EE613
Machine Learning for Engineers

NONLINEAR REGRESSION

Sylvain Calinon
Robot Learning & Interaction Group
Idiap Research Institute
Nov. 11, 2015

Outline

- Locally weighted regression (LWR)
- Radial basis functions (RBF)
- Gaussian mixture regression (GMR)
- Gaussian process regression (GPR)

Locally weighted regression (LWR)

demo_LWR01.m

[C. G. Atkeson, A. W. Moore, and S. Schaal. Locally weighted learning for control. Artificial Intelligence Review, 11(1-5):75–113, 1997]

Locally weighted regression (LWR)

Locally weighted regression (LWR) is an extension of weighted least squares, in which K independent weighted regressions are performed on the same dataset $\{\boldsymbol{X}, \boldsymbol{Y}\}$.

LWR thus computes K independent estimates, each with a different weighting function $\phi_k(\boldsymbol{x}_t)$ normalized such that $\sum_{k=1}^{K} \phi_k(\boldsymbol{x}_t) = 1$.

$\phi_k(\boldsymbol{x}_t)$ are usually defined as **radial basis functions (RBF)**

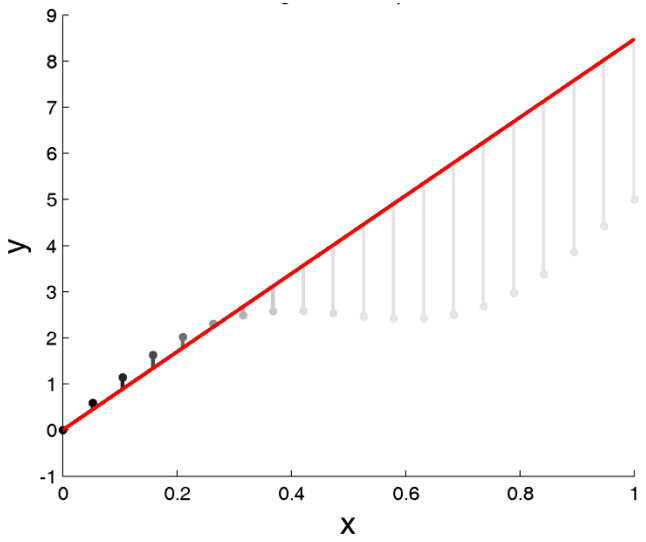
$$\phi_k(\boldsymbol{x}_t) = \exp\left(-(\boldsymbol{x}_t - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_t - \boldsymbol{\mu}_k)\right)$$

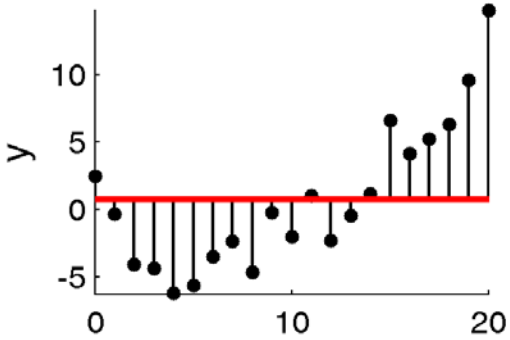
where $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are the parameters of the k-th RBF.

Most often, the centroids $\boldsymbol{\mu}_k$ are set to uniformly cover in the input space, and $\boldsymbol{\Sigma}_k = \boldsymbol{I}\sigma^2$ is considered (common bandwidth shared by all basis functions).

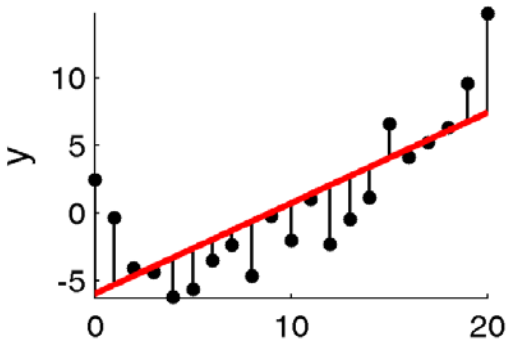
Locally weighted regression (LWR)

$$\hat{A} = (X^\top W X)^{-1} X^\top W Y$$

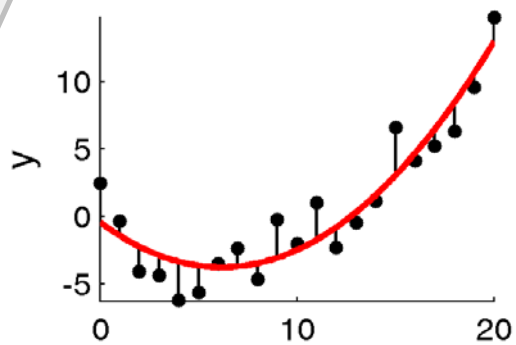




$$X = 1$$



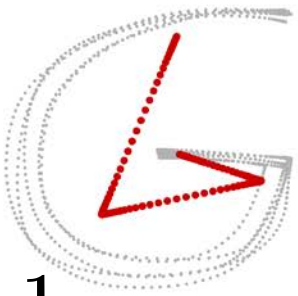
$$X = [x, 1]$$



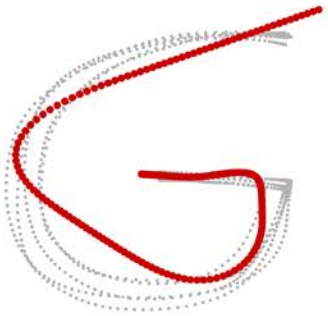
$$X = [x^2, x, 1]$$

LWR can be directly extended to local least squares polynomial fitting by changing the definition of the inputs.

Locally weighted regression (LWR)

