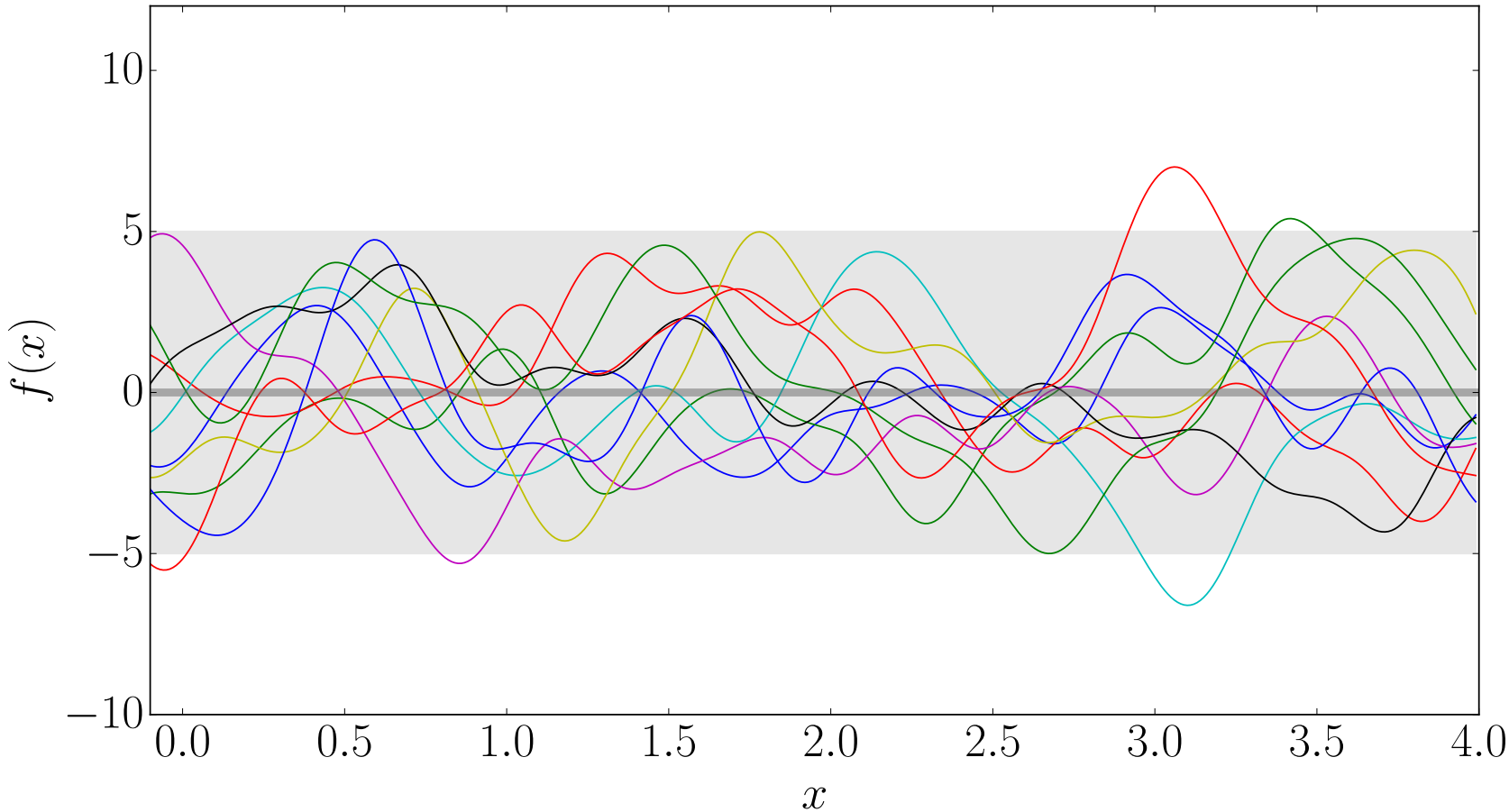
Sampling from the **prior distribution** of a GP at arbitrary points $\boldsymbol{X}_*$

$$\boldsymbol{f}_{\text{pri}}(\boldsymbol{X}_*) \sim \mathcal{GP}(\boldsymbol{m}(\boldsymbol{X}_*), \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*))$$

is equivalent to sampling **from an MVN**:

$$\boldsymbol{f}_{\text{pri}}(\boldsymbol{X}_*) \sim \mathcal{N}\left(\boldsymbol{m}(\boldsymbol{X}_*), \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*)\right).$$

Drawing samples from the prior



10 samples from the prior distribution using a squared exponential kernel (Eq. 1) with $l = 0.2$ and $\sigma = 2.5$. The dark grey line indicates $m(x)$, the gray area indicates the 95% confidence region, i.e. $m(x) \pm 2\sqrt{K(x,x)} = m(x) \pm 2\sigma$.

