EE613 - Machine Learning for Engineers

https://moodle.epfl.ch/course/view.php?id=16819

NONLINEAR REGRESSION

Sylvain Calinon
Robot Learning and Interaction Group
Idiap Research Institute
Nov 9, 2023

EE613 schedule

Thu. 21.09.2023	(C) 1. ML introduction
Thu. 28.09.2023	(C) 2. Bayesian 1 (C) 3. Bayesian 2
Thu. 12.10.2023	(C) 4. Hidden Markov Models
Thu. 19.10.2023	(C) 5. Dimensionality reduction
Thu. 26.10.2023	(C) 6. Decision trees
Thu. 02.11.2023	(C) 7. Linear regression
Thu. 09.11.2023	(C) 8. Nonlinear regression
Thu. 16.11.2023	(C) 9. Kernel Methods - SVM
Thu. 23.11.2023	(C) 10. Tensor factorization
Thu. 30.11.2023	(C) 11. Deep learning 1
Thu. 07.12.2023	(C) 12. Deep learning 2
Thu. 14.12.2023	(C) 13. Deep learning 3
Thu. 21.12.2023	(C) 14. Deep learning 4

Outline

Properties of multivariate Gaussian distributions:

- Product of Gaussians
- Linear transformation and combination
- Conditional distribution

Three nonlinear regression models:

- Locally weighted regression (LWR)
- Gaussian mixture regression (GMR)
- Gaussian process regression (GPR)

Modeling possible co-variations *(a.k.a. aleatoric uncertainty)*

Modeling uncertainty of the estimate *(a.k.a. epistemic uncertainty)*

Multivariate Gaussian distribution

Univariate Gaussian distribution:

$$\mathcal{N}(\mu, \sigma^2) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

Radial basis function (RBF)

$x \in \mathbb{R}$ Datapoint

$\mu \in \mathbb{R}$ Center (or mean)

$\sigma^2 \in \mathbb{R}$ Variance

Parameters $\{\mu, \sigma^2\}$

Multivariate Gaussian distribution:

$$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

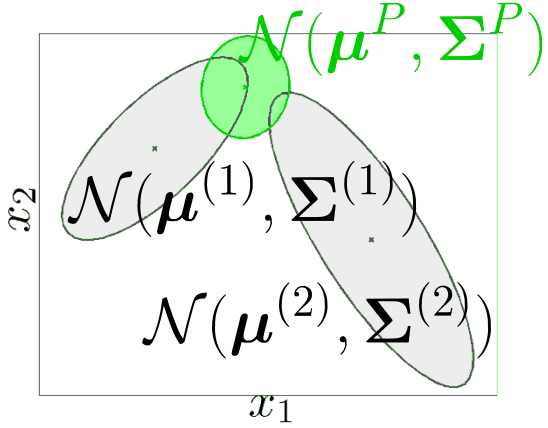
$\boldsymbol{x} \in \mathbb{R}^D$ Datapoint

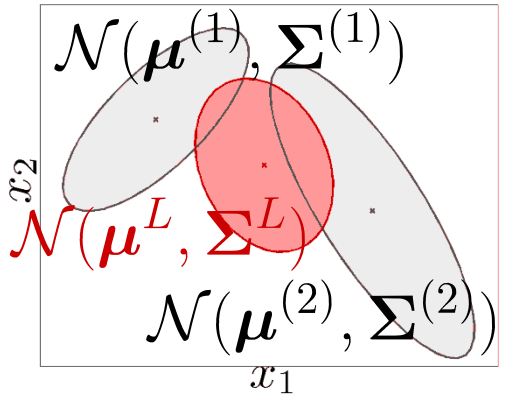
$\boldsymbol{\mu} \in \mathbb{R}^D$ Center (or mean)

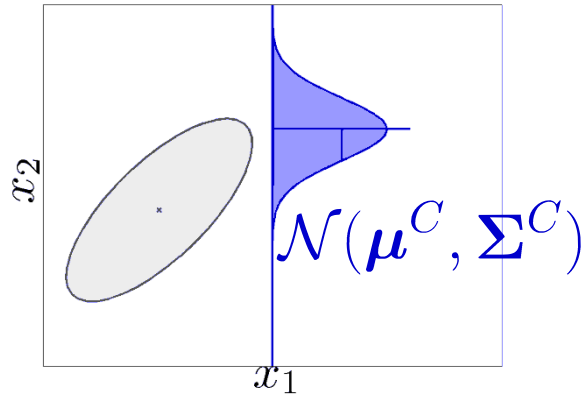
$\boldsymbol{\Sigma} \in \mathbb{R}^{D\times D}$ Covariance matrix