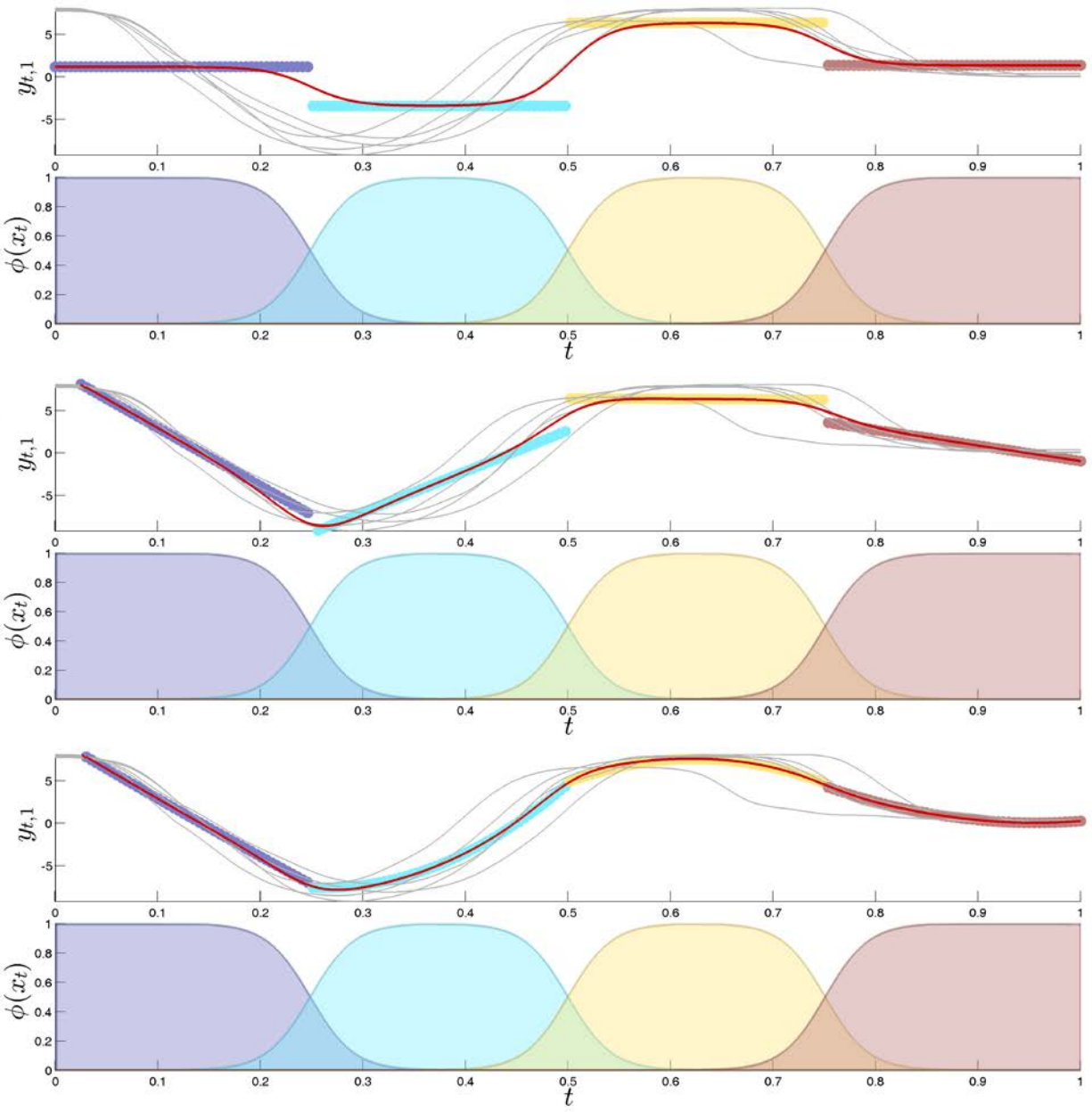


$\boldsymbol{X} = \mathbf{1}$

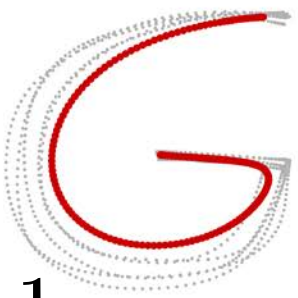


$\boldsymbol{X} = [\boldsymbol{x}, \mathbf{1}]$

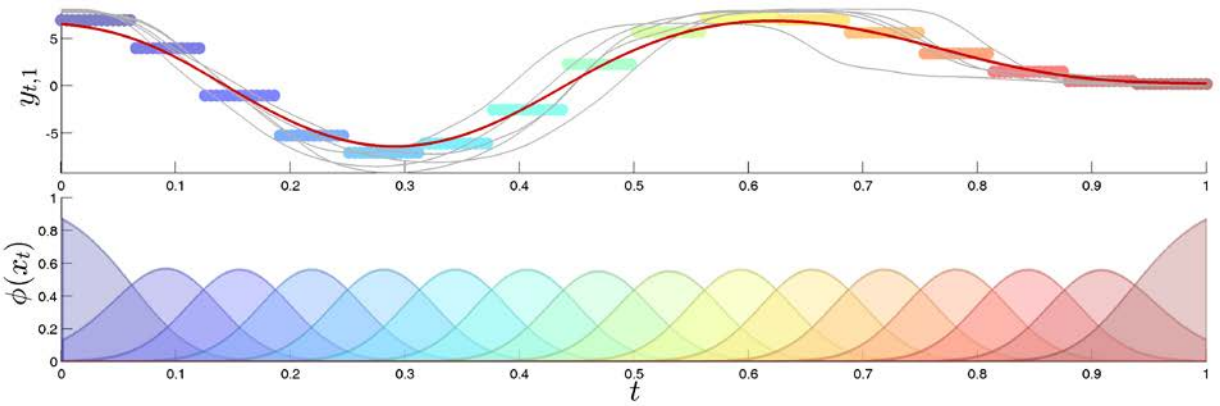
$\boldsymbol{X} = [\boldsymbol{x}^2, \boldsymbol{x}, \mathbf{1}]$



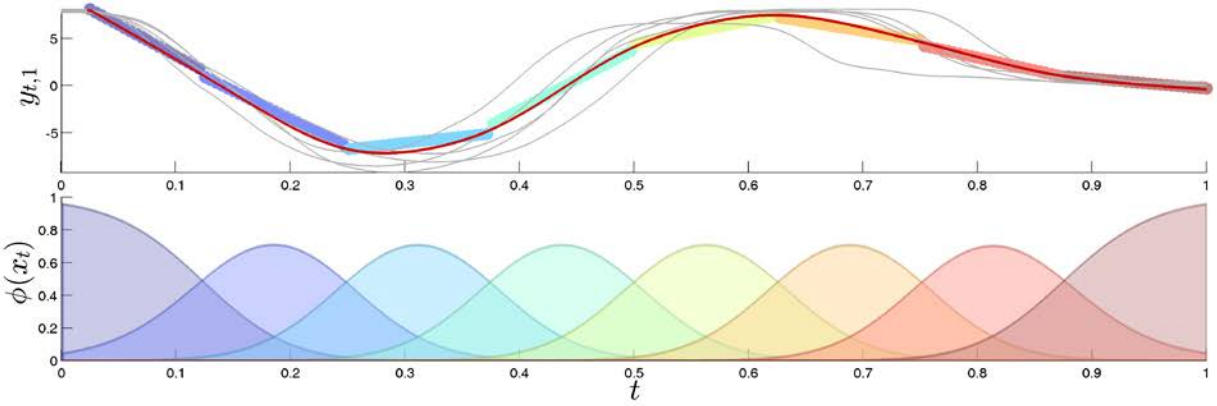
Locally weighted regression (LWR)



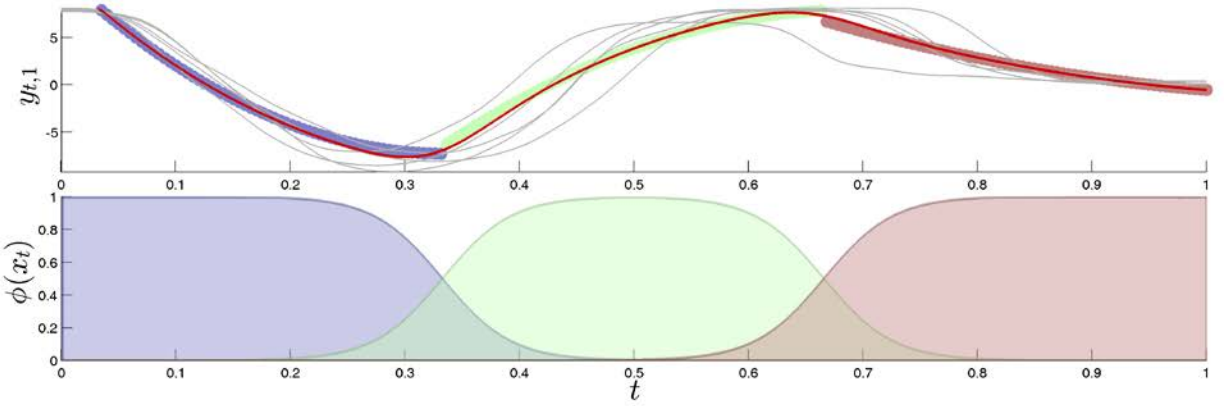
$$X = 1$$



$$X = [x, 1]$$



$$X = [x^2, x, 1]$$

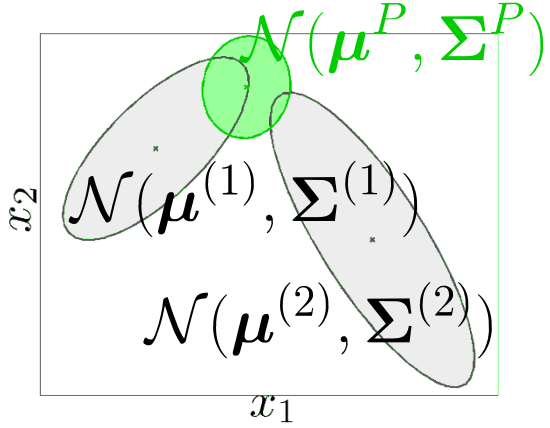


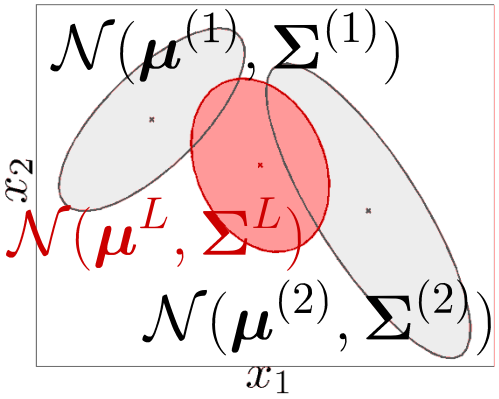
Gaussian mixture regression (GMR)

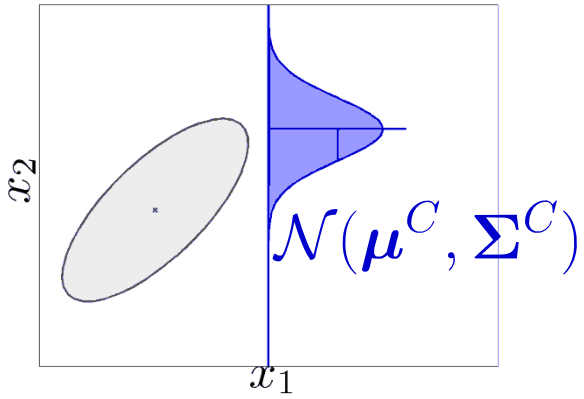
demo_GMR01.m
demo_GMR_polyFit01.m

[Z. Ghahramani and M. I. Jordan. Supervised learning from incomplete data via an EM approach. In Advances in Neural Information Processing Systems (NIPS), volume 6, pages 120–127, 1994]

Gaussian distribution properties







Product of Gaussians:

$$\mathcal{N}(\boldsymbol{\mu}^P, \boldsymbol{\Sigma}^P) \sim \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}) \cdot \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$$

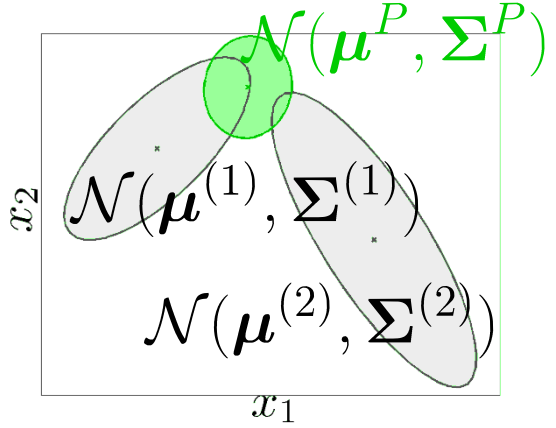
Linear combination:

$$\mathcal{N}(\boldsymbol{\mu}^L, \boldsymbol{\Sigma}^L) \sim \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}) + \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$$

Conditional probability:

$$\mathcal{N}(\boldsymbol{\mu}^C, \boldsymbol{\Sigma}^C) \sim \mathcal{P}(\boldsymbol{x}_2 | \boldsymbol{x}_1)$$

