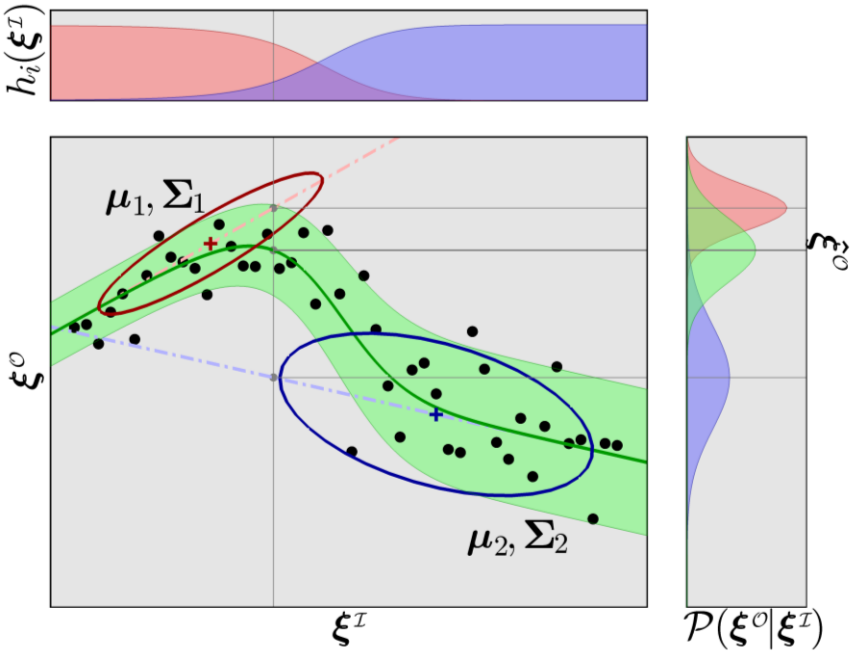
EE613
Machine Learning for Engineers

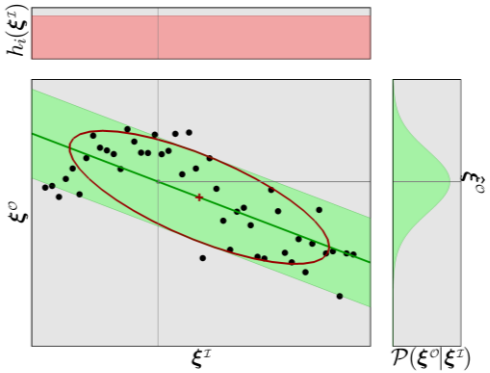
NONLINEAR REGRESSION II

Sylvain Calinon
Robot Learning & Interaction Group
Idiap Research Institute
Dec. 19, 2019

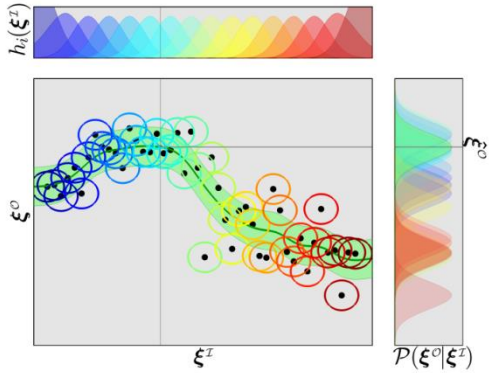
Gaussian mixture regression (GMR)



Least squares linear regression



Nadaraya-Watson kernel regression



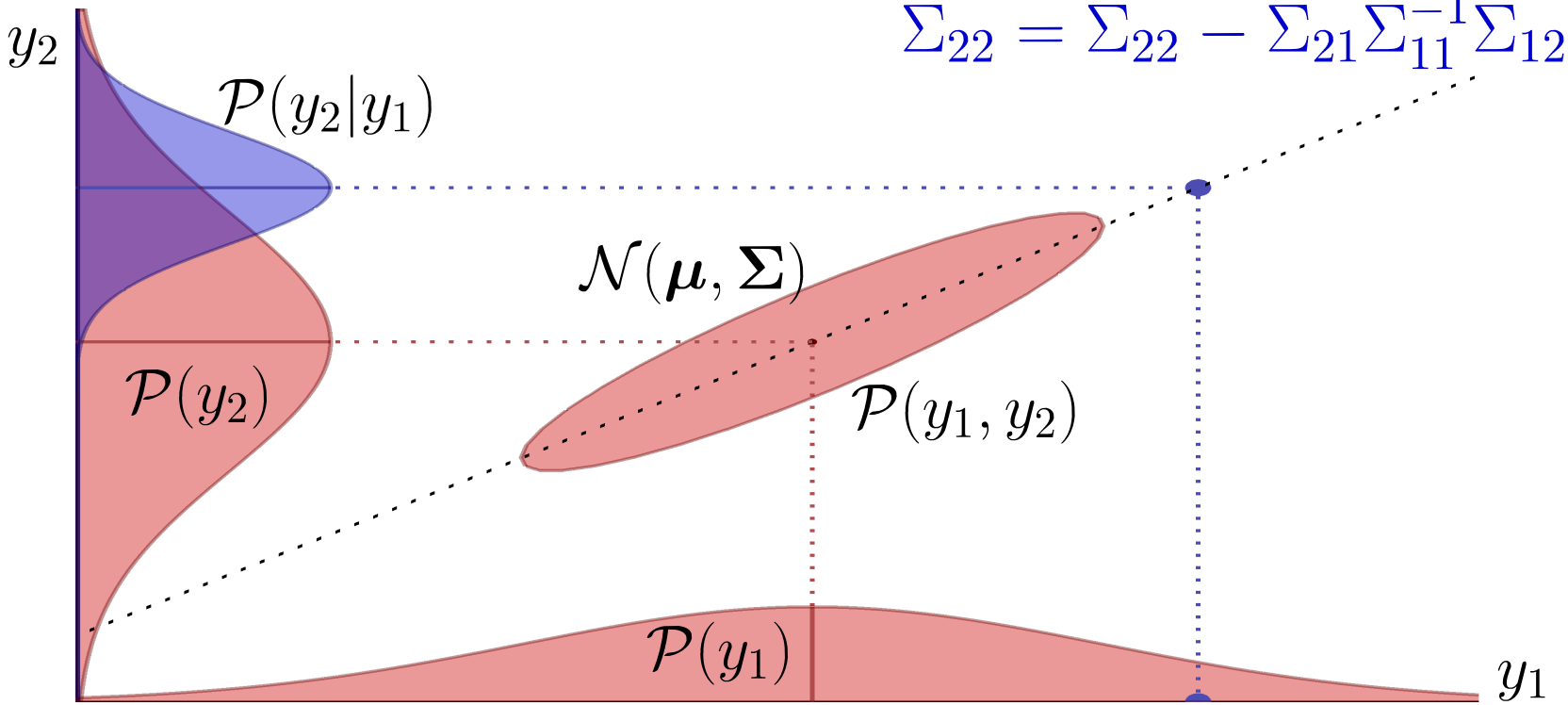
Gaussian process (GP)

Gaussian process - Informal interpretation

- A joint distribution represented by a bivariate Gaussian forms marginal distributions P(y_1) and P(y_2) that are unidimensional.
- Observing y_1 changes our belief about y_2, giving rise to a **conditional distribution**.
- Knowledge of the covariance lets us shrink uncertainty in one variable based on the observation of the other.

$$\hat{y}_2 = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1)$$

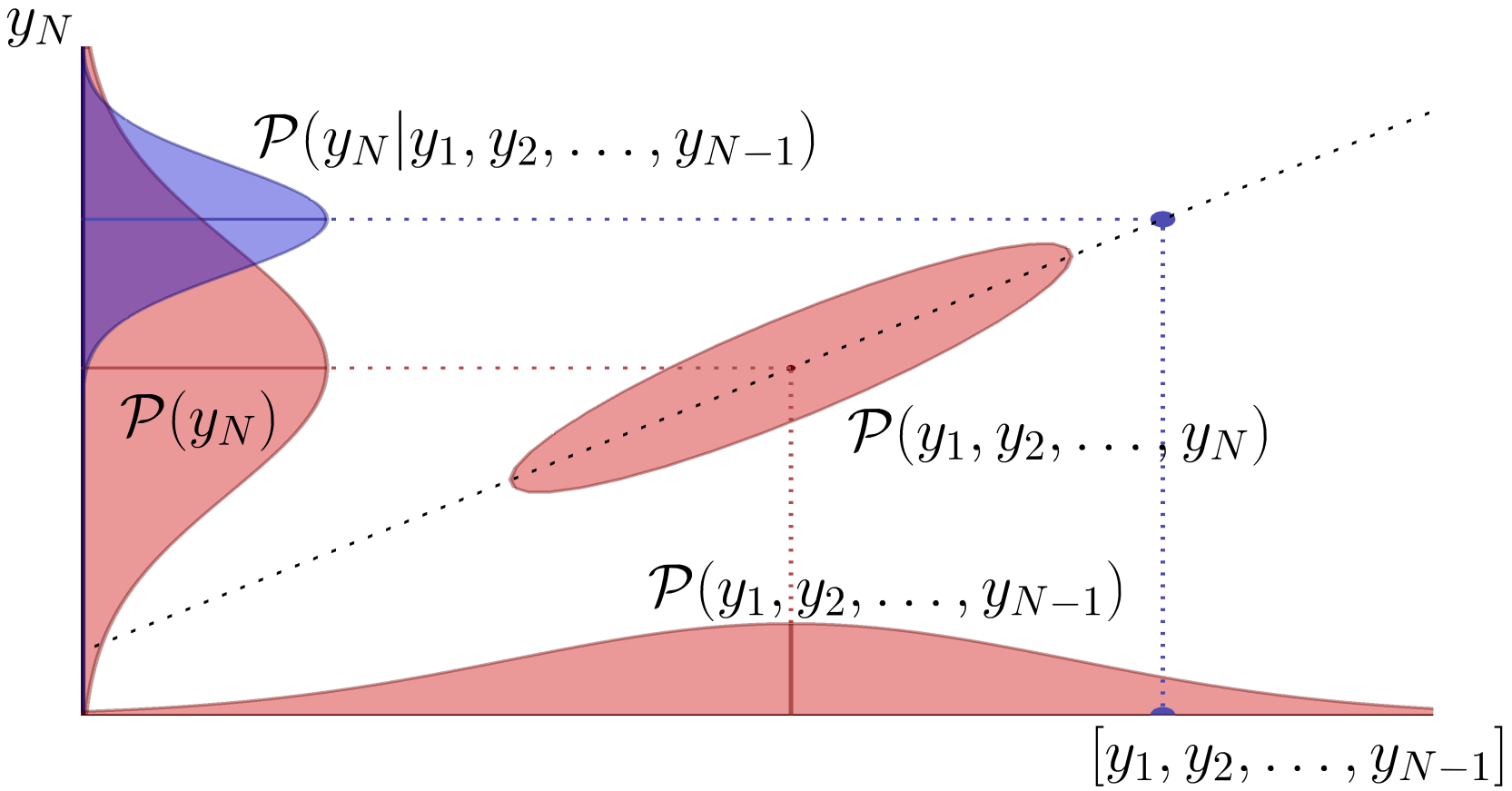
$$\hat{\Sigma}_{22} = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$$