Parameters $\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$

Properties of Gaussian distributions







Linear combination:

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

Product of Gaussians:

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

Conditional probability:

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

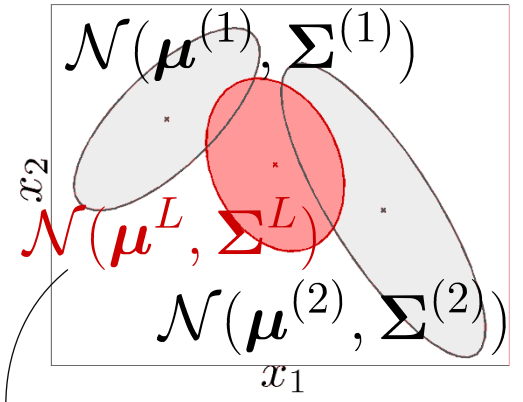
Linear combination

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

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



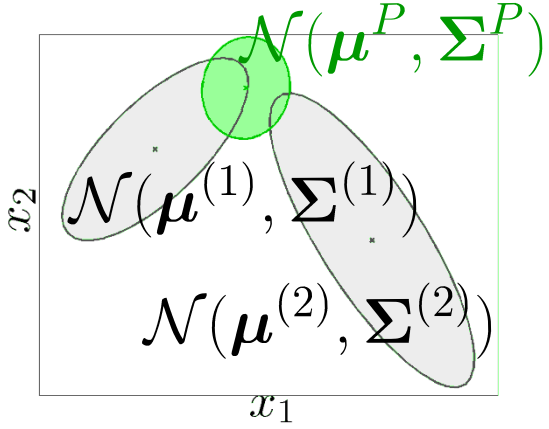
for $\boldsymbol{A}^{(1)} = \boldsymbol{A}^{(2)} = \frac{1}{2}\boldsymbol{I}$ and $\boldsymbol{c} = \boldsymbol{0}$

Product of Gaussians

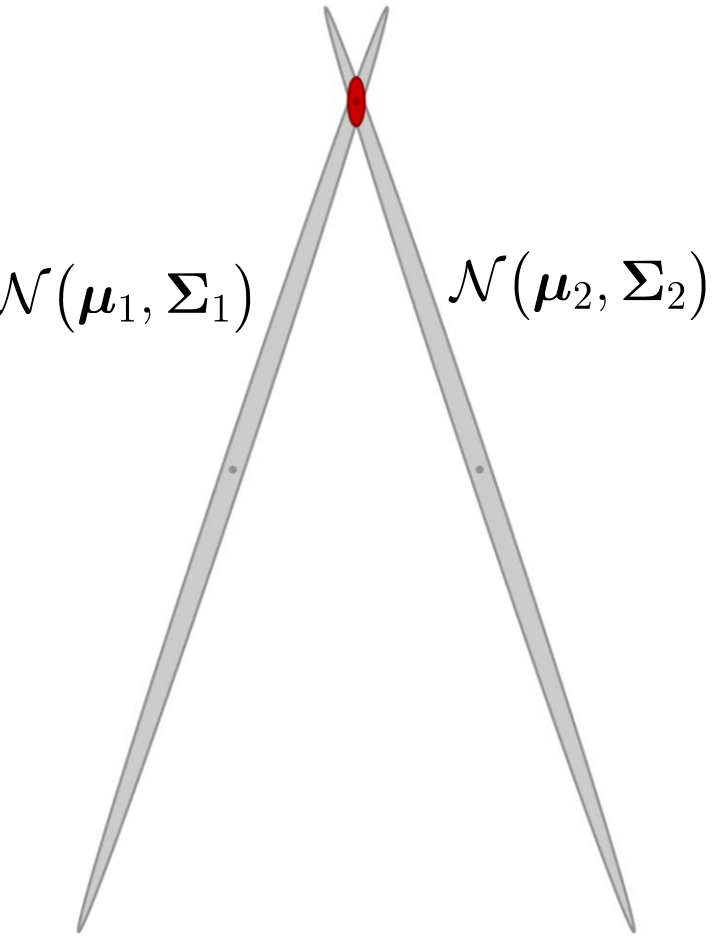
The product of two Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ and $\mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$ is defined by