Product of Gaussians

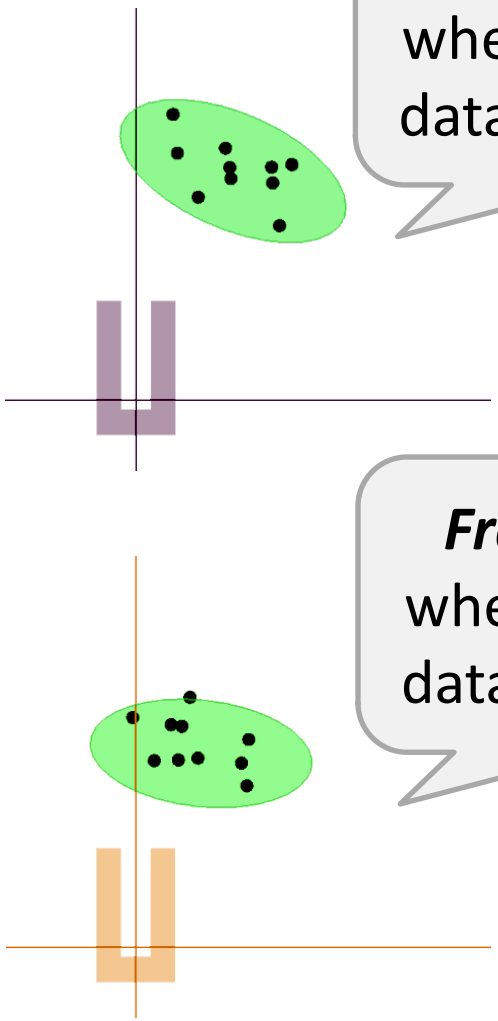


The product of two Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ and $\mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$ is defined by

$$c\,\mathcal{N}(\boldsymbol{\mu}^P, \boldsymbol{\Sigma}^P) = \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}) \cdot \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)}),$$

$$\begin{aligned} \text{with} \qquad c &= \mathcal{N}(\boldsymbol{\mu}^{(1)} | \boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(1)} + \boldsymbol{\Sigma}^{(2)}), \\ \boldsymbol{\Sigma}^P &= \left({\boldsymbol{\Sigma}^{(1)}}^{-1} + {\boldsymbol{\Sigma}^{(2)}}^{-1} \right)^{-1}, \\ \boldsymbol{\mu}^P &= \boldsymbol{\Sigma}^P \left({\boldsymbol{\Sigma}^{(1)}}^{-1} \boldsymbol{\mu}^{(1)} + {\boldsymbol{\Sigma}^{(2)}}^{-1} \boldsymbol{\mu}^{(2)} \right). \end{aligned}$$

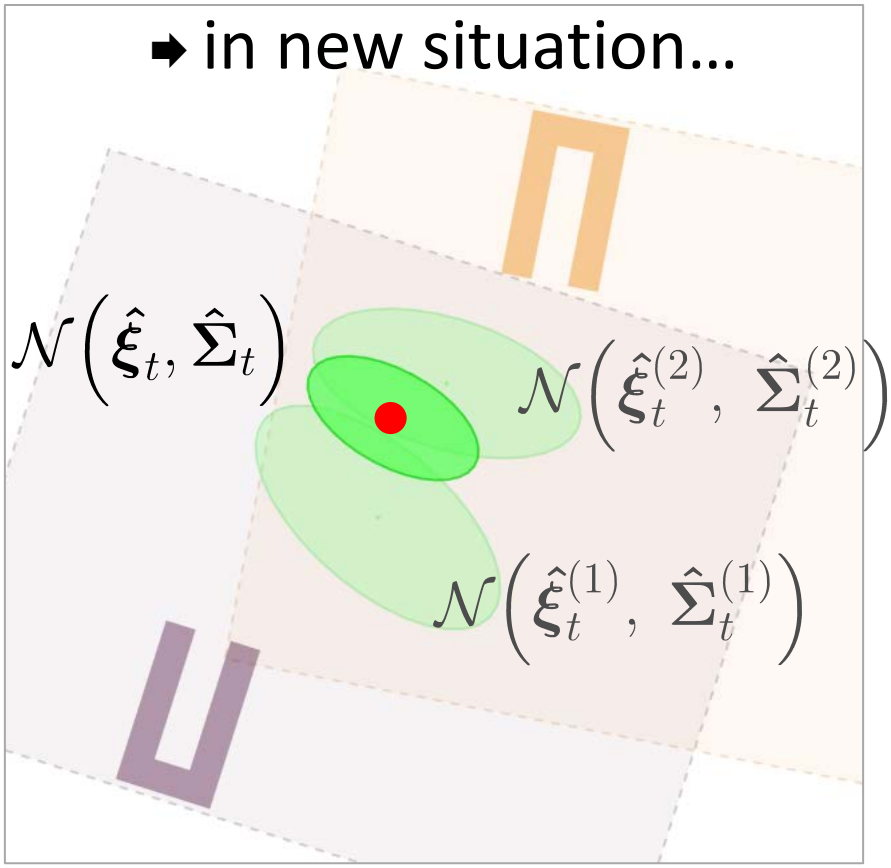
Product of Gaussians



Frame 1: This is where I expect the data to be located!

Frame 2: This is where I expect the data to be located!

➡ in new situation…

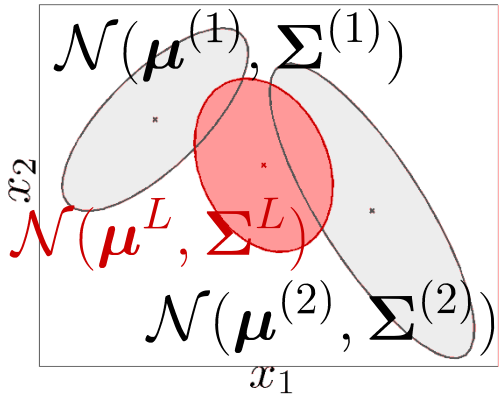


$$\hat{\xi}_t = \arg\min_{\xi_t} \sum_{j=1}^{2} \left(\xi_t - \hat{\xi}_t^{(j)}\right)^\top \hat{\Sigma}_t^{(j)^{-1}} \left(\xi_t - \hat{\xi}_t^{(j)}\right)$$

→ **Product of Gaussians**

[Calinon, S. (2015). A Tutorial on Task-Parameterized Movement Learning and Retrieval. Intelligent Service Robotics.]

Linear combination



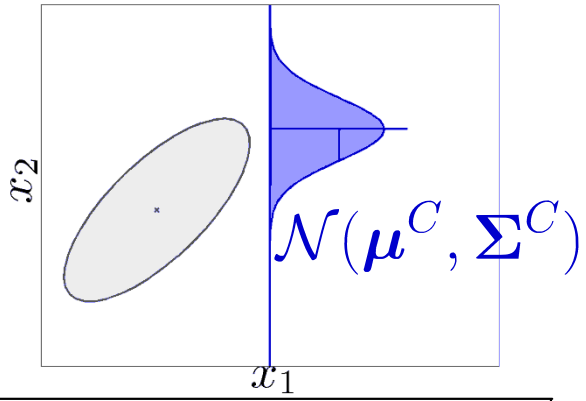
If $\boldsymbol{x}^{(1)} \sim \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ and $\boldsymbol{x}^{(2)} \sim \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$, the linear transformation $\boldsymbol{A}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{c}$ follows the distribution

$$\boldsymbol{A}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{c} \sim \mathcal{N}(\boldsymbol{\mu}^{L}, \boldsymbol{\Sigma}^{L}),$$

with

$$\begin{aligned} \boldsymbol{\mu}^{L} &= \boldsymbol{A}^{(1)}\boldsymbol{\mu}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{\mu}^{(2)} + \boldsymbol{c}, \\ \boldsymbol{\Sigma}^{L} &= \boldsymbol{A}^{(1)}\boldsymbol{\Sigma}^{(1)}\boldsymbol{A}^{(1)^{\top}} + \boldsymbol{A}^{(2)}\boldsymbol{\Sigma}^{(2)}\boldsymbol{A}^{(2)^{\top}}. \end{aligned}$$

Conditional probability



Let $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ be defined by

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{pmatrix}, \ \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \ \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}.$$

The conditional probability $\mathcal{P}(\boldsymbol{x}_2|\boldsymbol{x}_1)$ is defined by

$$\mathcal{P}(\boldsymbol{x}_2|\boldsymbol{x}_1) \ \sim \ \mathcal{N}(\boldsymbol{\mu}^C, \boldsymbol{\Sigma}^C),$$

$$\text{with} \qquad \begin{aligned} \boldsymbol{\mu}^C &= \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21}(\boldsymbol{\Sigma}_{11})^{-1}(\boldsymbol{x}_1 - \boldsymbol{\mu}_1), \\ \boldsymbol{\Sigma}^C &= \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}(\boldsymbol{\Sigma}_{11})^{-1}\boldsymbol{\Sigma}_{12}. \end{aligned}$$

Gaussian mixture regression (GMR)

Gaussian mixture regression (GMR) offers a simple solution to generate data from a GMM. It relies on basic properties of normal distributions (linear transformation and conditioning), and provides a probabilistic synthesis mechanism in which the model can compute output distributions in an online manner, with a **computation time independent of the number of datapoints** used to train the model.

A characteristic of GMR is that it does not model the regression function directly. It models the joint probability density of the data, and then **derives the regression function from the joint density model**.

The estimation of the model parameters can be achieved in an offline phase that depends linearly on the number of datapoints. **Regression can then be computed very rapidly**, which makes the approach an interesting alternative to regression methods such as GPR whose processing grows with the size of the dataset.

Gaussian mixture regression (GMR)

In GMR, both input and output variables can be multidimensional. Any subset of input-output dimensions can be selected, which can change, if required, at each iteration during reproduction.

It can for example handle different sources of missing data, as the system can consider during the retrieval phase **any combination of input-output mappings**, where expectations on the remaining dimensions can be computed very efficiently.

In the following, we will use a block decomposition of the datapoint $\boldsymbol{\xi}_t$, vectors $\boldsymbol{\mu}_i$ and matrices $\boldsymbol{\Sigma}_i$, which can be written as

$$\boldsymbol{\xi}_t = \begin{bmatrix} \boldsymbol{\xi}_t^{\mathcal{I}} \\ \boldsymbol{\xi}_t^{\mathcal{O}} \end{bmatrix} \quad , \quad \boldsymbol{\mu}_i = \begin{bmatrix} \boldsymbol{\mu}_i^{\mathcal{I}} \\ \boldsymbol{\mu}_i^{\mathcal{O}} \end{bmatrix} \quad , \quad \boldsymbol{\Sigma}_i = \begin{bmatrix} \boldsymbol{\Sigma}_i^{\mathcal{I}} & \boldsymbol{\Sigma}_i^{\mathcal{I}\mathcal{O}} \\ \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}} & \boldsymbol{\Sigma}_i^{\mathcal{O}} \end{bmatrix}$$

We will consider here the example of time-based trajectory retrieval.