Inference with GPs - noise-free case

We have training data $\boldsymbol{X}$ (of size $n \times D$), corresponding observations $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{X})$, and test data points $\boldsymbol{X}_*$ ($n_* \times D$) for which we want to infer function values $\boldsymbol{f}_* = \boldsymbol{f}(\boldsymbol{X}_*)$.

The GP defines a joint distribution for $p(\boldsymbol{f}, \boldsymbol{f}_* \mid \boldsymbol{X}, \boldsymbol{X}_*)$:

$$\begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}_* \end{bmatrix}, \begin{bmatrix} \boldsymbol{K} & \boldsymbol{K}_* \\ \boldsymbol{K}_*^T & \boldsymbol{K}_{**} \end{bmatrix} \right),$$

with $\boldsymbol{\mu} = \boldsymbol{m}(\boldsymbol{X}), \boldsymbol{\mu}_* = \boldsymbol{m}(\boldsymbol{X}_*)$,

$\boldsymbol{K} = \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})$, $\boldsymbol{K}_* = \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*)$, $\boldsymbol{K}_{**} = \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*)$.

To infer $\boldsymbol{f}_*$ or rather

$$p\left(\boldsymbol{f}_* \mid \boldsymbol{X}_*, \boldsymbol{X}, \boldsymbol{f}\right),$$

we need to apply the rules for conditioning multivariate Gaussians.

Conditionals of an MVN

Suppose $\boldsymbol{y}$ has joint Gaussian distribution:

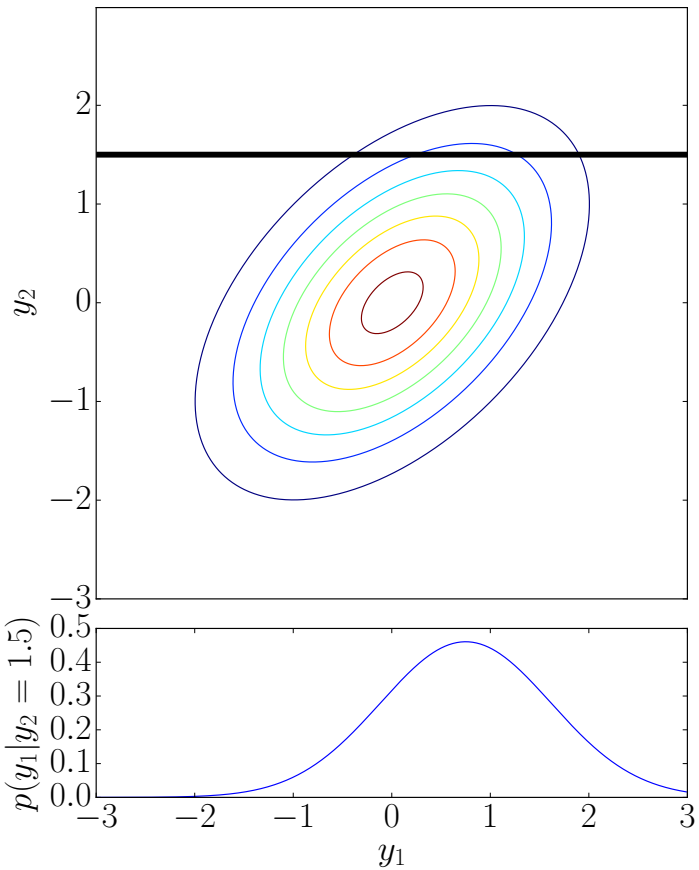
$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}\right)$$

The posterior conditional $p(\boldsymbol{y}_2 \mid \boldsymbol{y}_1)$ is then given by

$$\boldsymbol{y}_2 | \boldsymbol{y}_1 \sim \mathcal{N}(\boldsymbol{\mu}_{2|1}, \boldsymbol{\Sigma}_{2|1})$$

$$\boldsymbol{\mu}_{2|1} = \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}(\boldsymbol{y}_1 - \boldsymbol{\mu}_1)$$