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

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



Product of Gaussians



μ_i center of Gaussian

Σ_i covariance matrix

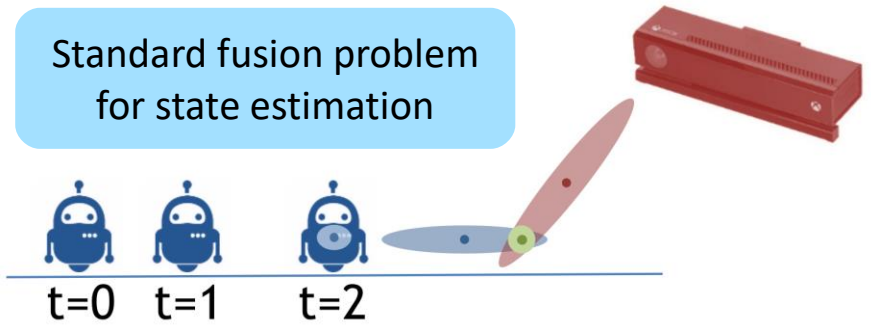
W_i precision matrix $(W_i = \Sigma_i^{-1})$

$$\hat{x} = \arg\min_x \ \|\mu_1 - x\|^2_{W_1} + \|\mu_2 - x\|^2_{W_2}$$

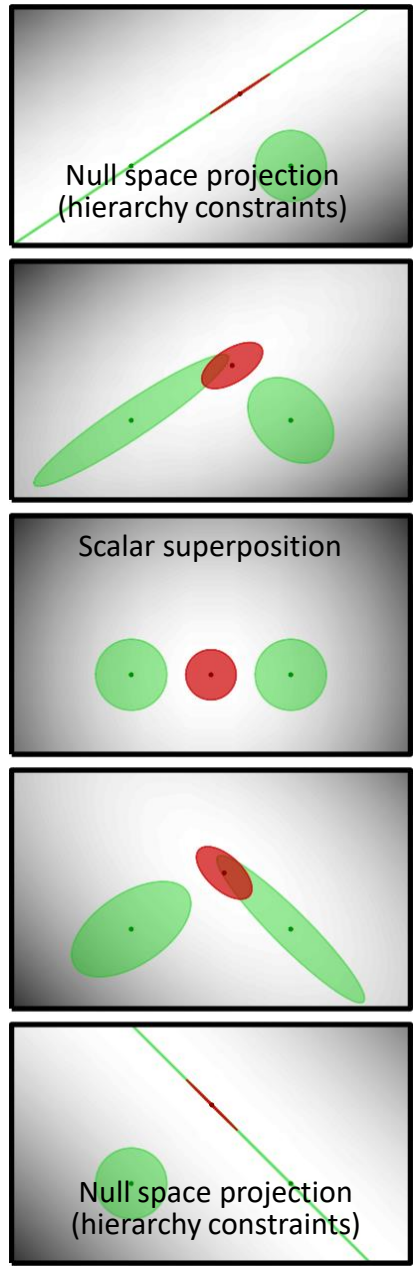
$$= (W_1 + W_2)^{-1}(W_1\mu_1 + W_2\mu_2)$$

→ Product of Gaussians

Standard fusion problem for state estimation



PoG can cover a wide range of behaviors!