Gaussian process - Informal interpretation

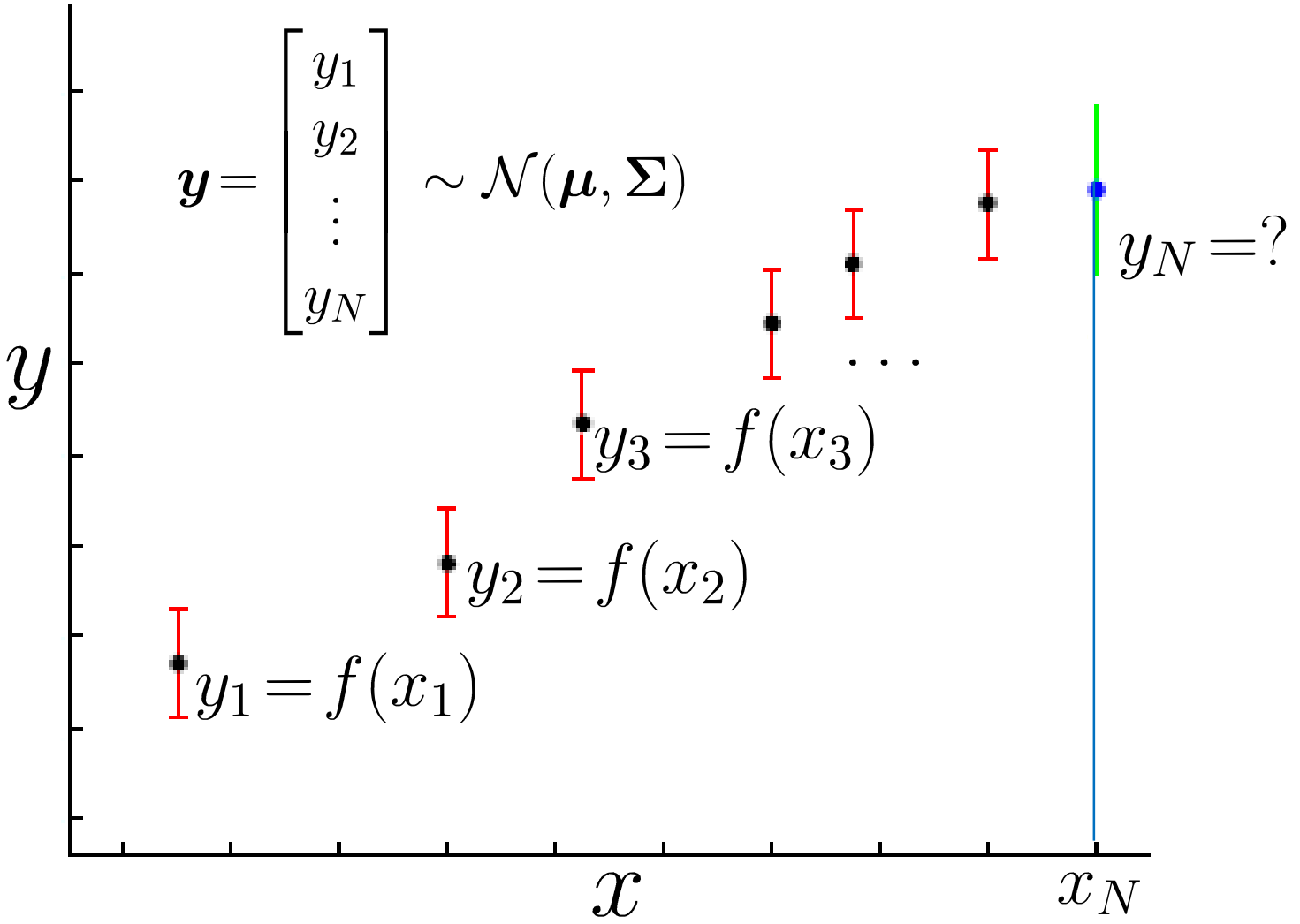
- This bivariate example can be extended to an arbitrarily large number of variables.
- Indeed, observations in an arbitrary dataset can always be imagined as a single point sampled from a multivariate Gaussian distribution.

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

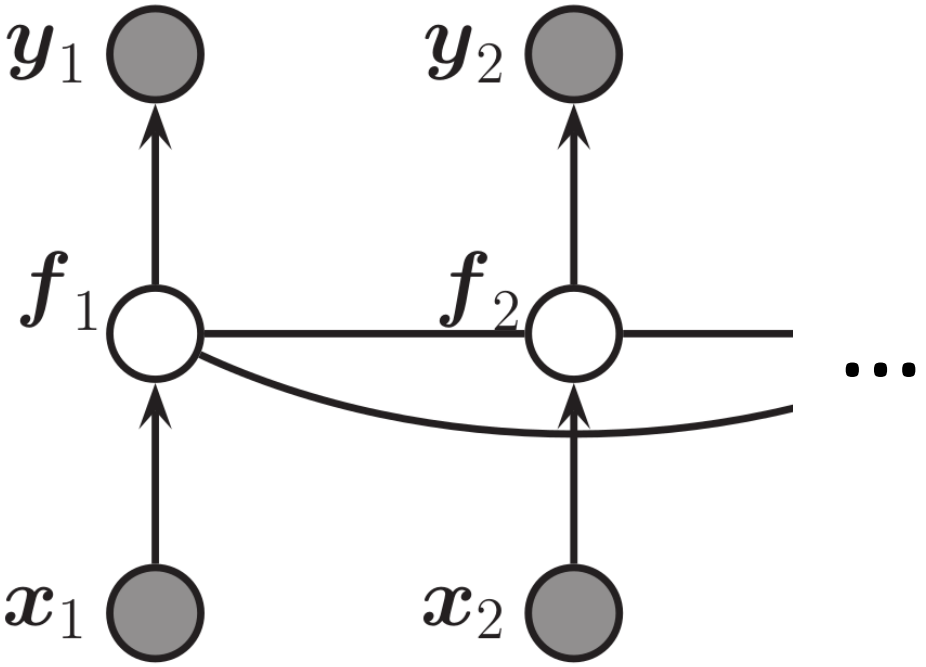


How to construct this joint distribution in GP?

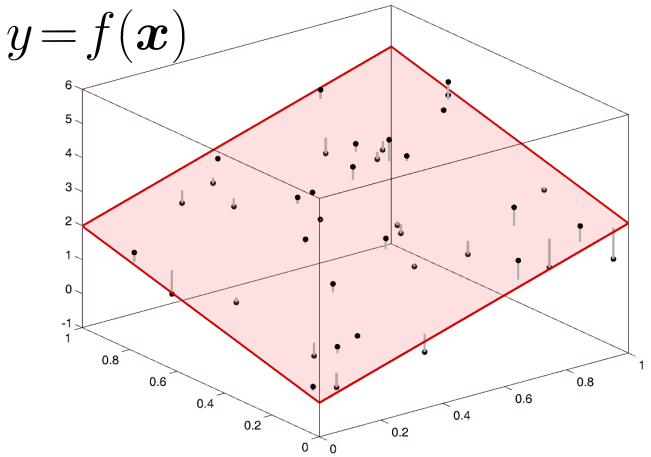
By looking at the similarities in the continuous **x** space, representing the locations at which we evaluate y = f(**x**)



Graphical model of a Gaussian process



$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$



x can be multivariate

Note that with GPs, we do not build a distribution on $\{x_1, x_2, \ldots x_N\}$!

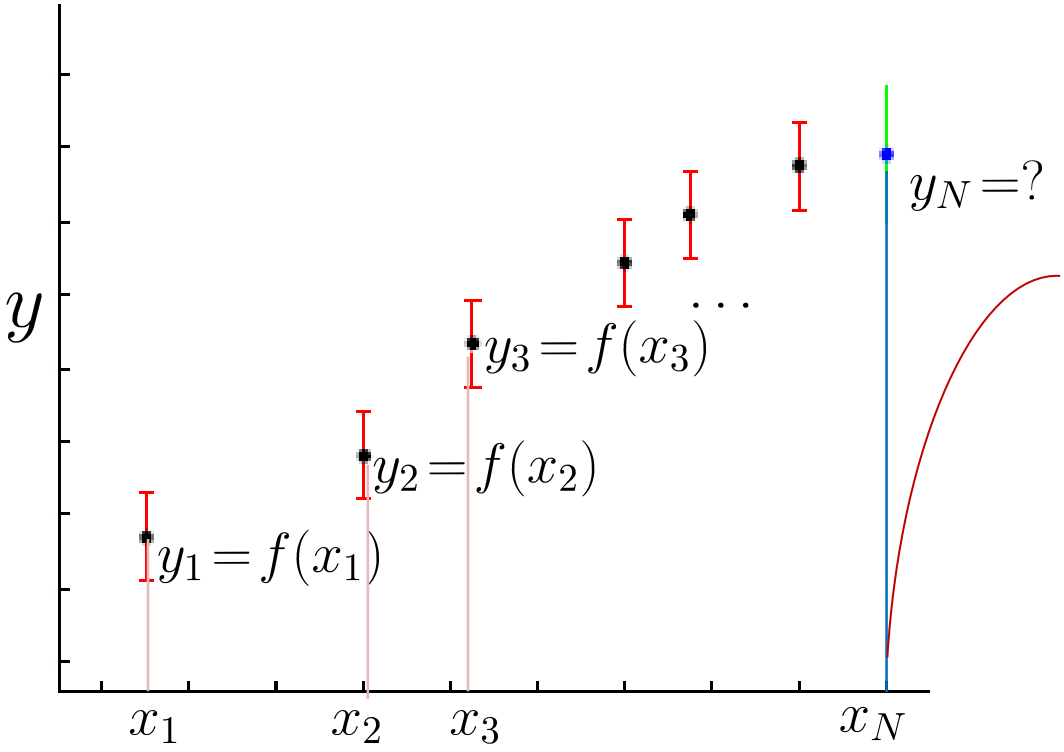
Gaussian process (GP)

- Gaussian processes (GPs) can be seen as an infinite-dimensional generalization of multivariate normal distributions.
- The **infinite joint distribution** over all possible variables is equivalent to a **distribution over a function space** y = f(**x**).
- **x** can for be a vector or any object, but y is a scalar output.
- To understand GPs, N observations of an arbitrary data set $y = \{y_1, \ldots, y_N\}$ should be imagined as a single point sampled from an N-variate Gaussian.

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

Gaussian process (GP)

- A covariance over an arbitrarily large set of variables can be defined through the **covariance kernel function** k(**x**$_i$, **x**$_j$), providing the covariance elements between any two sample locations **x**$_i$ and **x**$_j$.