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

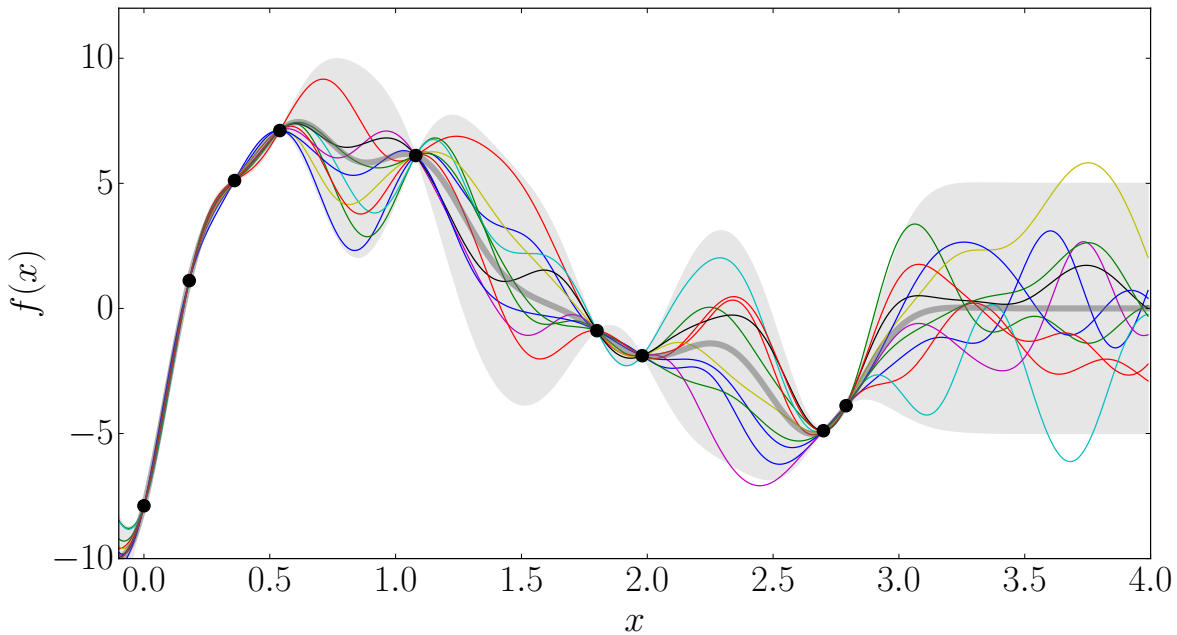


Inference with GPs - noise-free case (2)

Applying the above MVN conditionals to our problem yields:

$$\boldsymbol{f}_*|\boldsymbol{f}, \boldsymbol{X}, \boldsymbol{X}_* \sim \mathcal{N}\left(\boldsymbol{\mu}_* + \boldsymbol{K}_*^T \boldsymbol{K}^{-1}(\boldsymbol{f} - \boldsymbol{\mu}), \boldsymbol{K}_{**} - \boldsymbol{K}_*^T \boldsymbol{K}^{-1} \boldsymbol{K}_*\right)$$

We can draw samples from this distribution or compute the expectation:



10 samples from the posterior distribution using a squared exponential kernel (Eq. 1) with $l = 0.2$ and $\sigma = 2.5$. The dark grey line indicates $\mathbb{E}[f(x)]$, the grey area indicates the 95% confidence region.

Inference with GPs - noisy case

In the noise-free case (with some kernels, such as the SE kernel) the GP acts as an interpolator between observed values.

More often than not the assumption that our observations y_i correspond exactly to the function values $f(\boldsymbol{x}_i) = f_i$ is wrong.

We will now instead assume, that we observe a noisy version of the underlying function:

$$y_i = f_i + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, \sigma_y^2)$ is additive iid Gaussian noise.

Inference with GPs - noisy case (2)

Let's look at some GP with $m(\boldsymbol{x}) = 0$ and $K(\boldsymbol{x}, \boldsymbol{x}') = 1$ if $\boldsymbol{x} = \boldsymbol{x}'$.

In the noise-free case, where $y_i = f_i$:

$$\begin{pmatrix} y_i \\ f_i \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}\right),$$

then $y_i | f_i \sim \mathcal{N}(f_i, 0)$ is a degenerate Gaussian with zero variance.

In the noisy scenario we want $y_i | f_i \sim \mathcal{N}(f_i, \sigma_y^2)$, so we assume instead:

$$\begin{pmatrix} y_i \\ f_i \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} 1 + \sigma_y^2 & 1 \\ 1 & 1 \end{pmatrix}\right).$$

Inference with GPs - noisy case (3)

We can easily extend this idea to $\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{f}(\boldsymbol{X}_*) = \boldsymbol{f}_* \end{bmatrix}$ for arbitrary $\boldsymbol{X}_*$. Since the individual noise terms ϵ are independent we have to add a scaled identity matrix $\sigma_y^2 \boldsymbol{I}$.