Conditional probability

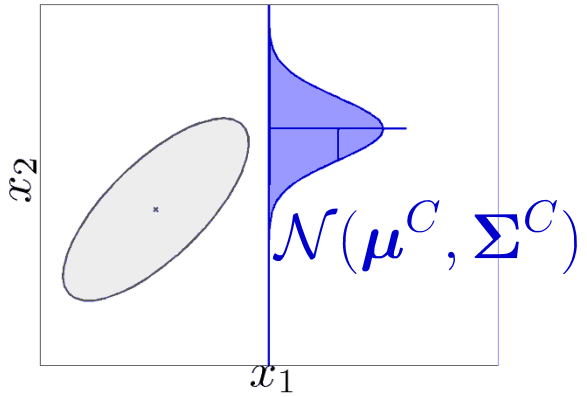
Let $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ be defined by

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

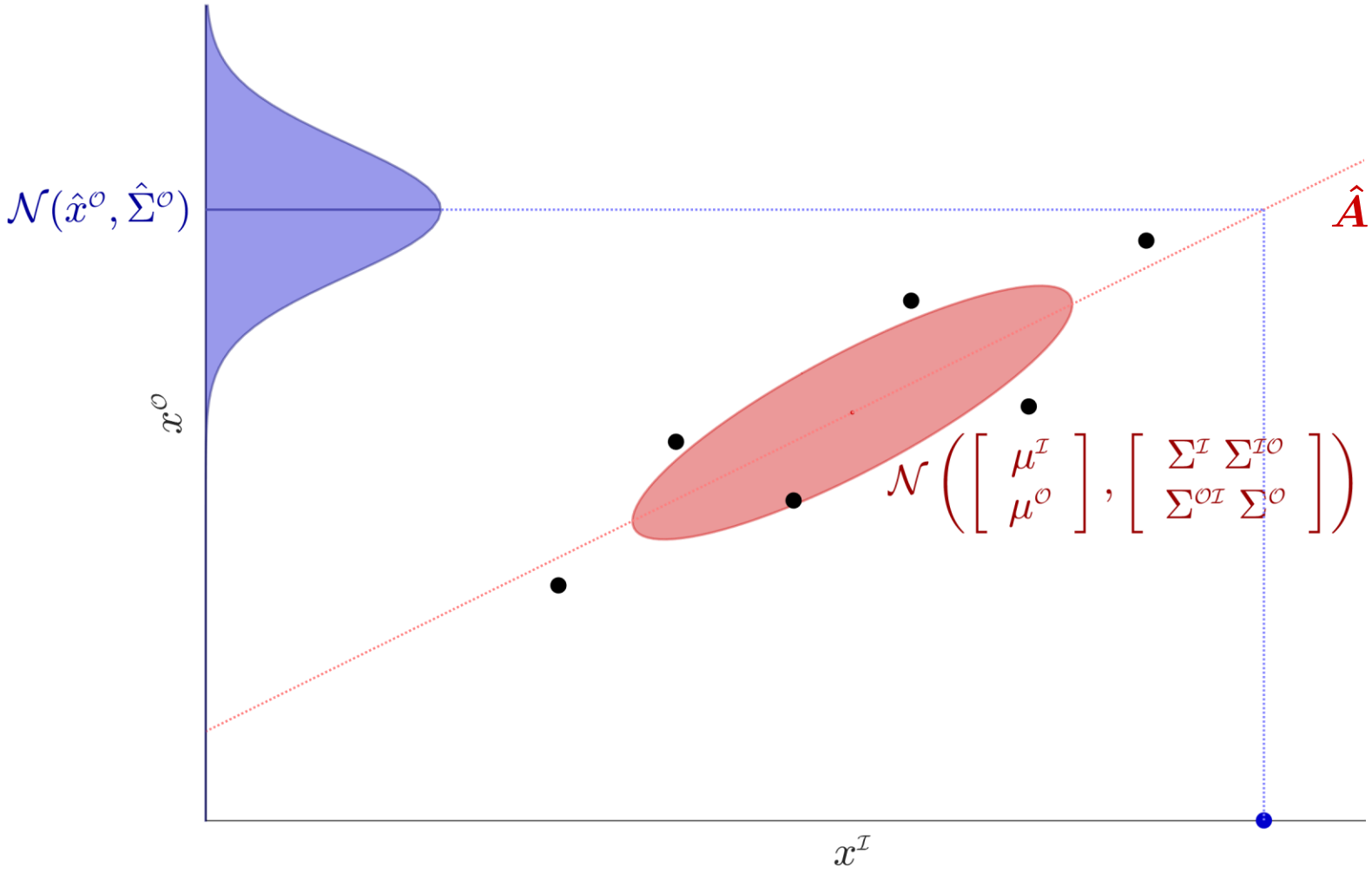
The conditional probability $\mathcal{P}(\boldsymbol{x}_2|\boldsymbol{x}_1)$ is defined by