If x_N is close to x_3, we also expect y_N to be close to y_3

Distribution over functions in GPs

For a set of spatial or temporal locations $\boldsymbol{x} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$, a positive semidefinite covariance matrix (also known as the Gram matrix) is defined as

$$\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & k(\boldsymbol{x}_1, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & k(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_2, \boldsymbol{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & k(\boldsymbol{x}_N, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix}.$$

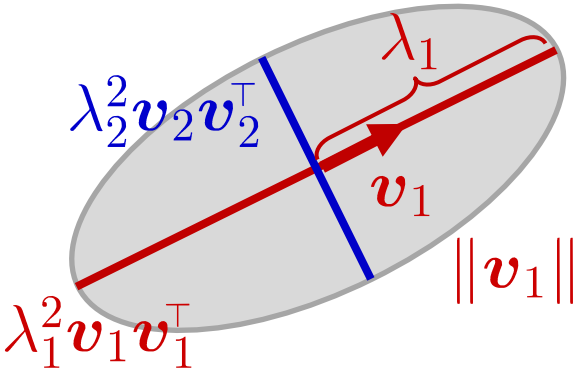
The entire function evaluation $y_n = f(\boldsymbol{x}_n)$ associated with the set of inputs $\boldsymbol{x}_n$ is a draw from a multivariate Gaussian distribution

$$\boldsymbol{y} \sim \mathcal{N}\Big(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})\Big),$$

specifying a **distribution over functions**.

Stochastic sampling with Gaussians

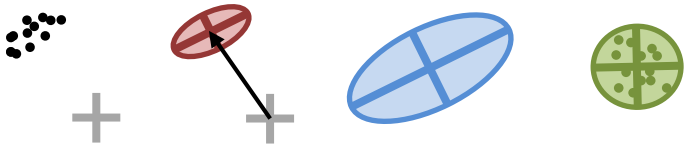
The eigendecomposition of $\boldsymbol{\Sigma}$ is expressed in a matrix form as



$$\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^\top = \sum_{j=1}^{D} \lambda_j^2 \boldsymbol{v}_j \boldsymbol{v}_j^\top$$

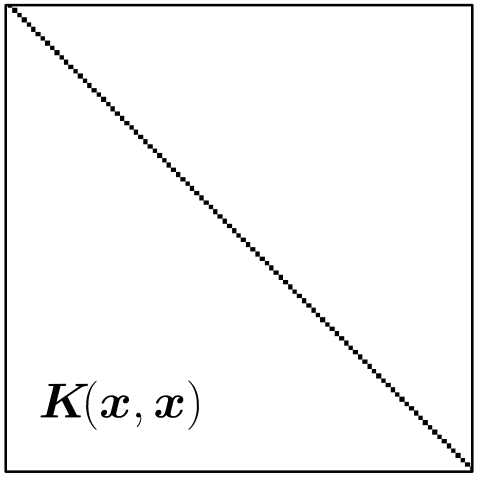
with $\boldsymbol{V} = \left[\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_D\right]$ and $\boldsymbol{D} = \begin{bmatrix} \lambda_1^2 & 0 & \cdots & 0 \\ 0 & \lambda_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_D^2 \end{bmatrix}$

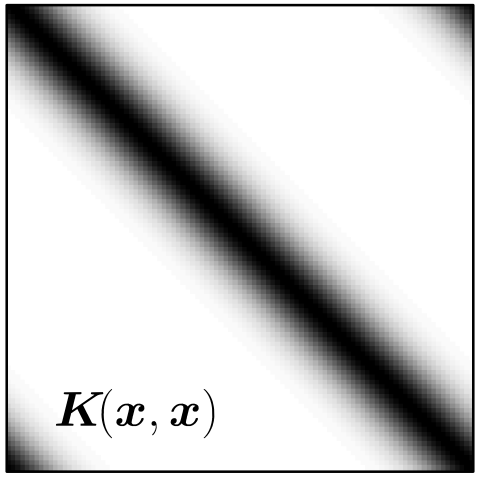
By using this notation, datapoints can be stochastically generated with



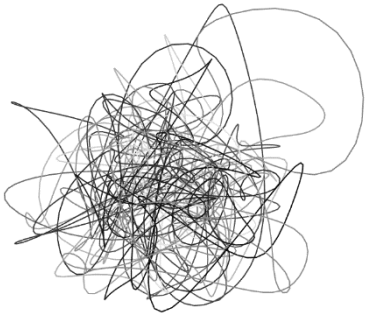
$$\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad \Longleftrightarrow \quad \boldsymbol{\xi} \sim \boldsymbol{\mu} + \boldsymbol{V}\boldsymbol{D}^{\frac{1}{2}}\,\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$$

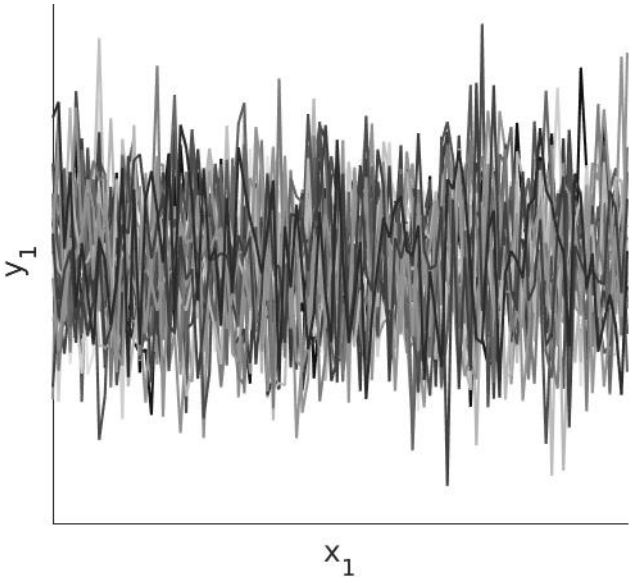
Distribution over functions in GPs - Sampling

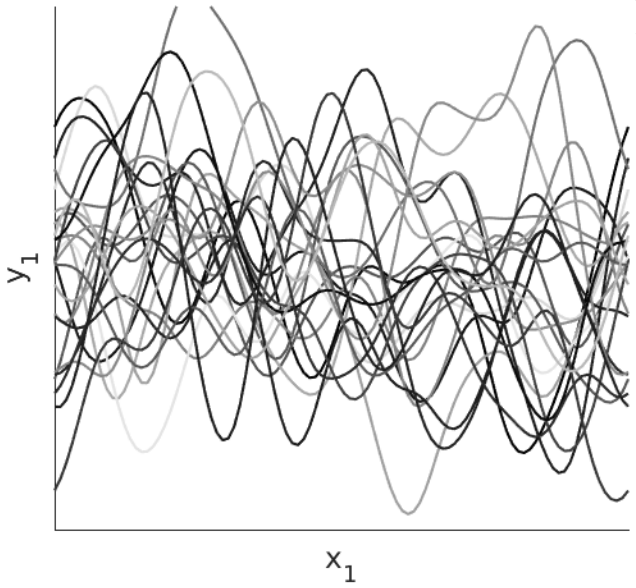




$$\boldsymbol{y} \sim \mathcal{N}\Big(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})\Big)$$







How to choose k(x_i,x_j)?

- We may know that our observations are samples from a process that is **smooth**, that is **continuous**, that has **typical amplitude**, or that the variations in the function take place within a **typical dynamic range.**
- These models require hyperparameters to be inferred, but **these hyperparameters define characteristics that are more generic** (such as the scale of a distribution) rather than acting explicitly on the structure or functional form of the signals.
- The notion of similarity will depend on the application: some of the basic aspects that can be defined through the covariance function k($\mathbf{x}_i$,$\mathbf{x}_j$) are the process **stationarity, isotropy, smoothness or periodicity**.
- With continuous time series, past observations will be informative about current data as a function of how long ago they were observed.
- This corresponds to a **stationary** covariance, dependent on the Euclidean distance $|\mathbf{x}_i - \mathbf{x}_j|$.
- This process is also considered as **isotropic** if it does not depend on directions between $\mathbf{x}_i$ and $\mathbf{x}_j$.
- A process that is both stationary and isotropic is **homogeneous**.

k(xi,xj) as squared exponential covariance

A popular homogeneous covariance function is the **squared exponential kernel**, also known as **radial basis function**

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right),$$

with two hyperparameters Θ_1^{GP} and Θ_2^{GP} corresponding respectively to **output and input scales** of the problem.

The radial basis function is widely employed when it is expected that nearby inputs $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ will have their corresponding outputs $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ also nearby (**assumption of continuity**).

k(xi,xj) as squared exponential covariance

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top \boldsymbol{M}(\boldsymbol{x}_i - \boldsymbol{x}_j)\right)$$

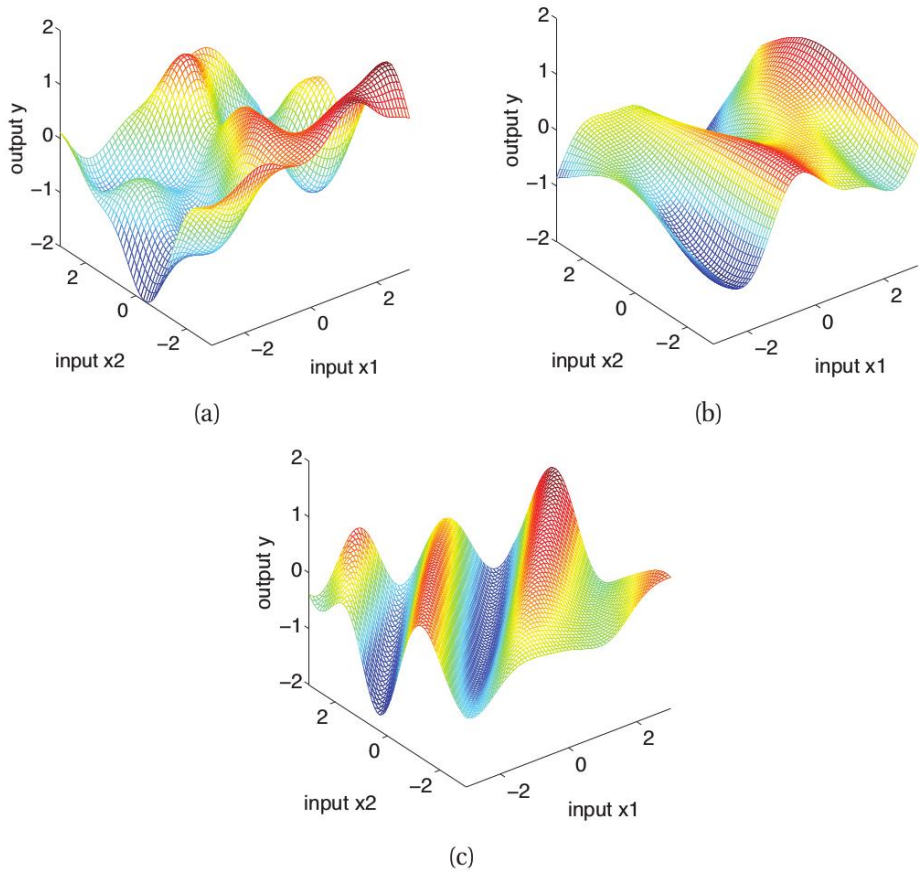


Figure from:

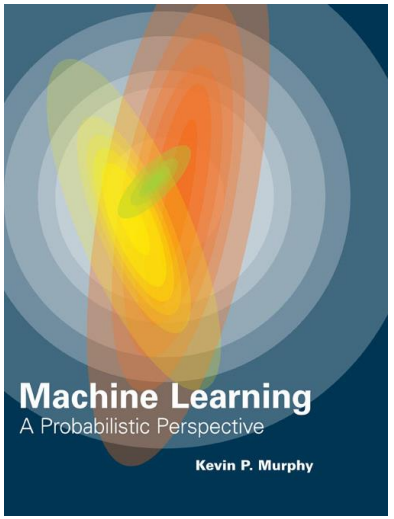


Figure 15.4 Some 2d functions sampled from a GP with an SE kernel but different hyper-parameters. The kernel has the form in Equation 15.20 where (a) $\mathbf{M} = \mathbf{I}$, (b) $\mathbf{M} = \mathrm{diag}(1, 3)^{-2}$, (c) $\mathbf{M} = (1, -1; -1, 1) + \mathrm{diag}(6, 6)^{-2}$. Based on Figure 5.1 of (Rasmussen and Williams 2006). Figure generated by `gprDemoArd`, written by Carl Rasmussen.