Gaussian mixture regression (GMR)

At each iteration step t during reproduction, $\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}})$ can be computed as the conditional distribution

$$\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) \sim \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\, \mathcal{N}\left(\hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}), \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right)$$

$$\text{with} \quad \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}) = \boldsymbol{\mu}_i^{\mathcal{O}} + \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}}\boldsymbol{\Sigma}_i^{\mathcal{I}-1}(\boldsymbol{\xi}_t^{\mathcal{I}} - \boldsymbol{\mu}_i^{\mathcal{I}})$$

$$\hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} = \boldsymbol{\Sigma}_i^{\mathcal{O}} - \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}}\boldsymbol{\Sigma}_i^{\mathcal{I}-1}\boldsymbol{\Sigma}_i^{\mathcal{I}\mathcal{O}}$$

$$\text{and} \quad h_i(\boldsymbol{\xi}_t^{\mathcal{I}}) = \frac{\pi_i \mathcal{N}(\boldsymbol{\xi}_t^{\mathcal{I}}|\ \boldsymbol{\mu}_i^{\mathcal{I}}, \boldsymbol{\Sigma}_i^{\mathcal{I}})}{\sum_k^K \pi_k \mathcal{N}(\boldsymbol{\xi}_t^{\mathcal{I}}|\ \boldsymbol{\mu}_k^{\mathcal{I}}, \boldsymbol{\Sigma}_k^{\mathcal{I}})}$$

$\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}})$ represents here a multimodal distribution.

Gaussian mixture regression (GMR)

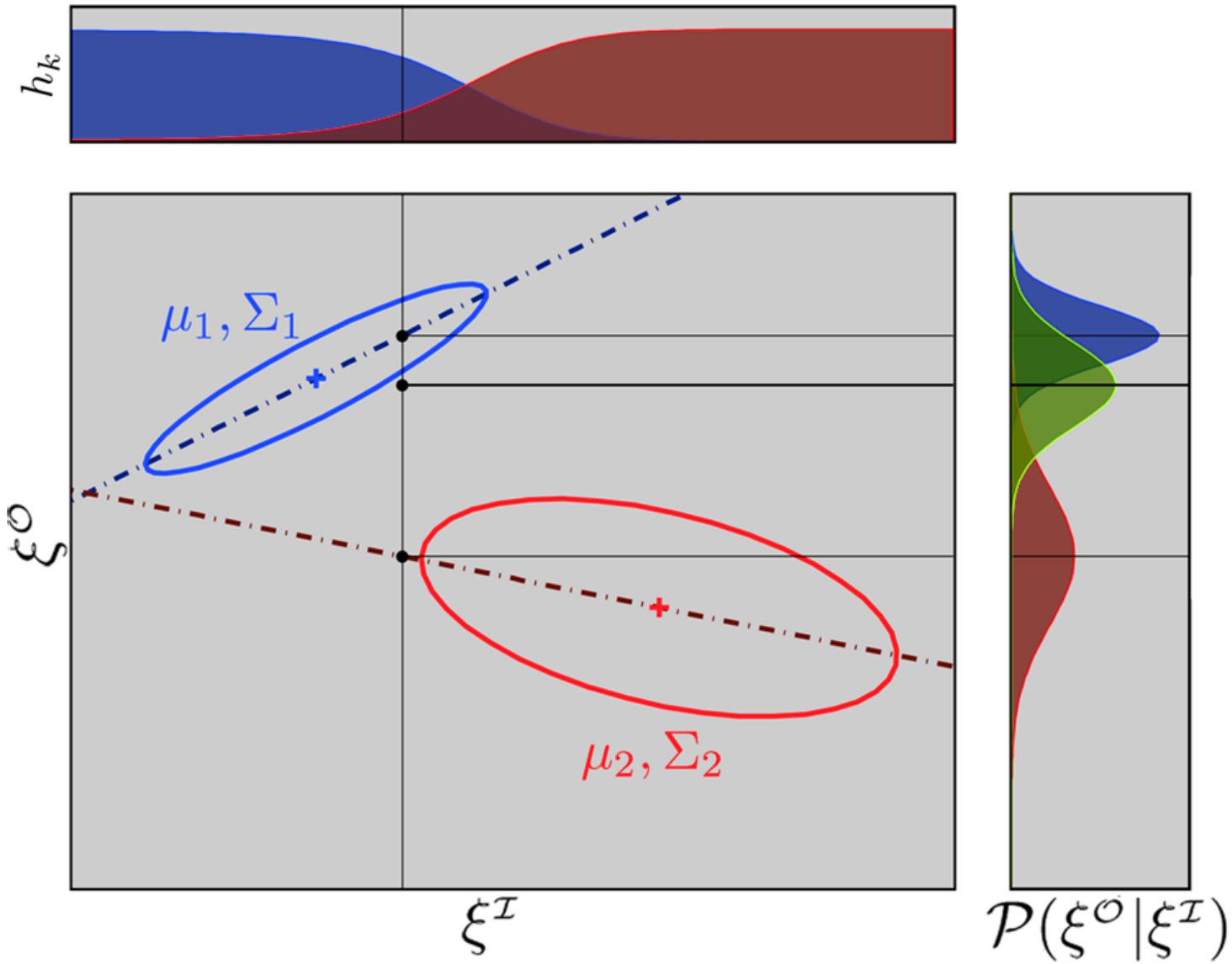
For problems in which a single peaked output distribution is preferred, the formulation can be approximated by a normal distribution

$$\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \mathcal{N}\left(\boldsymbol{\xi}_t^{\mathcal{O}}|\ \hat{\boldsymbol{\mu}}_t^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}_t^{\mathcal{O}}\right) \quad \text{with}$$

$$\hat{\boldsymbol{\mu}}_t^{\mathcal{O}} = \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\ \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})$$

$$\hat{\boldsymbol{\Sigma}}_t^{\mathcal{O}} = \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\left(\hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} + \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})\ \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})^{\top}\right) - \hat{\boldsymbol{\mu}}_t^{\mathcal{O}}\,\hat{\boldsymbol{\mu}}_t^{\mathcal{O}\top}$$

Gaussian mixture regression (GMR)



Gaussian mixture regression (GMR)

To demonstrate the GMR result, we will consider a datapoint $\boldsymbol{\xi}_t$ distributed as for the Gaussian product, with $\mathcal{P}(\boldsymbol{\xi}_t) = \mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}}, \boldsymbol{\xi}_t^{\mathcal{O}})$ the joint distribution describing the data. The conditional probability of an output given an input is

$$\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \frac{\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}}, \boldsymbol{\xi}_t^{\mathcal{O}})}{\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}})} = \frac{\sum_{i=1}^{K} \mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}}, \boldsymbol{\xi}_t^{\mathcal{O}}|z_i)\mathcal{P}(z_i)}{\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}})}$$

where z_i represents the i-th component of the GMM. Namely,

$$\begin{aligned}\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) &= \sum_{i=1}^{K} \mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}, z_i) \frac{\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}}|z_i)\mathcal{P}(z_i)}{\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{I}})} \\ &= \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\, \mathcal{N}\left(\hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}), \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right)\end{aligned}$$

Gaussian mixture regression (GMR)

The conditional mean can be computed as

$$\hat{\boldsymbol{\mu}}_t^{\mathcal{O}} = \mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \int \boldsymbol{\xi}_t^{\mathcal{O}}\ \mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}})\ d\boldsymbol{\xi}_t^{\mathcal{O}}$$

$$= \int \boldsymbol{\xi}_t^{\mathcal{O}} \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\ \mathcal{N}\left(\hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}), \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right) d\boldsymbol{\xi}_t^{\mathcal{O}}$$

$$= \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\ \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})$$

$$\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\ \mathcal{N}\left(\hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}), \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right)$$

Gaussian mixture regression (GMR)

In order to evaluate the covariance, we calculate

$$\text{cov}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}^\top}|\boldsymbol{\xi}_t^{\mathcal{I}}) - \mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}})\mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}^\top}|\boldsymbol{\xi}_t^{\mathcal{I}})$$

We have that

$$\begin{aligned}
\mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}^\top}|\boldsymbol{\xi}_t^{\mathcal{I}}) &= \int \boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}^\top}\mathcal{P}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}})\, d\boldsymbol{\xi}_t^{\mathcal{O}} \\
&= \int \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\, \boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}^\top}\mathcal{N}\left(\hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}), \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right) d\boldsymbol{\xi}_t^{\mathcal{O}} \\
&= \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}}) \int \boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}^\top}\mathcal{N}\left(\hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}}), \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right) d\boldsymbol{\xi}_t^{\mathcal{O}}
\end{aligned}$$

Gaussian mixture regression (GMR)

With a Gaussian distribution, we obtain

$$\mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}\top}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} + \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\ \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})\ \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})^{\top}$$

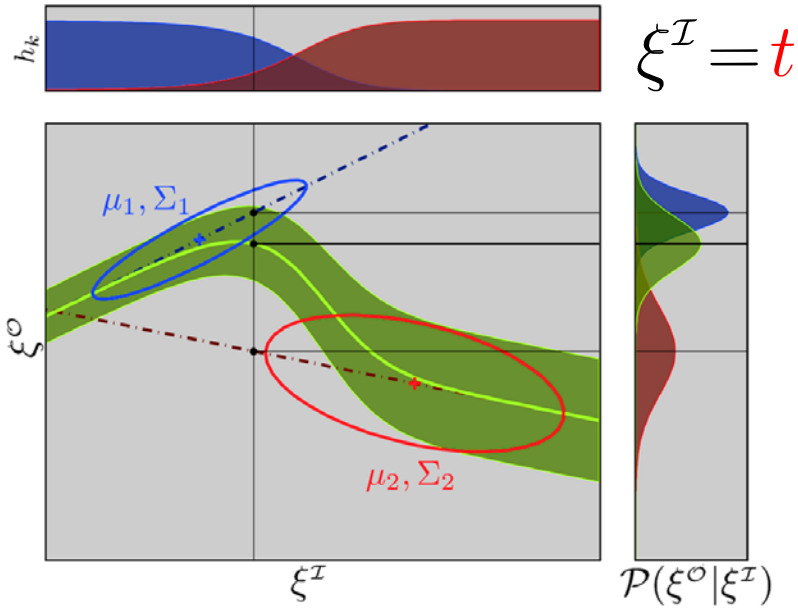
We can then finally show that

$$\hat{\boldsymbol{\Sigma}}_t^{\mathcal{O}} = \mathrm{cov}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \sum_{i=1}^{K} h_i(\boldsymbol{\xi}_t^{\mathcal{I}})\left(\hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} + \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})\ \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}(\boldsymbol{\xi}_t^{\mathcal{I}})^{\top}\right) - \hat{\boldsymbol{\mu}}_t^{\mathcal{O}}\hat{\boldsymbol{\mu}}_t^{\mathcal{O}\top}$$

$$\mathrm{cov}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}}) = \mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}\boldsymbol{\xi}_t^{\mathcal{O}\top}|\boldsymbol{\xi}_t^{\mathcal{I}}) - \mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}}|\boldsymbol{\xi}_t^{\mathcal{I}})\mathbb{E}(\boldsymbol{\xi}_t^{\mathcal{O}\top}|\boldsymbol{\xi}_t^{\mathcal{I}})$$