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

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



Conditional distribution



$$\hat{\boldsymbol{A}} = \arg\min_{\boldsymbol{A}} (\boldsymbol{X}^{\mathcal{O}} - \boldsymbol{X}^{\mathcal{I}}\boldsymbol{A})^{\top}(\boldsymbol{X}^{\mathcal{O}} - \boldsymbol{X}^{\mathcal{I}}\boldsymbol{A})$$
$$= (\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{X}^{\mathcal{I}})^{-1}\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{X}^{\mathcal{O}} = \boldsymbol{X}^{\mathcal{I}\dagger}\boldsymbol{X}^{\mathcal{O}}$$

→ Linear regression from joint distribution

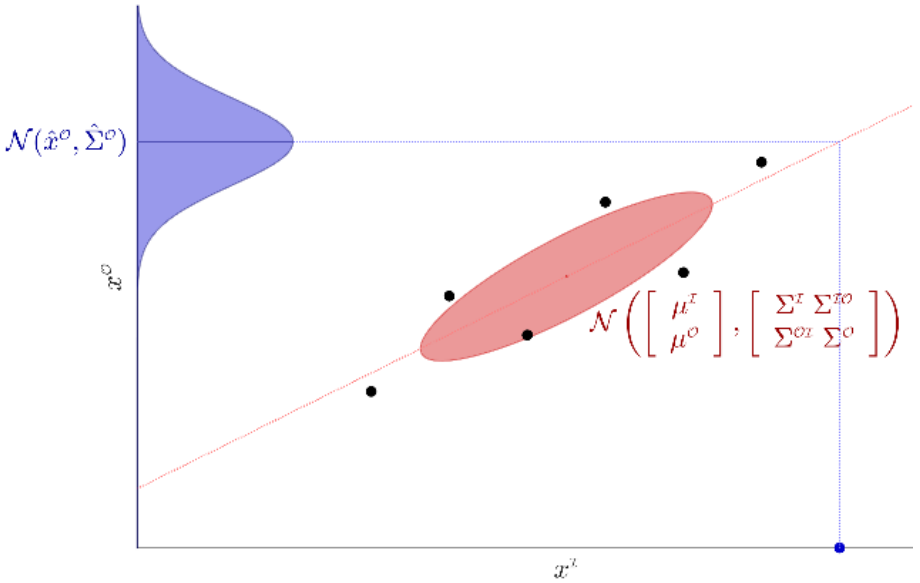
Conditional distribution

We consider multivariate datapoints $\boldsymbol{x}$ and multivariate Gaussian distributions characterized by centers $\boldsymbol{\mu}$ and covariances $\boldsymbol{\Sigma}$, that can be partitioned as

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}^{\mathcal{I}} \\ \boldsymbol{x}^{\mathcal{O}} \end{bmatrix}, \quad \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}^{\mathcal{I}} \\ \boldsymbol{\mu}^{\mathcal{O}} \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}^{\mathcal{I}} & \boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}} \\ \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}} & \boldsymbol{\Sigma}^{\mathcal{O}} \end{bmatrix}.$$

If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we have $\boldsymbol{x}^{\mathcal{O}} | \boldsymbol{x}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\right)$, with parameters

$$\begin{aligned} \hat{\boldsymbol{x}}^{\mathcal{O}} &= \boldsymbol{\mu}^{\mathcal{O}} + \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}^{\mathcal{I}-1} (\boldsymbol{x}^{\mathcal{I}} - \boldsymbol{\mu}^{\mathcal{I}}), \\ \hat{\boldsymbol{\Sigma}}^{\mathcal{O}} &= \boldsymbol{\Sigma}^{\mathcal{O}} - \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}^{\mathcal{I}-1} \boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}}. \end{aligned}$$