Modeling noise in the observed y_n

If we assume there is noise associated with the observed function values $\boldsymbol{y}_n = f(\boldsymbol{x}_n) + \boldsymbol{\eta}$, this noise term can also be modeled in the covariance.

This noise is most often assumed to be uncorrelated from sample to sample, meaning that the noise term is only added to the diagonal elements of $\boldsymbol{K}$, giving a modified covariance for noisy observations of the form

$$\tilde{\boldsymbol{K}}(\boldsymbol{x}, \boldsymbol{x}) = \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) + \Theta_3^{\mathrm{GP}} \boldsymbol{I}$$

where $\boldsymbol{I}$ is the identity matrix and Θ_3^{GP} is a Gaussian process hyperparameter representing the noise variance.

Modeling noise in the observed y_n

In the example of a covariance defined as squared exponential function, if noisy observations $\boldsymbol{y}$ are assumed, the kernel can be directly defined as

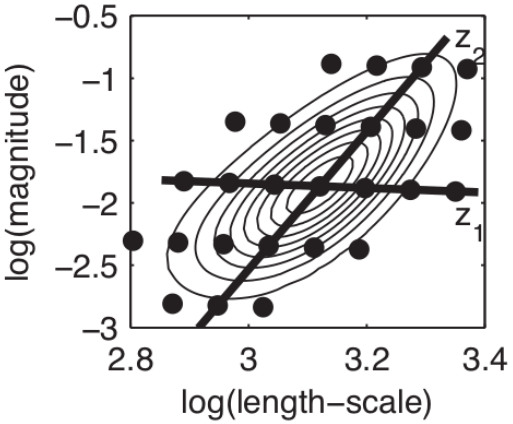
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}}\delta_{i,j},$$

where $\delta_{i,j} = \mathbb{I}(i = j)$ is equal to one only when $i = j$ and is zero otherwise, resulting in a covariance matrix $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})$ with noise related to observations only present in the diagonal (noise uncorrelated from sample to sample).

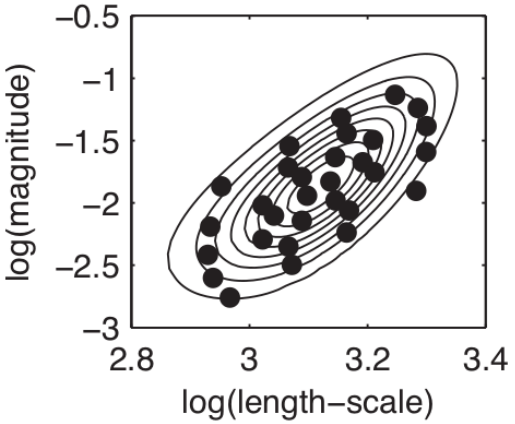
Learning the kernel function parameters

Several approaches exist to estimate the hyperparameters of the covariance function: Maximum Likelihood Estimation (MLE), cross-validation (CV), Bayesian approaches involving sampling algorithms such as MCMC, etc.

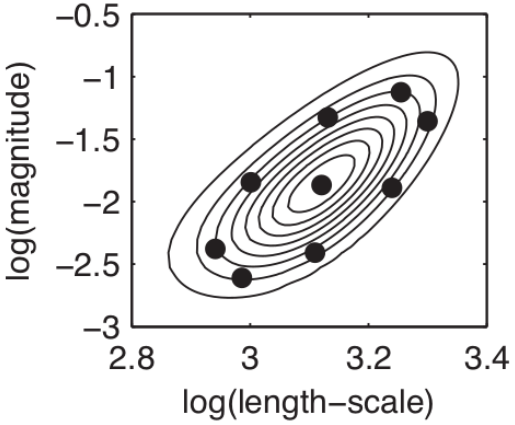
Grid-based



Monte-Carlo



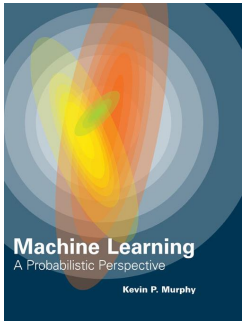
2nd order model



For example, given an expression for the log marginal likelihood and its derivative, we can estimate the kernel parameters using standard gradient-based optimizer.

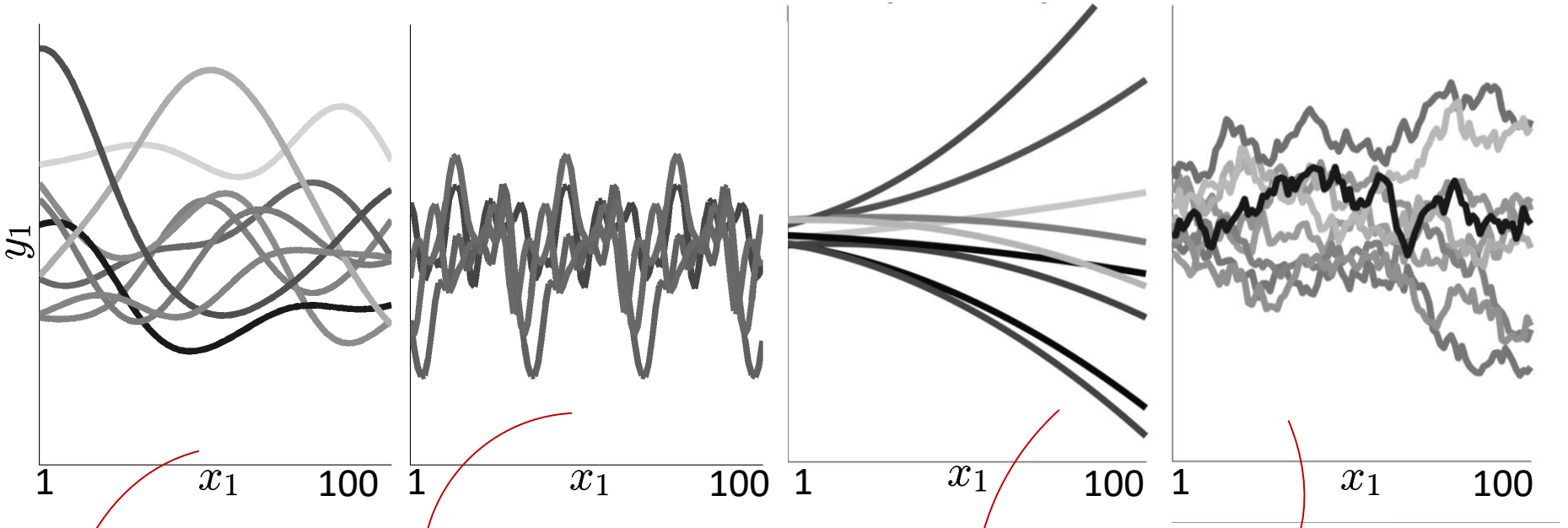
Since the objective is not convex, local minima can still be a problem.

Figure from:



Stochastic sampling from covariance matrix

$$\boldsymbol{y} \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})\big) \qquad \boldsymbol{x} = [1, 2, \ldots, 100]^\top$$



$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}} \sin^2\left(\Theta_4^{\text{GP}} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|\right)\right)$$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \min(\boldsymbol{x}_i, \boldsymbol{x}_j) + \Theta_1^{\text{GP}}$$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}} (\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right)$$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j + \Theta_1^{\text{GP}})^2$$

Gaussian process regression (GPR)

a.k.a.

Kriging

Python notebook:
demo_GPR.ipynb

Matlab code:
demo_GPR01.m

Gaussian process regression (GPR)

We are interested in the **posterior distribution** of $\boldsymbol{y}^*$ to be computed at some location(s) $\boldsymbol{x}^*$.

The **joint distribution** of the already observed $\boldsymbol{y}$ (at location $\boldsymbol{x}$) augmented by $\boldsymbol{y}^*$ (at location $\boldsymbol{x}^*$) is

$$\boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}) = \begin{bmatrix} k(\boldsymbol{x}_1,\boldsymbol{x}_1) & k(\boldsymbol{x}_1,\boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_1,\boldsymbol{x}_N) \\ k(\boldsymbol{x}_2,\boldsymbol{x}_1) & k(\boldsymbol{x}_2,\boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_2,\boldsymbol{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N,\boldsymbol{x}_1) & k(\boldsymbol{x}_N,\boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_N,\boldsymbol{x}_N) \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{y}^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}(\boldsymbol{x}) \\ \boldsymbol{\mu}(\boldsymbol{x}^*) \end{bmatrix}, \begin{bmatrix} \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}) & \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}^*) \\ \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x}) & \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x}^*) \end{bmatrix} \right)$$

We can use the conditional probability property of Gaussians to evaluate the posterior distribution over $\boldsymbol{y}^*$, yielding a Gaussian

$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}\left(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right)$$

$$\text{with} \quad \boldsymbol{\mu}^* = \boldsymbol{\mu}(\boldsymbol{x}^*) + \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x})\, \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1} \left(\boldsymbol{y} - \boldsymbol{\mu}(\boldsymbol{x})\right)$$

$$\boldsymbol{\Sigma}^* = \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x}^*) - \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x})\, \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1} \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}^*)$$

GPR in practice

$$\boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}) = \begin{bmatrix} k(\boldsymbol{x}_1,\boldsymbol{x}_1) & k(\boldsymbol{x}_1,\boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_1,\boldsymbol{x}_N) \\ k(\boldsymbol{x}_2,\boldsymbol{x}_1) & k(\boldsymbol{x}_2,\boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_2,\boldsymbol{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N,\boldsymbol{x}_1) & k(\boldsymbol{x}_N,\boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_N,\boldsymbol{x}_N) \end{bmatrix}$$

It is also often assumed in practice that $\begin{bmatrix} \boldsymbol{\mu}(\boldsymbol{x}) \\ \boldsymbol{\mu}(\boldsymbol{x}^*) \end{bmatrix} = \mathbf{0}$.

In this case, Gaussian processes can be completely defined by second-order statistics, where the Gram matrix $\boldsymbol{K}$ is a positive semi-definite covariance.