The joint distribution in the noisy case is then

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}_* \end{bmatrix}, \begin{bmatrix} \boldsymbol{K} + \sigma_y^2 \boldsymbol{I} & \boldsymbol{K}_* \\ \boldsymbol{K}_*^T & \boldsymbol{K}_{**} \end{bmatrix} \right),$$

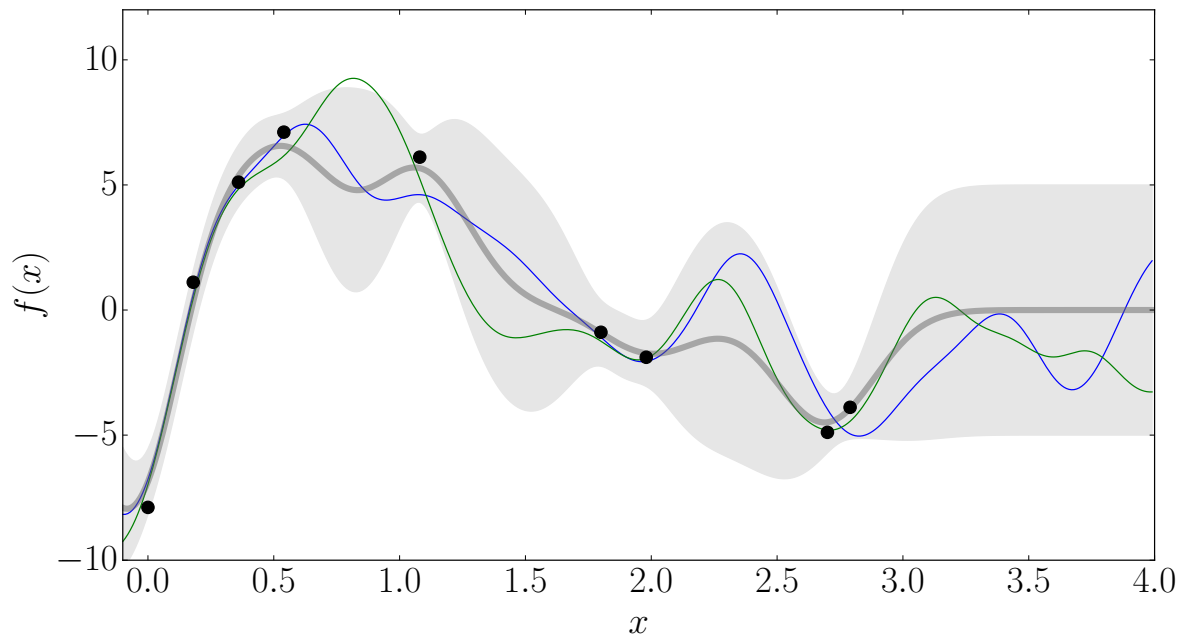
and the conditional (predictive) distribution

$$\begin{aligned} \boldsymbol{f}_* | \boldsymbol{y}, \boldsymbol{X}, \boldsymbol{X}_* \sim \mathcal{N}(&\boldsymbol{\mu}_* + \boldsymbol{K}_*^T [\boldsymbol{K} + \sigma_y^2 \boldsymbol{I}]^{-1} (\boldsymbol{y} - \boldsymbol{\mu}), \\ &\boldsymbol{K}_{**} - \boldsymbol{K}_*^T [\boldsymbol{K} + \sigma_y^2 \boldsymbol{I}]^{-1} \boldsymbol{K}_*). \end{aligned}$$

Inference with GPs - noisy case (4)

Obviously, we will use the expectation as our point prediction:

$$\hat{\boldsymbol{f}}_* = \mathbb{E}[\boldsymbol{f}_* | \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{X}_*] = \boldsymbol{\mu}_* + \boldsymbol{K}_*^T [\boldsymbol{K} + \sigma_y^2 \boldsymbol{I}]^{-1} (\boldsymbol{y} - \boldsymbol{\mu})$$
$$\text{cov}(\boldsymbol{f}_*) = \boldsymbol{K}_{**} - \boldsymbol{K}_*^T [\boldsymbol{K} + \sigma_y^2 \boldsymbol{I}]^{-1} \boldsymbol{K}_*.$$



$\mathbb{E}[f(x)]$, confidence region and two samples. $\sigma_y = .5$, other parameters as before.