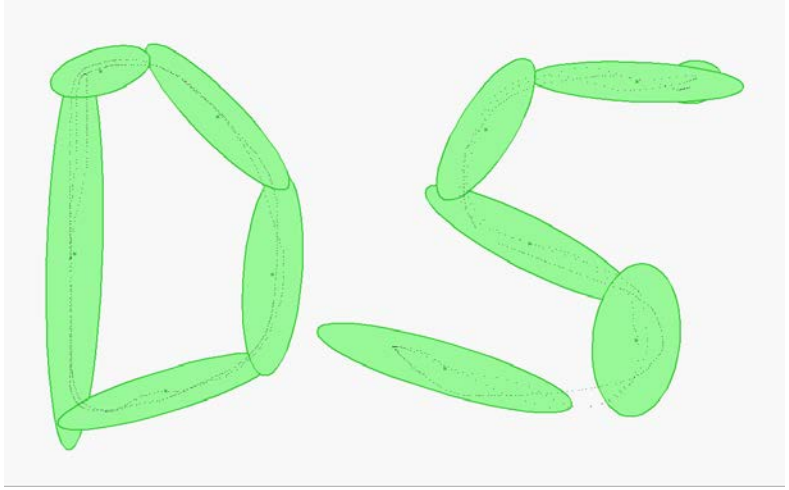
Thus, GMR encapsulates variation and correlation information in the form of full covariance matrices.

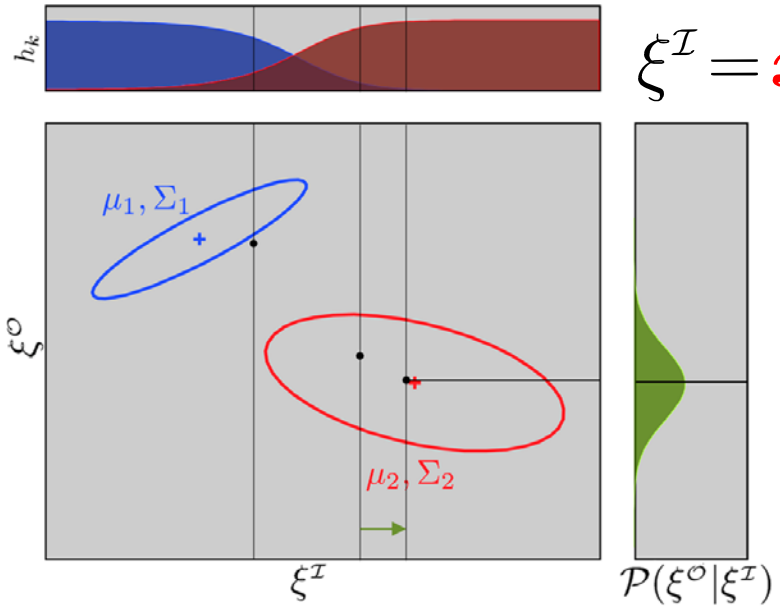
Gaussian mixture regression (GMR)



$\xi^{\mathcal{I}} = t,\ \xi^{\mathcal{O}} = \boldsymbol{x}$

[Calinon, Guenter and Billard, IEEE Trans. on SMC-B 37(2), 2007]





$\xi^{\mathcal{I}} = \boldsymbol{x},\ \xi^{\mathcal{O}} = \dot{\boldsymbol{x}}$

With expectation-maximization (EM):
(maximizing log-likelihood)

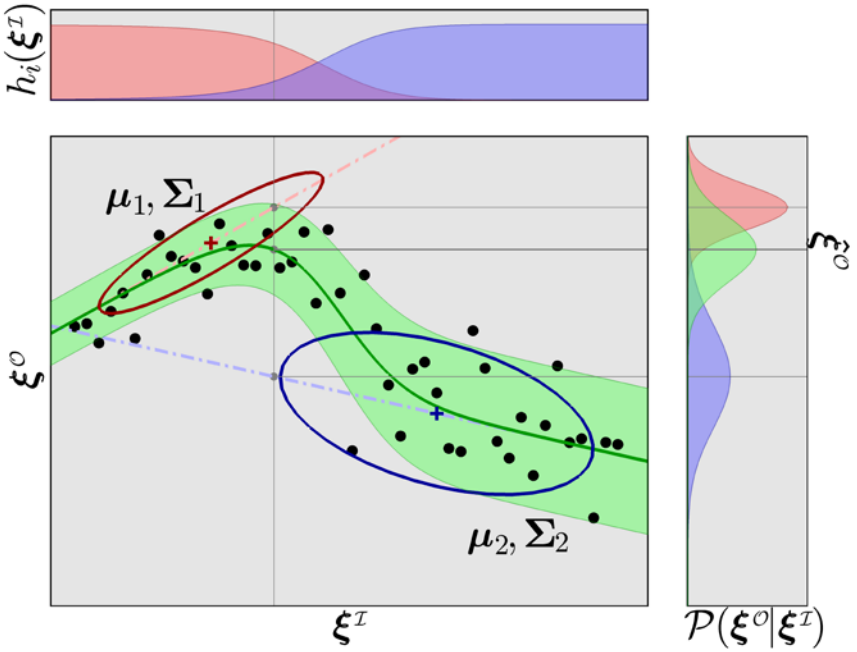
[Hersch, Guenter, Calinon and Billard, IEEE Trans. on Robotics 24(6), 2008]

With quadratic programming solver:
(maximizing log-likelihood s.t. stability constraints)

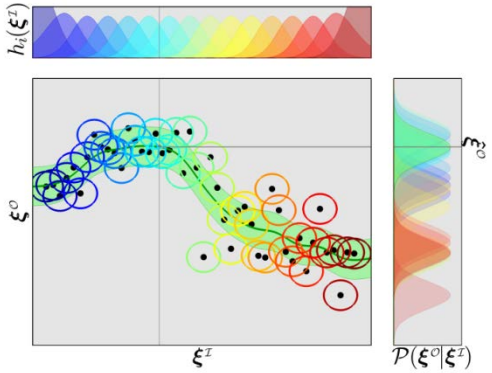
[Khansari-Zadeh and Billard, IEEE Trans. on Robotics 27(5), 2011]

Gaussian mixture regression (GMR)

Least squares
linear regression



Nadaraya-Watson
kernel regression



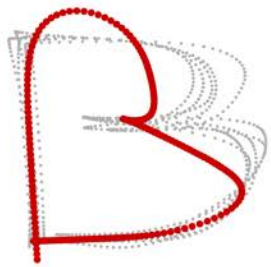
GMR can cover a large spectrum
of regression mechanisms

Both $\xi^{\mathcal{I}}$ and $\xi^{\mathcal{O}}$ can be multidimensional

$\mathcal{P}(\xi^{\mathcal{I}}, \xi^{\mathcal{O}})$ encoded in **Gaussian mixture model (GMM)**

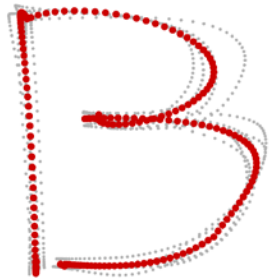
$\mathcal{P}(\xi^{\mathcal{O}}|\xi^{\mathcal{I}})$ retrieved by **Gaussian mixture regression (GMR)**

Gaussian mixture regression (GMR)