Note that $\boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1}$ can be pre-computed so that the posterior distribution can be compted faster

$$\boldsymbol{y}^*|\boldsymbol{y} \sim \mathcal{N}\left(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right)$$

$$\text{with} \quad \boldsymbol{\mu}^* = \boldsymbol{\mu}(\boldsymbol{x}^*) + \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x})\, \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1} \left(\boldsymbol{y} - \boldsymbol{\mu}(\boldsymbol{x})\right)$$

$$\boldsymbol{\Sigma}^* = \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x}^*) - \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x})\, \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1} \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}^*)$$

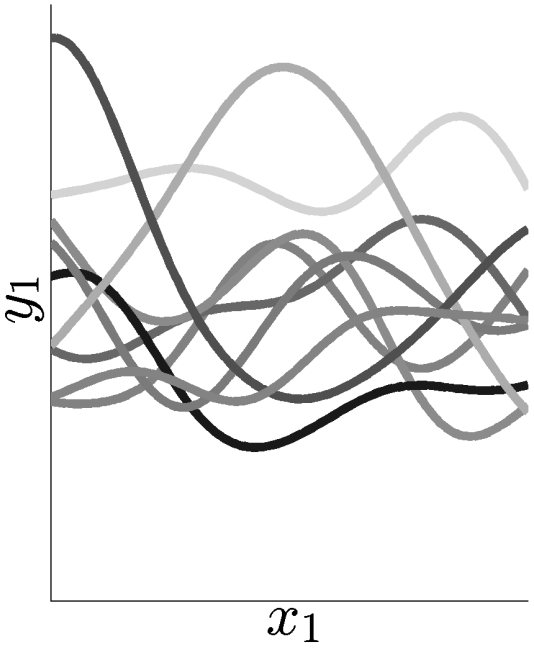
Examples of covariance functions

k(xi,xj) as squared exponential covariance

$$\Theta_1^{\mathrm{GP}} = 1, \quad \Theta_2^{\mathrm{GP}} = 0.1, \quad \Theta_3^{\mathrm{GP}} = 0$$

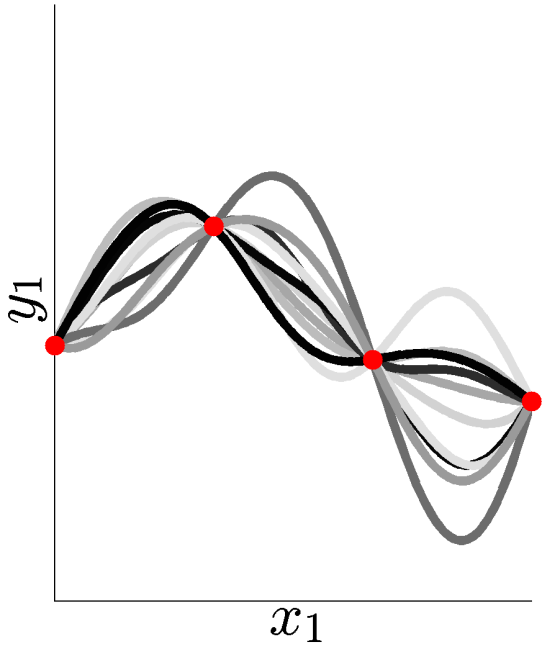
$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$

Samples from prior



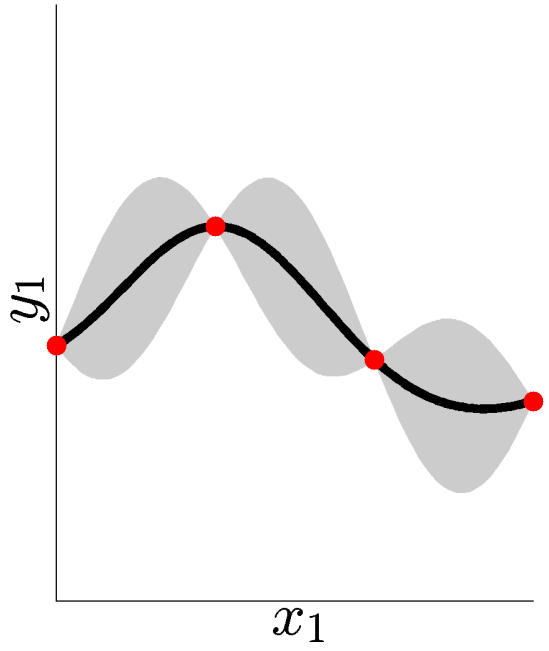
$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Samples from posterior



$\mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Trajectory distribution



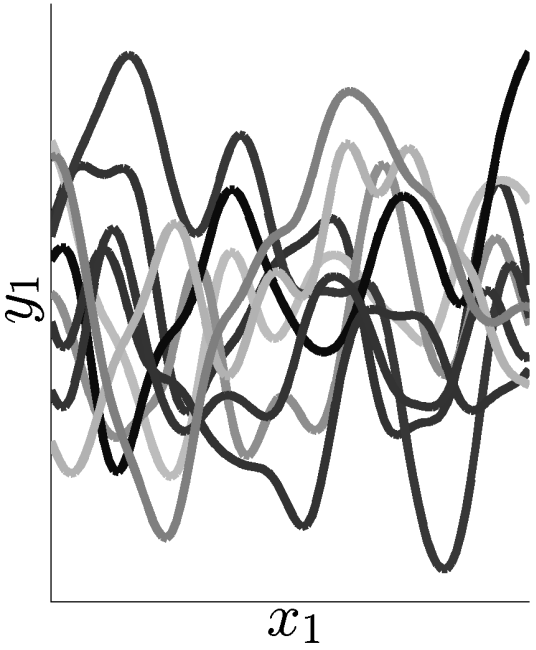
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}}\delta_{i,j}$$

k(xi,xj) as squared exponential covariance

$$\Theta_1^{\mathrm{GP}} = 1, \quad \Theta_2^{\mathrm{GP}} = 0.01, \quad \Theta_3^{\mathrm{GP}} = 0$$

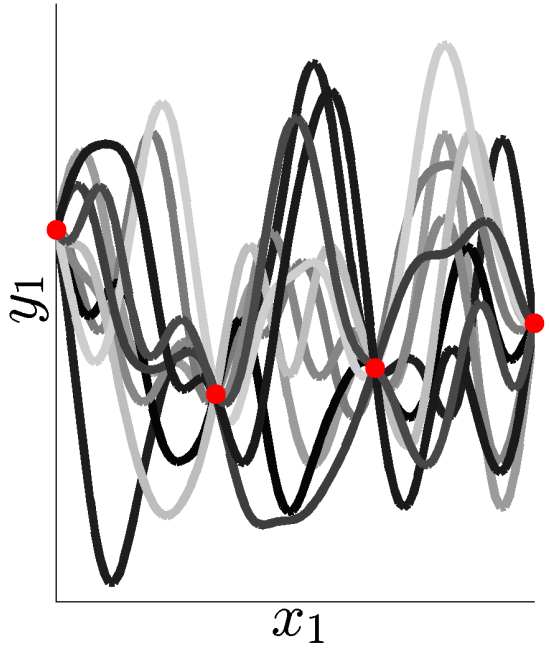
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

Samples from prior



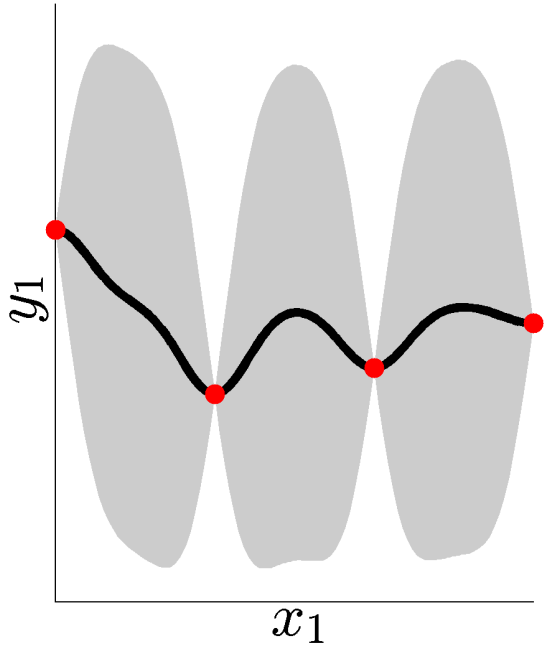
$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Samples from posterior



$$\mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Trajectory distribution



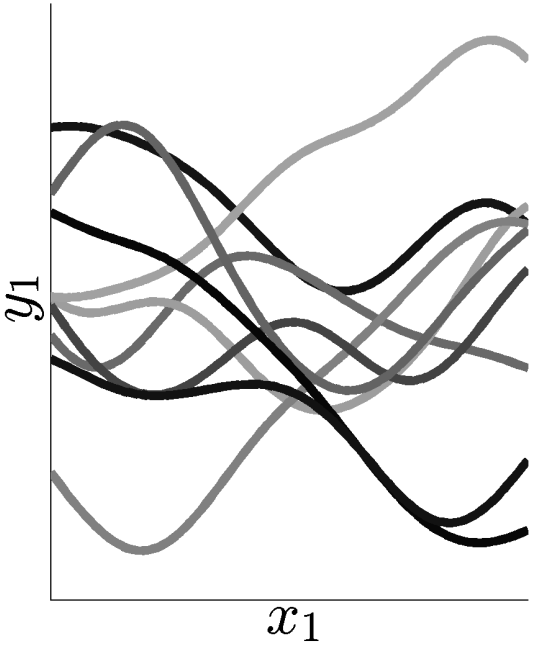
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}} \delta_{i,j}$$

k(xi,xj) as squared exponential covariance

$$\Theta_1^{\mathrm{GP}} = 1, \quad \Theta_2^{\mathrm{GP}} = 0.1, \quad \Theta_3^{\mathrm{GP}} = 0.01$$

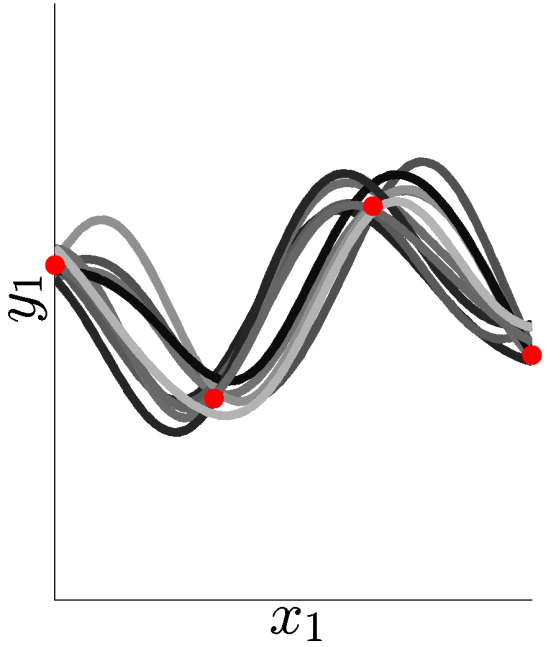
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