A Gaussian Process is a non-parametric model

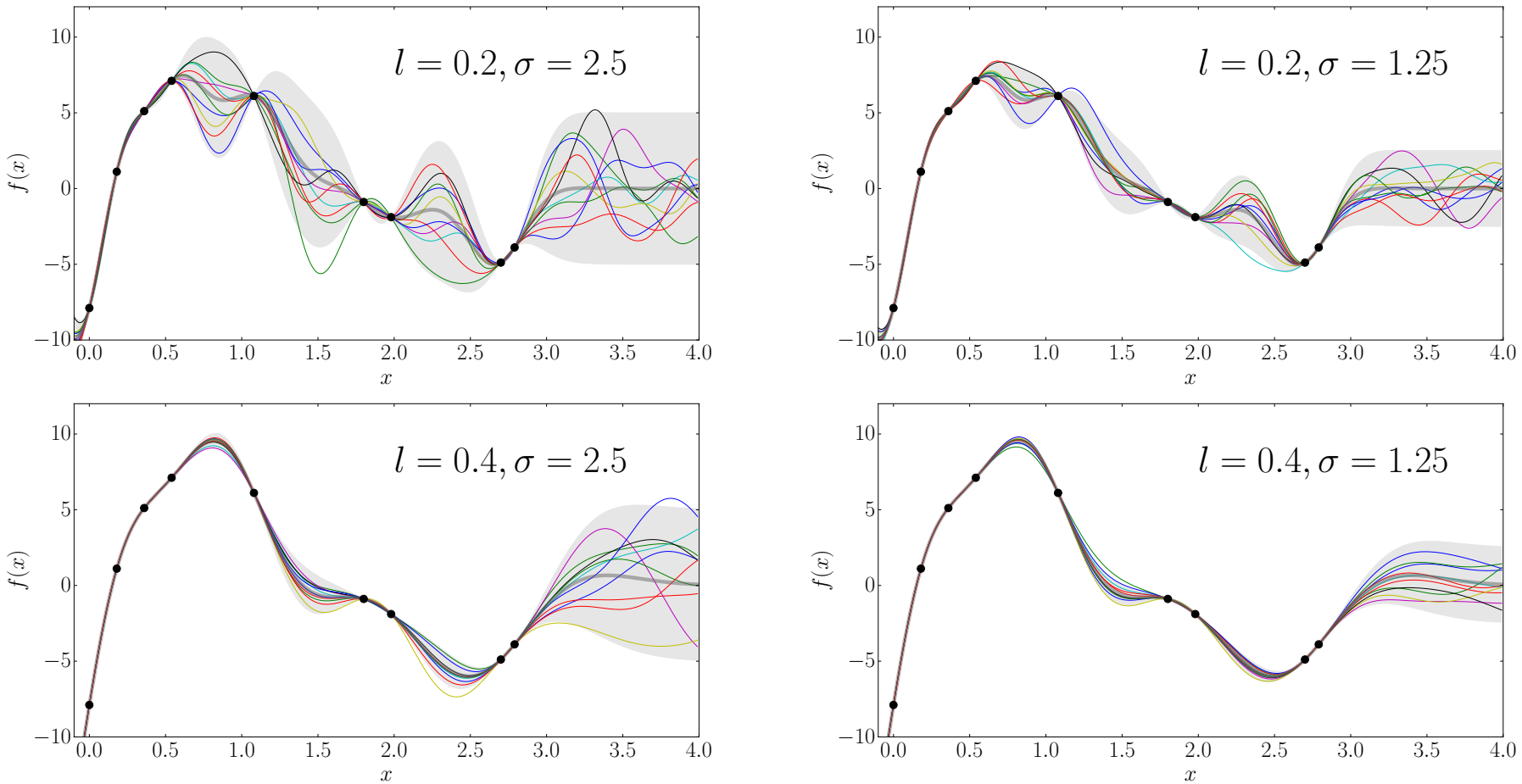
The distinction between non-parametric and parametric models is not always clear. One of the definitions:

The number of parameters in a parametric model is fixed before training, while in non-parametric models it grows with the number of training samples.

Non-parametric models often require no or little training (e.g., kNN), while for parametric models (e.g., Linear Regression) training is typically more expensive than inference.

GPs are an example of a non-parametric model.

Influence of hyper parameters



10 samples from the prior distribution using a squared exponential kernel (Eq. 1) with different settings for l and σ. The dark grey line indicates $m(x)$, the gray area indicates the 95% confidence region, i.e. $m(x) \pm 2\sqrt{K(x,x)} = m(x) \pm 2\sigma$.

What we learned

- Gaussian Processes: a probabilistic model that can fit arbitrary functions
- Mean and covariance function
- Inference in the noise-free and noisy case

Out of scope:

- Gaussian processes for classification
- Learning hyper parameters (e.g., parameters of the covariance function, noise)
- Handling of large data sets