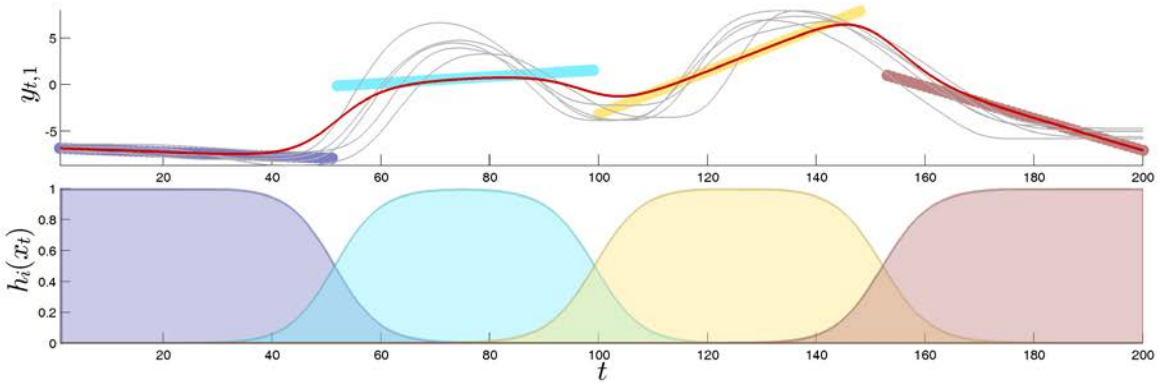


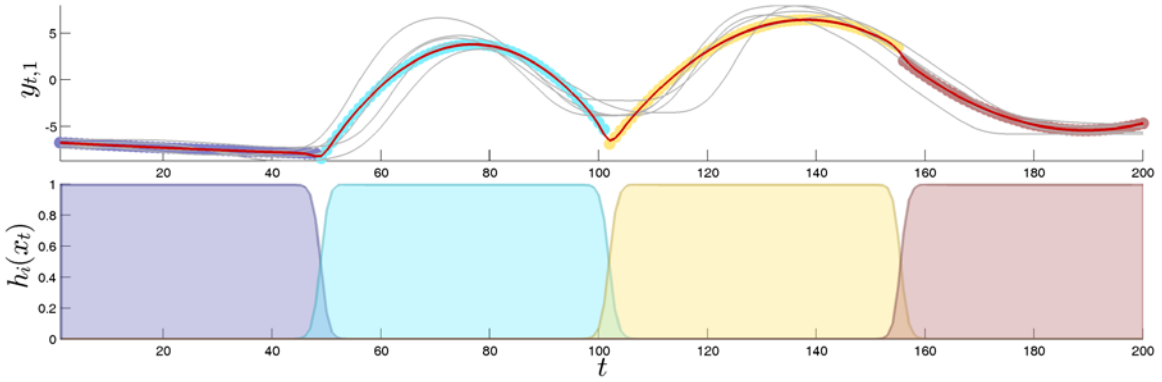
$$\boldsymbol{X} = \boldsymbol{x}$$

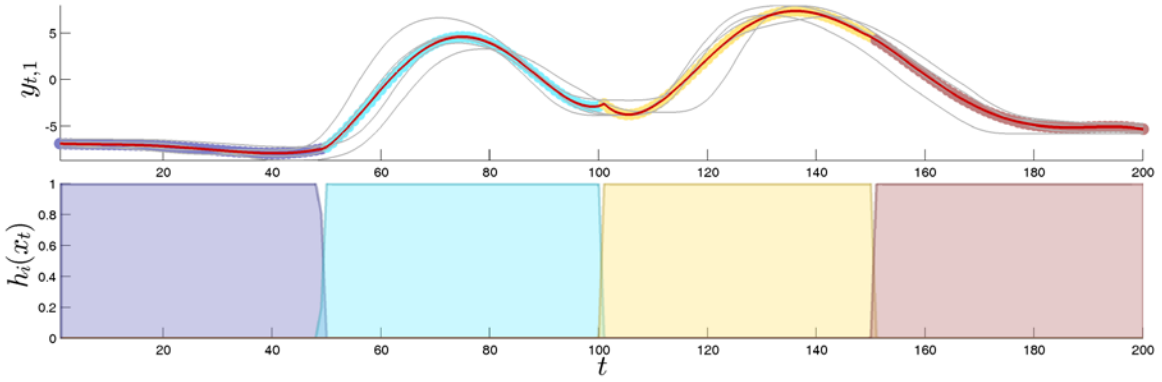
$$\boldsymbol{X} = [\boldsymbol{x}^2, \boldsymbol{x}]$$



$$\boldsymbol{X} = [\boldsymbol{x}^3, \boldsymbol{x}^2, \boldsymbol{x}]$$







Gaussian process regression (GPR)

demo_GPR01.m

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

[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]

Gaussian process regression (GPR)

We consider the regression problem of the form $\boldsymbol{y} = f(\boldsymbol{x}) + \boldsymbol{\eta}$, with f an unknown function and $\boldsymbol{\eta}$ an additive noise process.

By assuming the existence of a dataset of observations as input-output pairs $\{\boldsymbol{x}_t, \boldsymbol{y}_t\}_{t=1}^{N}$, the goals of inference are:

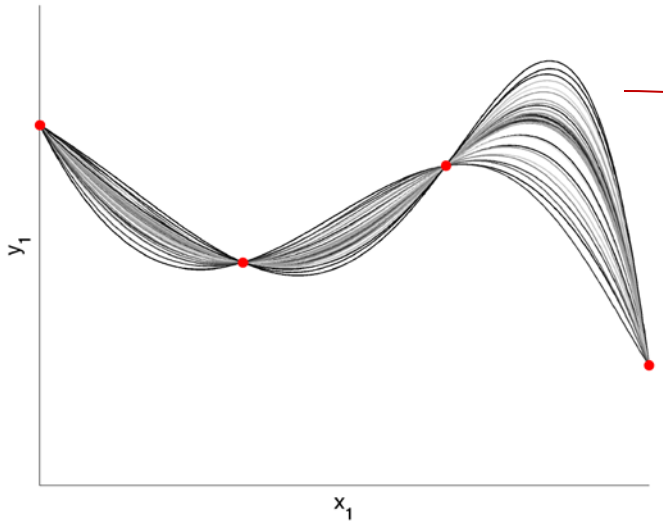
1. to evaluate the form of f
2. to evaluate the distribution of $\boldsymbol{y}^*$ for some $\boldsymbol{x}^*$, i.e., $\mathcal{P}(\boldsymbol{y}^* | \boldsymbol{x}^*)$

We have seen in the last course that the nullspace could be used to generate multiple solutions of a fitting problem, thus obtaining a family of curves differing in regions where we have no observations.

$\rightarrow$ This property can be treated as a distribution of curves, each offering an explanation for the observed data.

$\rightarrow$ Working with such distribution over the curves is central to Bayesian modeling.

Gaussian process regression (GPR)



Polynomial fitting with least squares and nullspace optimization

We have seen polynomial fitting as an example of parametric modeling technique, where we provided the degree of the polynomial.

There are many scenarios in which we have little or no prior knowledge about the appropriate model to use, but where we might still have some domain specific knowledge in a more convenient form.

We will see that Gaussian processes can be used as a way of reflecting various forms of **prior knowledge about the physical process** under investigation.

Gaussian process regression (GPR)

For example, we may know that our observations are samples from an underlying process that is smooth, that is continuous, that has typical amplitude, or that the variations in the function take place over known time scales (e.g., within a typical dynamic range).

$\rightarrow$ We will work mathematically with the **infinite space of all functions** that have these characteristics.

When f can be characterised by less explicit sets of parameters to be inferred, Bayesian non-parametric modeling approaches can be used.

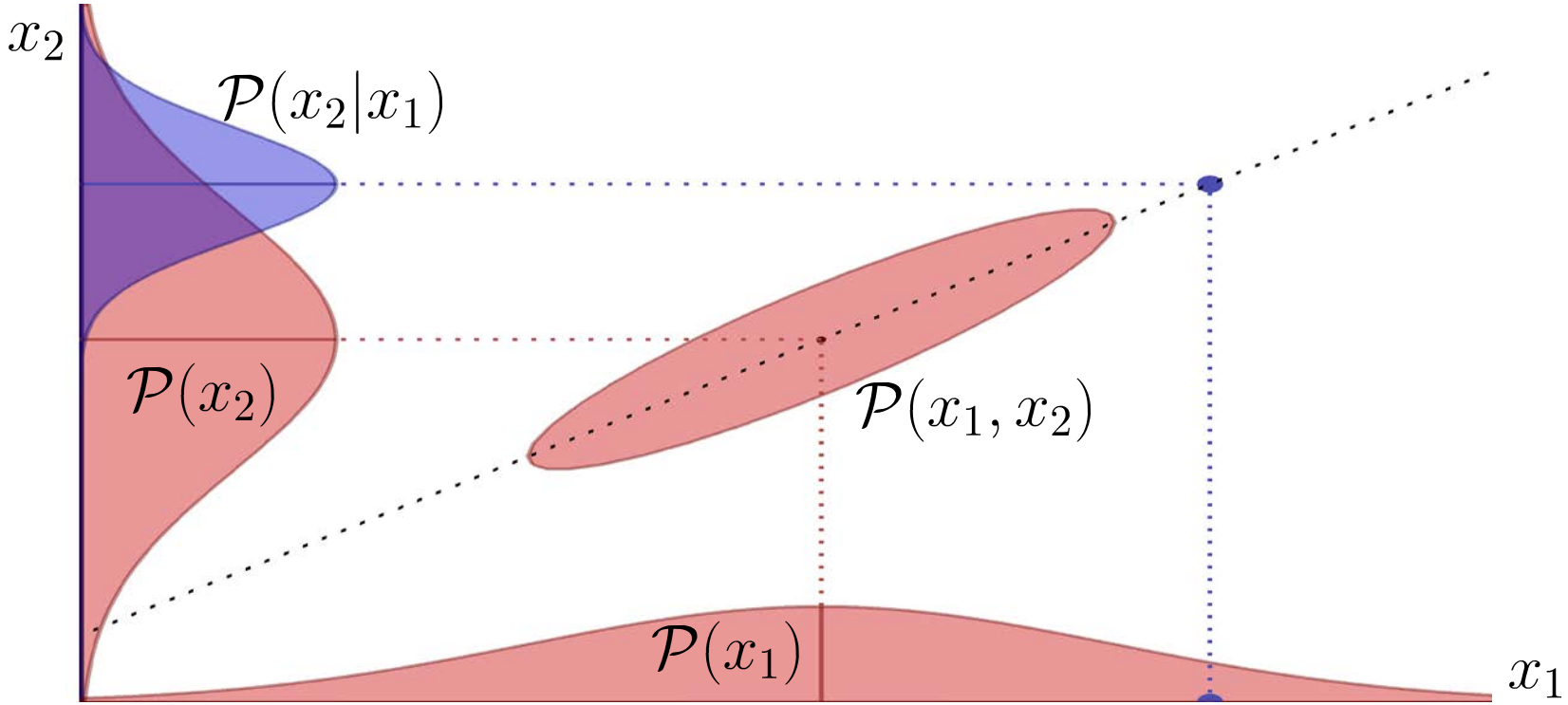
The underlying models still require hyperparameters to be inferred, but **these parameters govern characteristics that are more generic** such as the scale of a distribution rather than acting explicitly on the structure or functional form of the signals.