Samples from prior



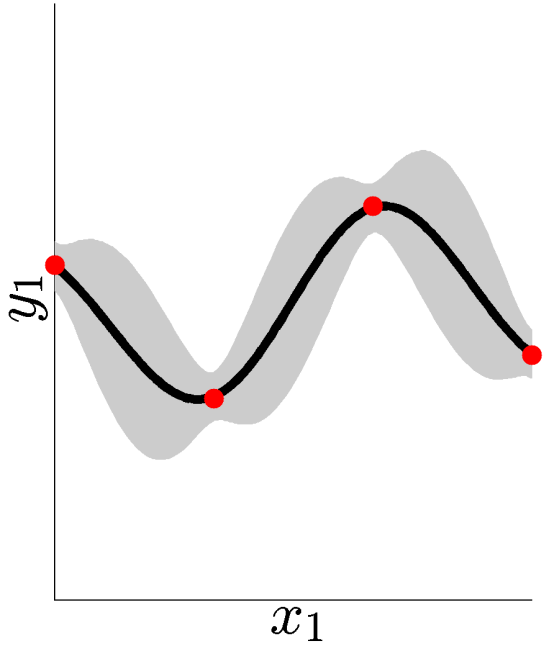
$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Samples from posterior



$$\mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Trajectory distribution



$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}}\delta_{i,j}$$

k(xi,xj) as squared exponential covariance

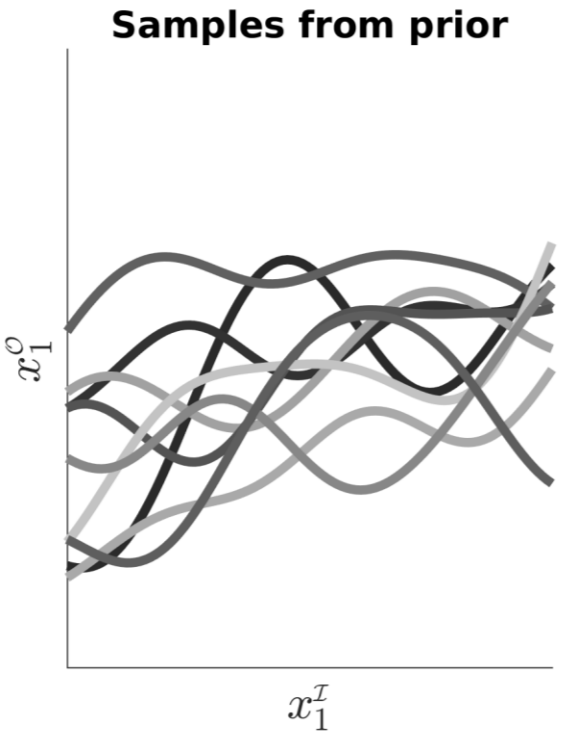
$$\Theta_1^{\text{GP}} = 1, \quad \Theta_2^{\text{GP}} = 0.1, \quad \Theta_3^{\text{GP}} = 0.01$$

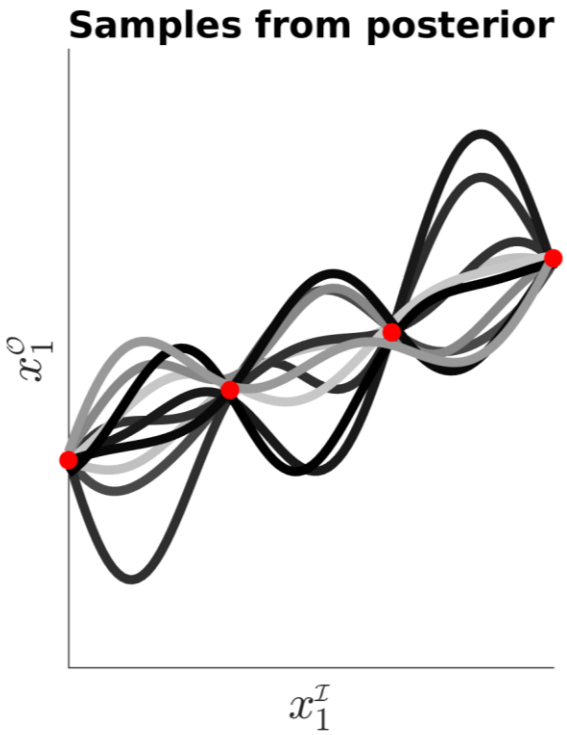
$$\boldsymbol{\mu}(\boldsymbol{x}) = \alpha\, \boldsymbol{x}$$

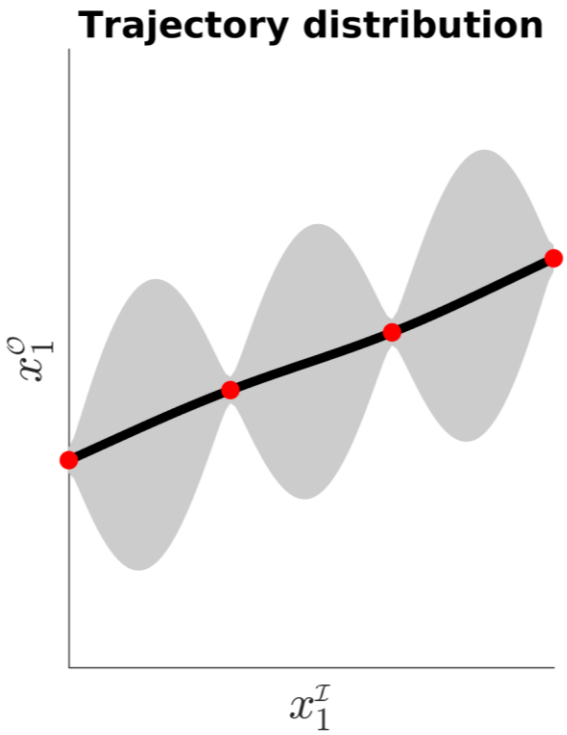
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

$$\mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

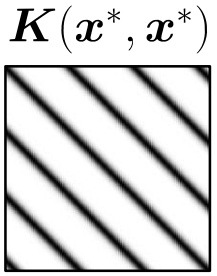






$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\text{GP}} \delta_{i,j}$$

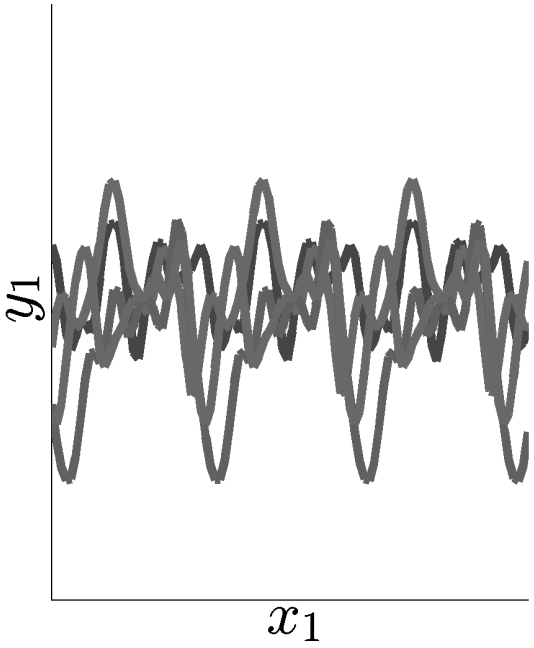
k(xi,xj) as periodic covariance function



$$\Theta_1^{\text{GP}} = 0.1, \quad \Theta_2^{\text{GP}} = 0.1, \quad \Theta_3^{\text{GP}} = 0, \quad \Theta_4^{\text{GP}} = 10$$

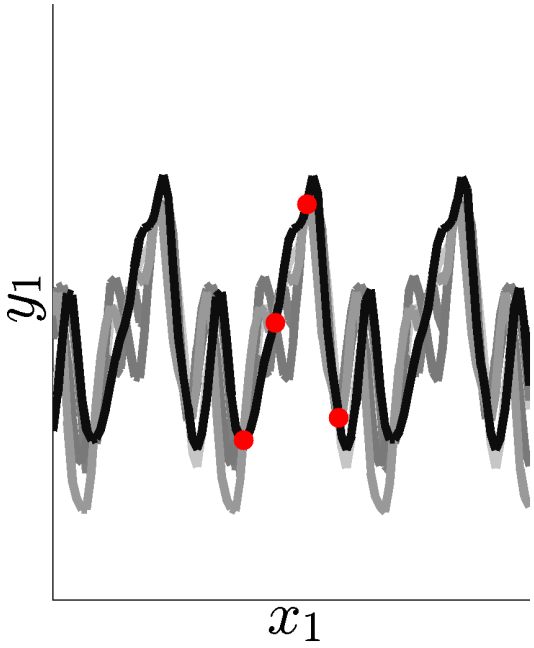
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

Samples from prior



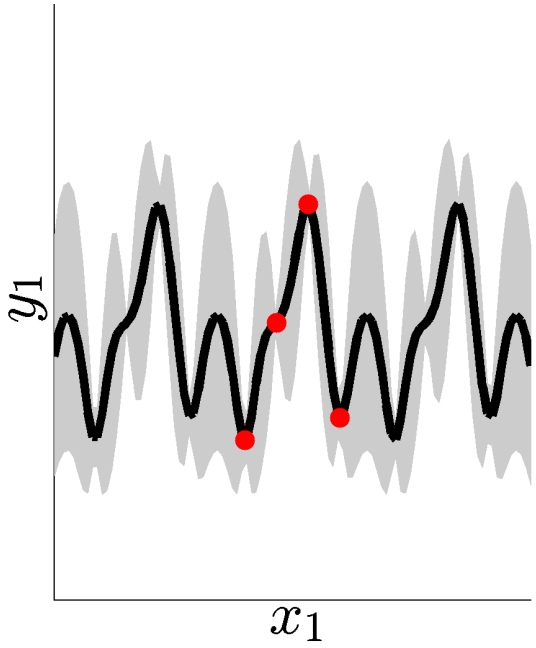
$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Samples from posterior



$$\mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Trajectory distribution



$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}} \sin^2\left(\Theta_4^{\text{GP}} |\boldsymbol{x}_i - \boldsymbol{x}_j|\right)\right) + \Theta_3^{\text{GP}} \delta_{i,j}$$

k(xi,xj) as Matern covariance function

Another popular covariance kernel function is the Matérn function

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{d}{\rho} \right)^{\nu} K_\nu \left(\sqrt{2\nu} \frac{d}{\rho} \right)$$

$$\text{with } d = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$$

where Γ is the gamma function, K_ν is the modified Bessel function of the second kind, and ρ and ν are non-negative parameters of the covariance.