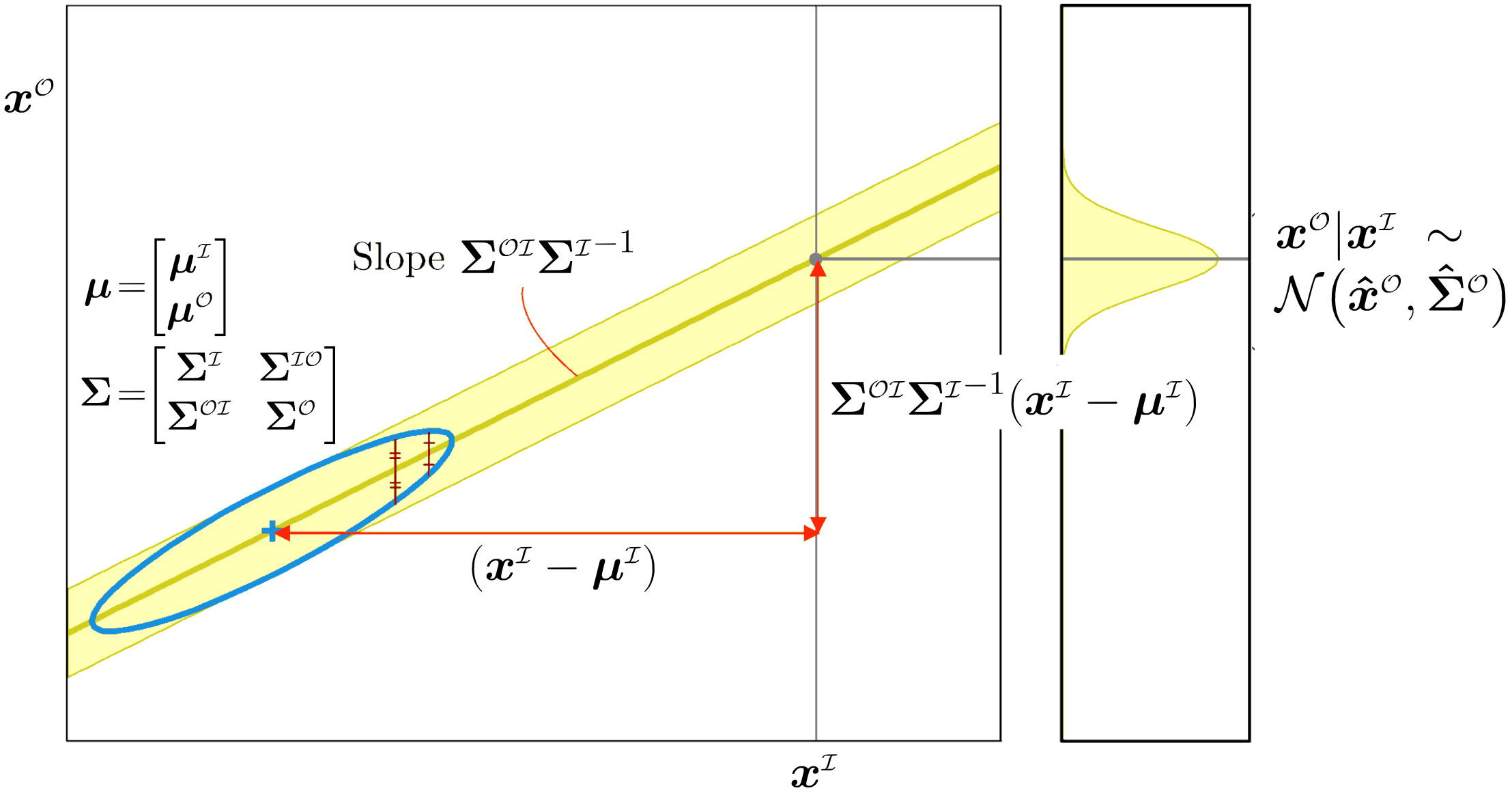


We can see that $\hat{\boldsymbol{x}}^{\mathcal{O}}$ is linearly dependent on $\boldsymbol{x}^{\mathcal{I}}$, and that $\hat{\boldsymbol{\Sigma}}^{\mathcal{O}}$ is independent of $\boldsymbol{x}^{\mathcal{I}}$.

We can also notice that for full joint covariance, the conditional covariance $\hat{\boldsymbol{\Sigma}}^{\mathcal{O}}$ will typically be smaller than the marginal $\boldsymbol{\Sigma}^{\mathcal{O}}$.