Gaussian process regression (GPR)

A joint distribution represented by a bivariate Gaussian forms marginal distributions $\mathcal{P}(x_1)$ and $\mathcal{P}(x_2)$ that are unidimensional.

Observing x_1 changes our beliefs about x_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.



Gaussian process regression (GPR)

This 2D example can be extended to arbitrarily large numbers of variables.

Indeed, observations in an arbitrary data set can always be imagined as a single point sampled from some multivariate Gaussian distribution.

The **infinite joint distribution** over all possible variables is equivalent to a **distribution over a function space**.

Similarly to this example, the covariance lies at the core of Gaussian process inference, where a covariance over an arbitrarily large set of variables can be defined through the use of a **covariance kernel function** $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ providing the covariance elements between any two sample locations, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

Gaussian process regression (GPR)

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

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

This means that the entire function evaluation $f(\boldsymbol{x})$ associated with the set of inputs $\boldsymbol{x}$ is a draw from a multivariate Gaussian distribution

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

Therefore a GP specifies a distribution over functions.

Gaussian process regression (GPR)

If we assume there is noise associated with the observed function values $\boldsymbol{y}_t$, we can model this noise term into 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 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.

Gaussian process regression (GPR)

In regression, we are interested in the Gaussian process **posterior distribution** of $\boldsymbol{y}^*$ given some input datapoint(s) $\boldsymbol{x}^*$.

The **joint distribution** of the already observed input-output pair $\boldsymbol{x}$ and $\boldsymbol{y}$ augmented by $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ is

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

As we have seen previously, we can use the conditional probability property of Gaussian distributions to evaluate the posterior distribution over $\boldsymbol{y}^*$, yielding a Gaussian with mean and covariance

$$\begin{aligned} \boldsymbol{y}^* | \boldsymbol{y} &\sim \mathcal{N}\left(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right) \\ \text{with} \quad \boldsymbol{\mu}^* &= \boldsymbol{\mu}(\boldsymbol{x}^*) + \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x})\, \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1} \left(\boldsymbol{y} - \boldsymbol{\mu}(\boldsymbol{x})\right) \\ \boldsymbol{\Sigma}^* &= \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x}^*) - \boldsymbol{K}(\boldsymbol{x}^*,\boldsymbol{x})\, \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x})^{-1} \boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}^*) \end{aligned}$$

Gaussian process regression (GPR)

In the above, $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})$ can be replaced by $\tilde{\boldsymbol{K}}(\boldsymbol{x}, \boldsymbol{x})$ is we assume noise on the observed function values $\boldsymbol{y}$.

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

Gaussian processes can thus be completely defined by their second-order statistics, where the Gram matrix $\boldsymbol{K}$ is a positive semi-definite covariance built on a scalar product of vectors.

Gaussian process regression (GPR)

Which functional form the kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ should take?

The kernel function is chosen to express a property of similarity so that for points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ that are similar, the corresponding values $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ will be more strongly correlated than for dissimilar points.

The notion of similarity will depend on the application. Some of the basic aspects that can be defined through the covariance function k are the process stationarity, isotropy, smoothness or periodicity.

When considering continuous time series, it can usually be assumed that past observations can be informative about current data as a function of how long ago they were observed. This would for example correspond to a **stationary** covariance dependent on the Euclidean distance $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$. The process is then considered as **isotropic** and does not depend on directions between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

A process that is both stationary and isotropic is considered to be **homogeneous**.

Gaussian process regression (GPR)

The most employed covariance function of this type is the squared exponential kernel, also known as **radial basis function (RBF)**

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

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

Θ_1^{GP} sets the maximum allowable covariance. $\rightarrow$ This should be high for functions which cover a broad range on the axis.

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**).

Gaussian process regression (GPR)

When noisy observations $\boldsymbol{y}$ are assumed, the kernel is defined as

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

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).

Periodic kernels is another important family of functions inducing periodic patterns within the behaviour of the process.

In a more general perspective, it is important to note that a complicated covariance function can be defined as a linear combination of other simpler covariance functions, which can be exploited to incorporate different insights about the dataset.

Gaussian process regression (GPR)

Another powerful approach to the construction of kernels is to exploit probabilistic models.

Given a generative model $\mathcal{P}(\boldsymbol{x})$, a valid kernel can be defined as $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathcal{P}(\boldsymbol{x}_i)\mathcal{P}(\boldsymbol{x}_j)$, which can be interpreted as an inner product in the one-dimensional feature space defined by the mapping $\mathcal{P}(\boldsymbol{x})$. Namely, two inputs $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ will be similar if they both have high probabilities.

This approach allows the **application of generative models in a discriminative setting**, thus combining the respective performance of both generative and discriminative models.

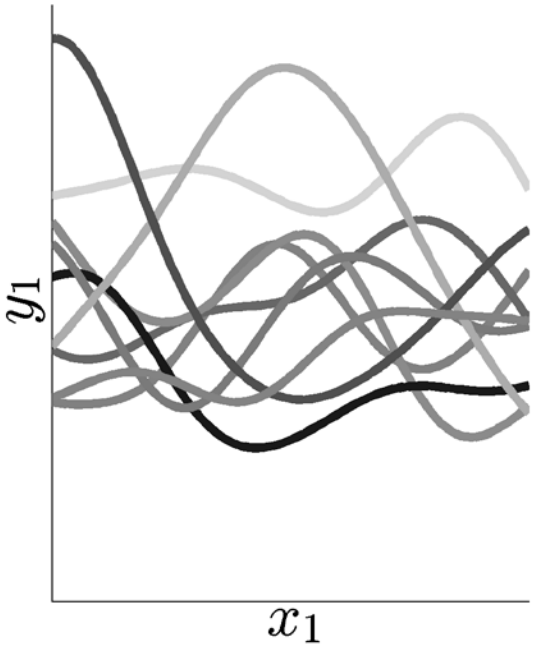
This can bring additional properties to the underlying process such as the capability of handling missing data or partial sequences of various lengths (e.g., with HMM).

Gaussian process regression (GPR)

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

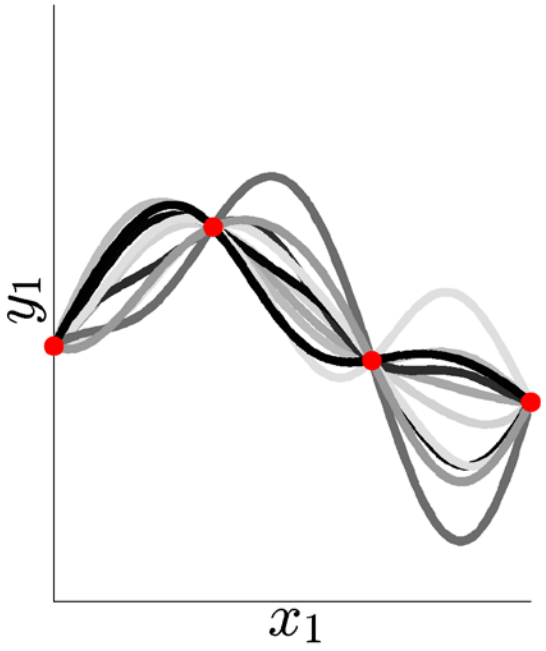
$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$

Samples from prior



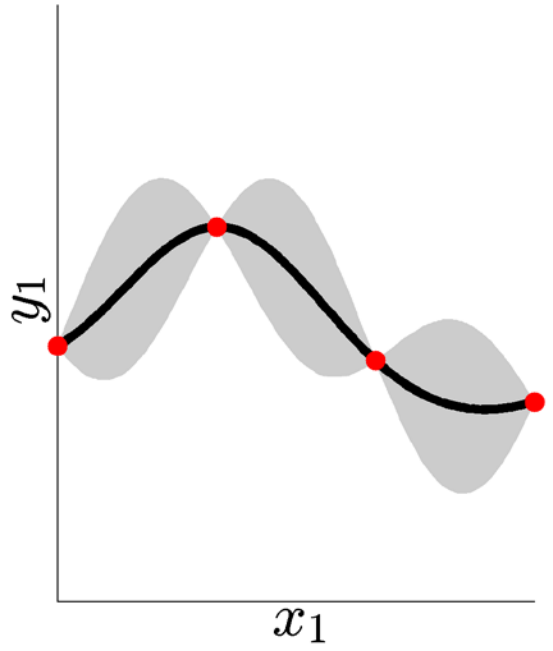
$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Samples from posterior



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

Trajectory distribution



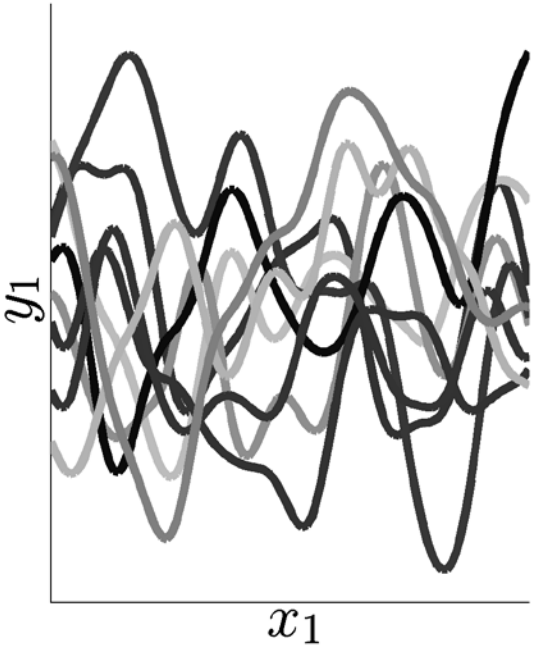
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}}\delta_{i,j}$$

Gaussian process regression (GPR)