A Gaussian process with Matérn covariance has sample paths that are $\lfloor \nu - 1 \rfloor$ times differentiable.

k(xi,xj) as Matern covariance function

Simplification for ν half integer

When $\nu = p + 1/2,\ p \in \mathbb{N}^+$, the Matérn covariance can be written as a product of an exponential and a polynomial of order p:

$$C_{p+1/2}(d) = \sigma^2 \exp\left(-\frac{\sqrt{2\nu}d}{\rho}\right) \frac{\Gamma(p+1)}{\Gamma(2p+1)} \sum_{i=0}^{p} \frac{(p+i)!}{i!(p-i)!} \left(\frac{\sqrt{8\nu}d}{\rho}\right)^{p-i}$$

For $\nu = 1/2$ $(p = 0)$: $C_{1/2}(d) = \sigma^2 \exp\left(-\frac{d}{\rho}\right)$

For $\nu = 3/2$ $(p = 1)$: $C_{3/2}(d) = \sigma^2 \left(1 + \frac{\sqrt{3}d}{\rho}\right) \exp\left(-\frac{\sqrt{3}d}{\rho}\right)$

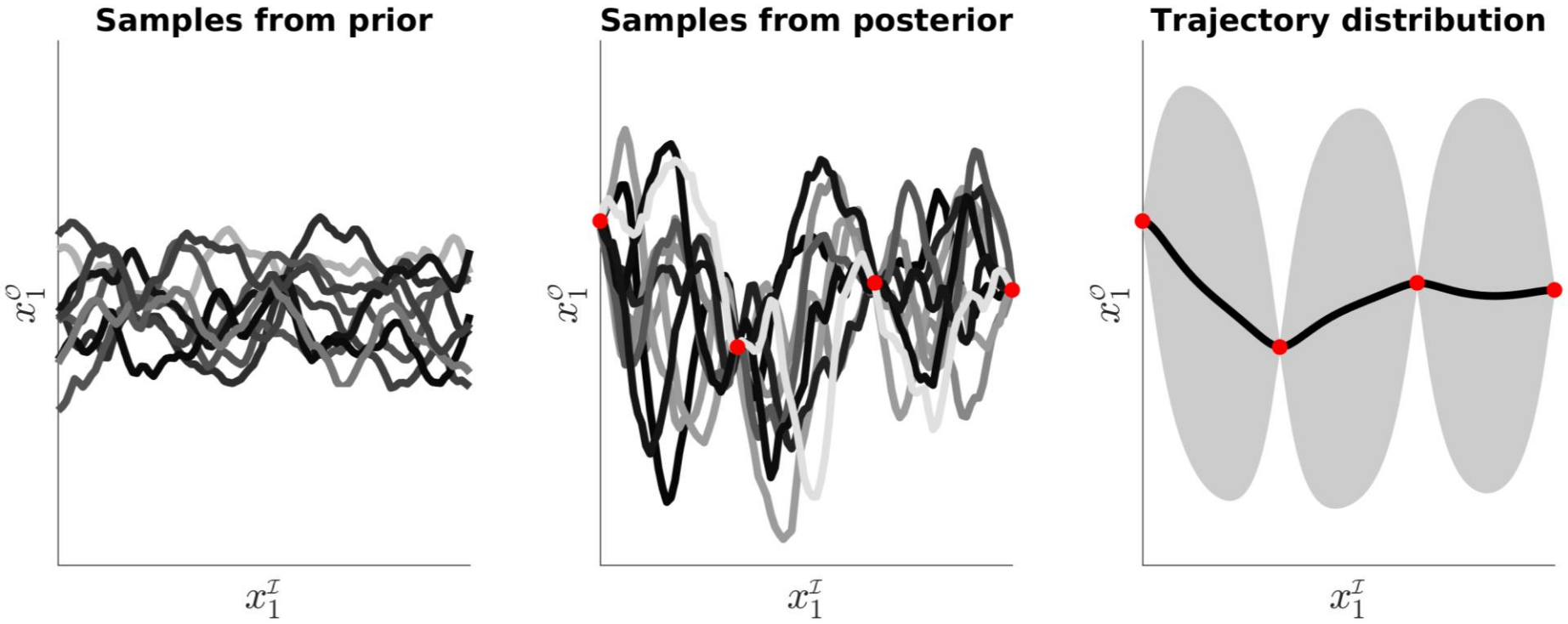
For $\nu = 5/2$ $(p = 2)$: $C_{5/2}(d) = \sigma^2 \left(1 + \frac{\sqrt{5}d}{\rho} + \frac{5d^2}{3\rho^2}\right) \exp\left(-\frac{\sqrt{5}d}{\rho}\right)$

As $\nu \to \infty$, the Matérn covariance converges to the squared exponential covariance function.

k(xi,xj) as Matern covariance function

$$\Theta_1^{\mathrm{GP}} = 0.1, \quad \Theta_2^{\mathrm{GP}} = 0.1, \quad \Theta_3^{\mathrm{GP}} = 0.0001$$

$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big) \qquad \boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*) \qquad \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

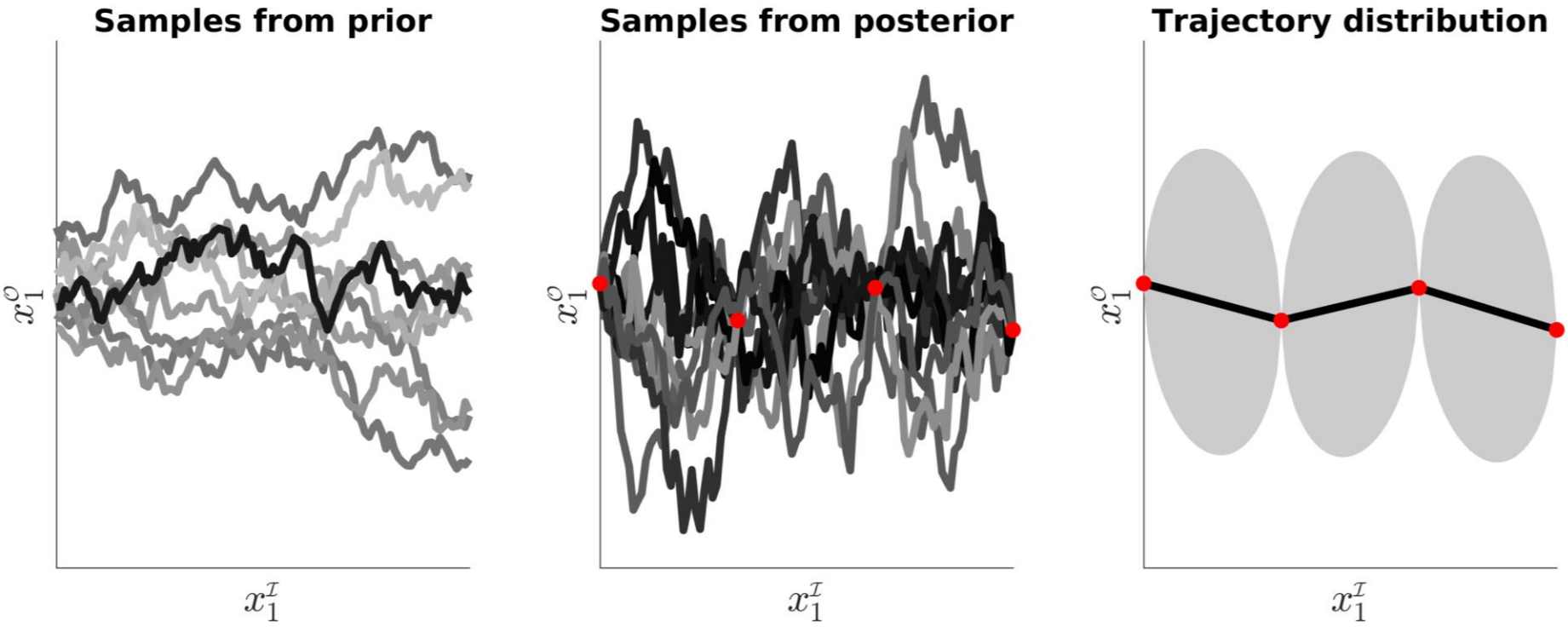


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \left(1 + \frac{\sqrt{3}\,\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{\Theta_2^{\mathrm{GP}}}\right) \exp\left(-\frac{\sqrt{3}\,\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{\Theta_2^{\mathrm{GP}}}\right)$$

k(xi,xj) as Brownian motion covariance function

The **Wiener process** is a simple continuous-time stochastic process often put in connection to the Brownian motion.

$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big) \qquad \boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*) \qquad \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

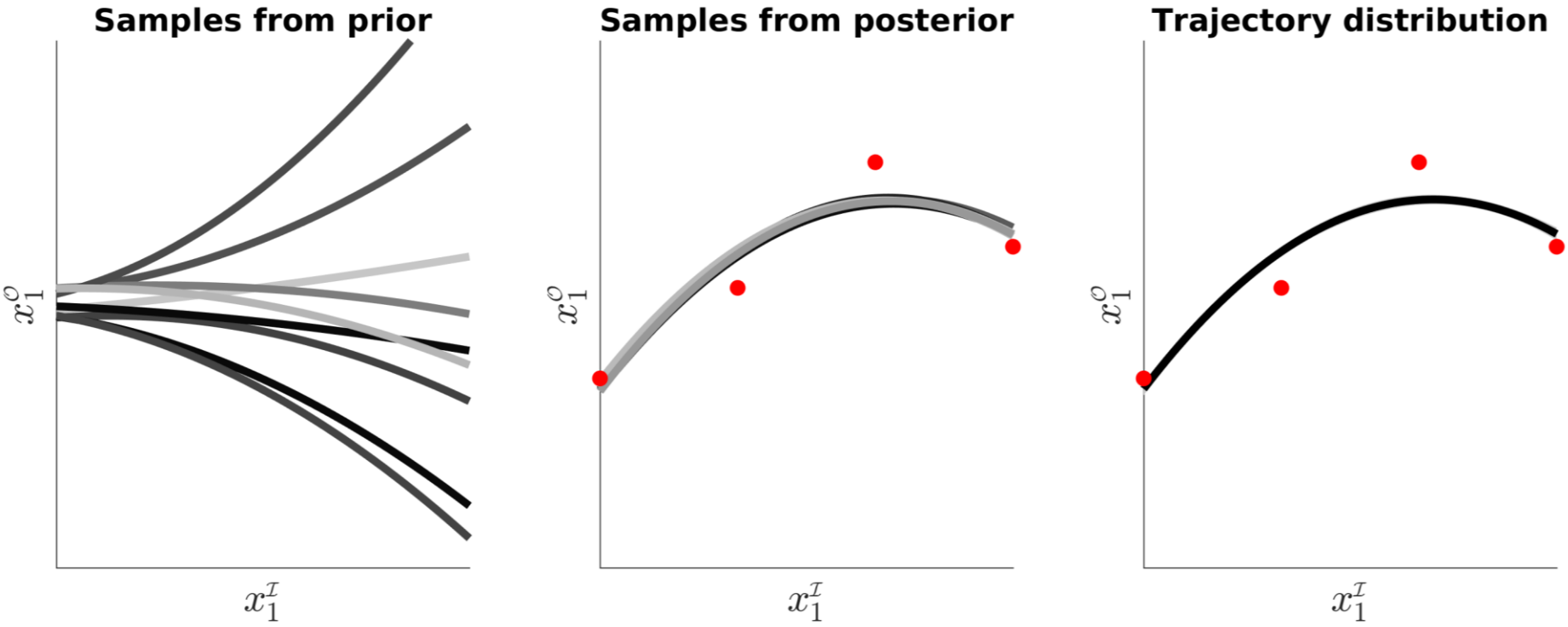


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \min(\boldsymbol{x}_i, \boldsymbol{x}_j) + \Theta_1^{\mathrm{GP}} \qquad \Theta_1^{\mathrm{GP}} = 0.1$$

k(xi,xj) as quadratic covariance function

Bayesian linear regression is equivalent to a GP with covariance function $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^\top \boldsymbol{x}_j$.

$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big) \qquad \boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*) \qquad \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$



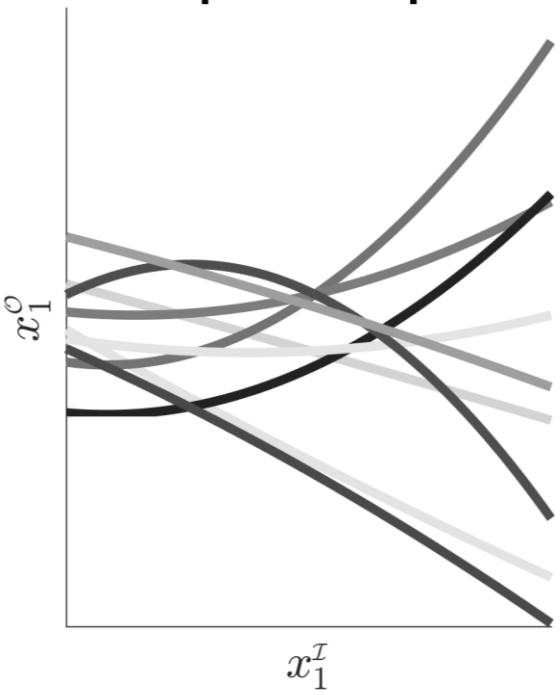
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j + \Theta_1^{\mathrm{GP}})^2 \qquad \Theta_1^{\mathrm{GP}} = 0.1$$

k(xi,xj) as polynomial covariance function

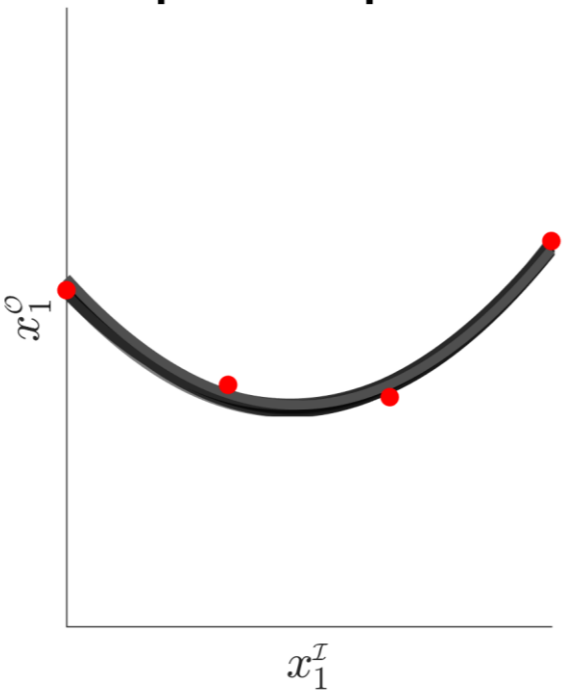
$$\Theta_1^{\mathrm{GP}} = 0.1$$

$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big) \qquad \boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*) \qquad \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

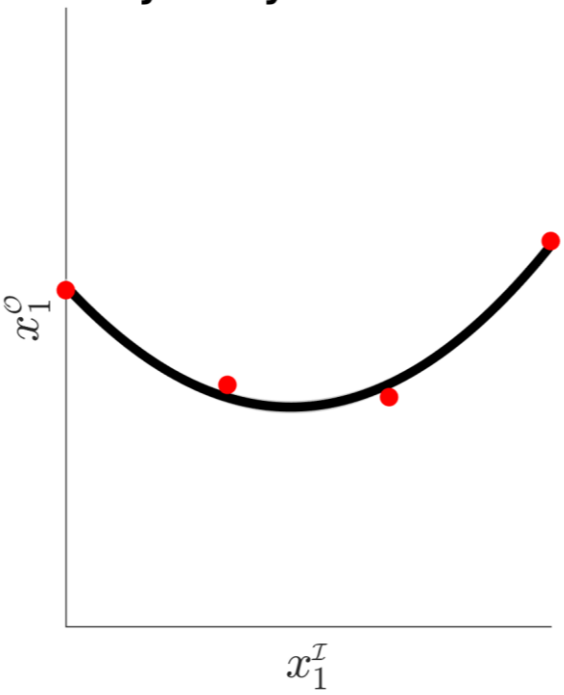
Samples from prior



Samples from posterior



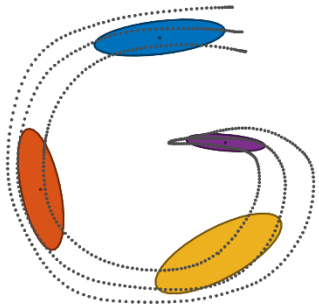
Trajectory distribution

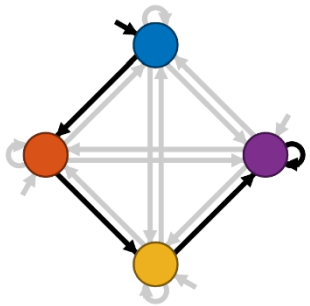


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j)^2 + \boldsymbol{x}_i^\top \boldsymbol{x}_j + \Theta_1^{\mathrm{GP}}$$

$k(x_i,x_j)$ as probabilistic model covariance

- Another powerful approach to the construction of kernels is to exploit probabilistic models.
- Given a generative model $P(\mathbf{x})$, a valid kernel can be defined as $k(\mathbf{x}_i, \mathbf{x}_j) = P(\mathbf{x}_i)\, P(\mathbf{x}_j)$, which can be interpreted as an inner product in the one-dimensional feature space defined by the mapping $P(\mathbf{x})$.
- Namely, two inputs $\mathbf{x}_i$ and $\mathbf{x}_j$ will be similar if they both have high probabilities to belong to the model.
- This can bring additional properties to the underlying process such as the capability of handling missing data or partial sequences of various lengths (e.g., with HMM).





k(xi,xj) as weighted sum of kernel functions

- A covariance function can be defined as a **linear combination of other covariance functions**, which can be exploited to incorporate **different insights about the dataset**.
- Such an approach can be exploited as an alternative to optimizing kernel parameters (also known as multiple kernel learning).
- The idea is to define the kernel as a **weighted sum of basis kernels**, and then to **optimize the weights instead of the kernel parameters**.

Dictionary of basis kernel functions

$$\begin{cases} k_1(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j)^2 + \boldsymbol{x}_i^\top \boldsymbol{x}_j \\ k_2(\boldsymbol{x}_i, \boldsymbol{x}_j) = \min(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ k_3(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) \end{cases}$$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} k_1(\boldsymbol{x}_i, \boldsymbol{x}_j) + \Theta_2^{\mathrm{GP}} k_2(\boldsymbol{x}_i, \boldsymbol{x}_j) + \Theta_3^{\mathrm{GP}} k_3(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

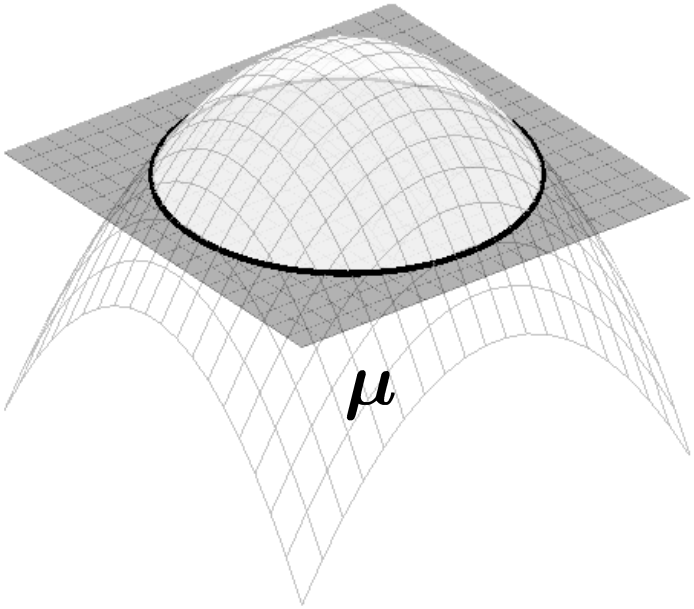
Some extensions of Gaussian processes

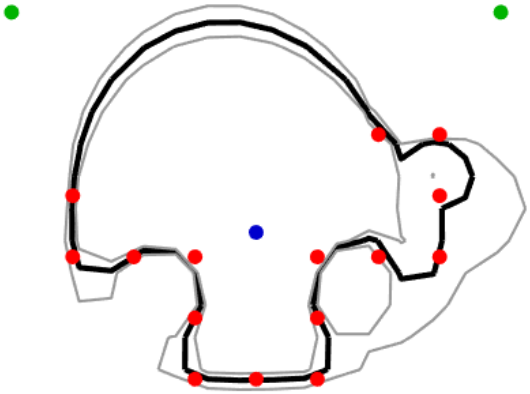
- **Cokriging:**
  Extending GPR to multiple target variables y.

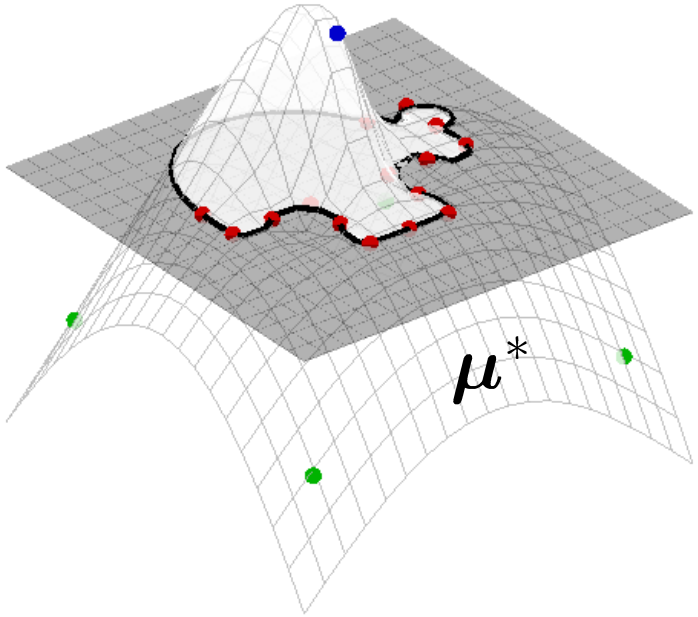
- **Sparse GP:**
  A known bottleneck in Gaussian process prediction is that the computational complexity of prediction is $O(N^3)$
  → not feasible for large data sets!
  Sparse Gaussian processes circumvent this issue by building a representative set for the given process $y = f(\mathbf{x})$.

- **Gaussian process latent variable models (GPLVM):**
  GPLVM is a probabilistic dimensionality reduction method that uses GPs to find a lower dimensional non-linear embedding of high dimensional data.

Gaussian Process Implicit Surface (GPIS)







+1

0

-1

References

GPR

C.K.I. Williams and C.E. Rasmussen. Gaussian processes for regression.
In Advances in Neural Information Processing Systems (NIPS), pages 514–520, 1996

C.E. Rasmussen and C.K.I. Williams. Gaussian processes for machine learning. MIT Press, Cambridge, MA, USA, 2006

S. Roberts, M. Osborne, M. Ebden, S. Reece, N. Gibson, and S. Aigrain. Gaussian processes for time-series modelling. Philosophical Trans. of the Royal Society A, 371(1984):1–25, 2012

GPLVM

N. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. Journal of machine learning research, 6:1783-1816, 2005

GPIS

O. Williams and A. Fitzgibbon. Gaussian Process Implicit Surfaces. In Gaussian Processes in Practice, 2007