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

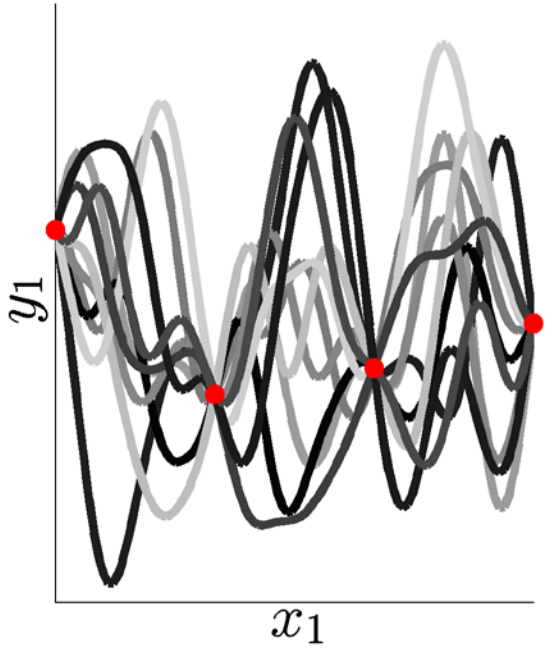
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

Samples from prior



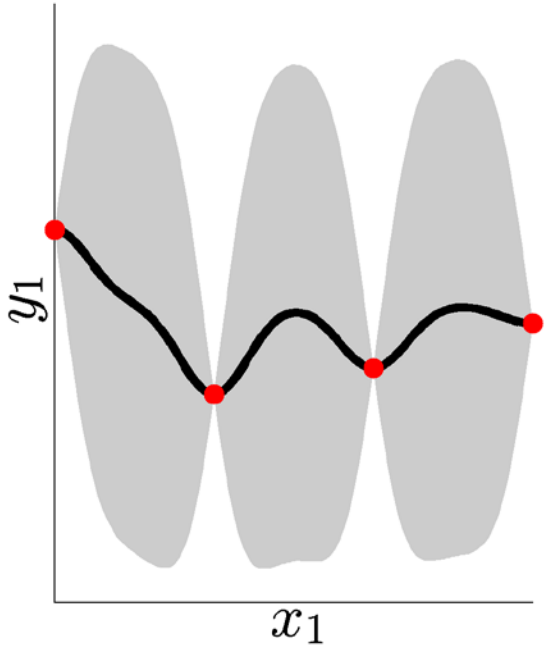
$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Samples from posterior



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

Trajectory distribution



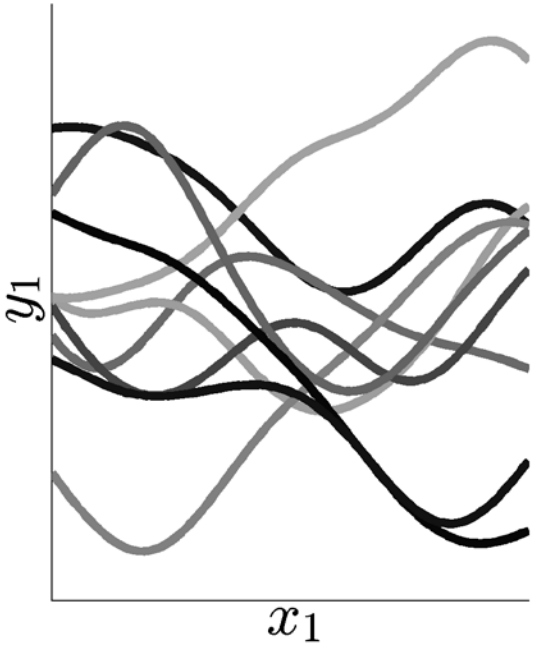
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}} \delta_{i,j}$$

Gaussian process regression (GPR)

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

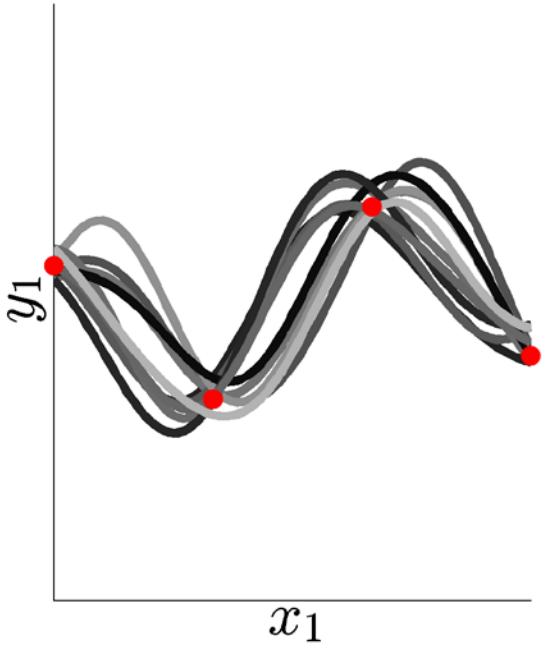
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

Samples from prior



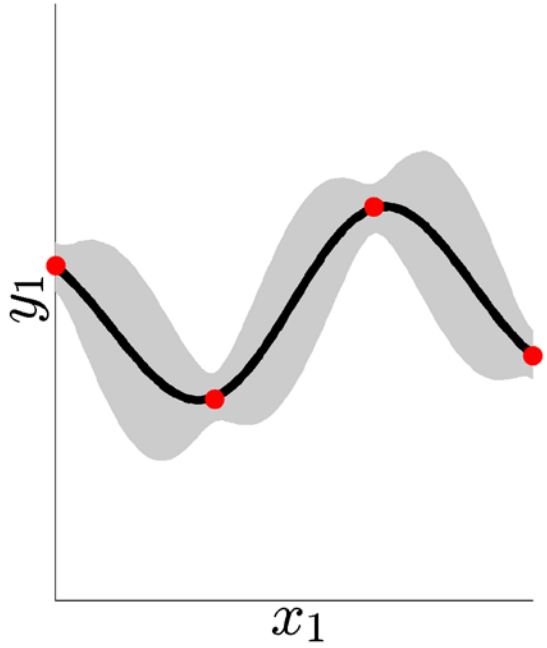
$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Samples from posterior



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

Trajectory distribution



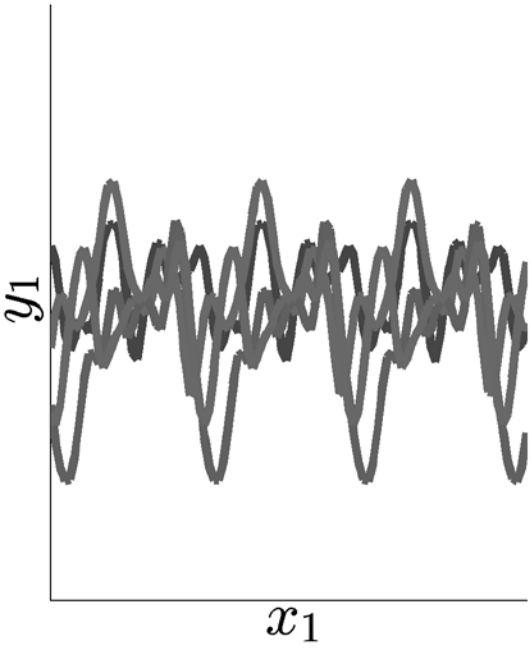
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}} \delta_{i,j}$$

Gaussian process regression (GPR)

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

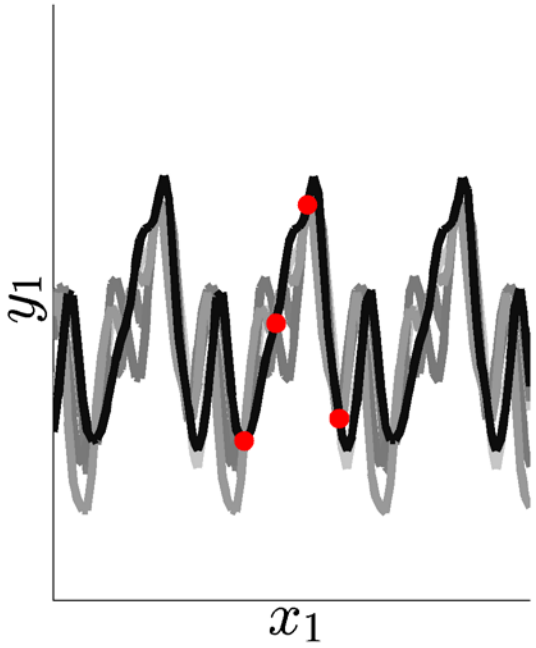
$$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$$

Samples from prior



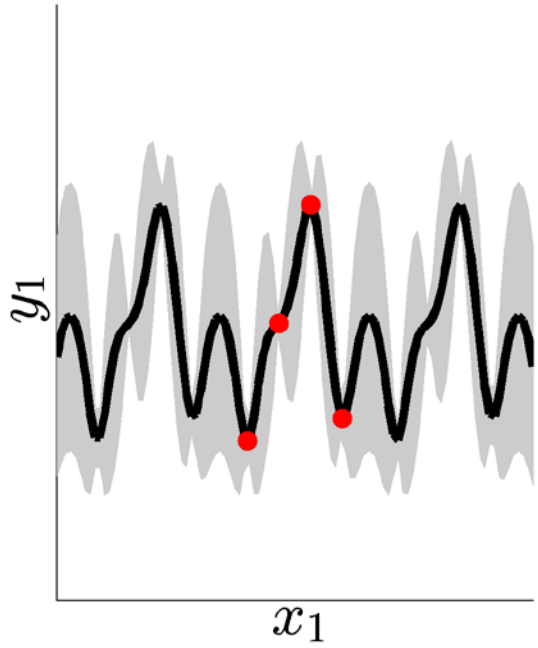
$$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

Samples from posterior



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

Trajectory distribution



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

Main references

Regression

F. Stulp and O. Sigaud. Many regression algorithms, one unified model – a review. Neural Networks, 69:60–79, September 2015

LWR

C. G. Atkeson, A. W. Moore, and S. Schaal. Locally weighted learning for control. Artificial Intelligence Review, 11(1-5):75–113, 1997

GMR

Z. Ghahramani and M. I. Jordan. Supervised learning from incomplete data via an EM approach. In Advances in Neural Information Processing Systems (NIPS), volume 6, pages 120–127, 1994

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

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