Conditional distribution

$$\hat{\boldsymbol{x}}^{\mathcal{O}} = \boldsymbol{\mu}^{\mathcal{O}} + \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}}\boldsymbol{\Sigma}^{\mathcal{I}-1}(\boldsymbol{x}^{\mathcal{I}} - \boldsymbol{\mu}^{\mathcal{I}})$$



$$\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\right)$$

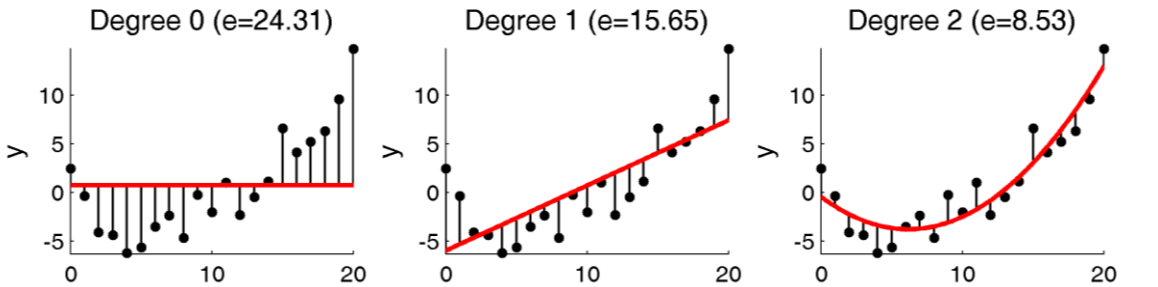
Locally weighted regression (LWR)

Python notebooks:
demo_LWR.ipynb

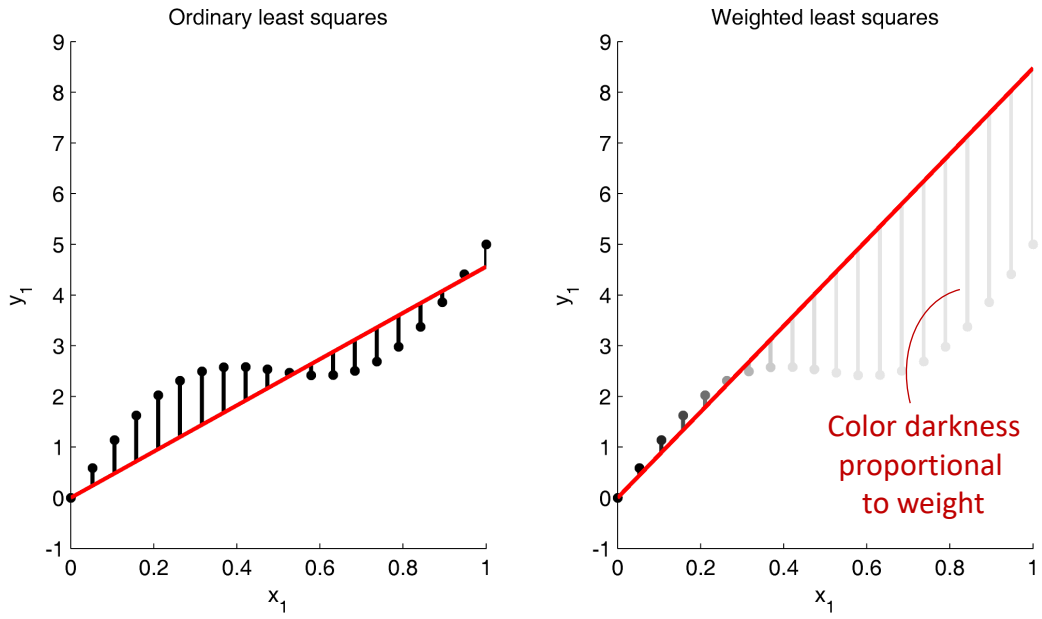
Matlab codes:
demo_LWR01.m

Recap: Linear regression

$$\hat{\boldsymbol{A}} = \arg\min_{\boldsymbol{A}} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{A})^\top (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{A})$$
$$= (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{Y} = \boldsymbol{X}^\dagger \boldsymbol{Y}$$

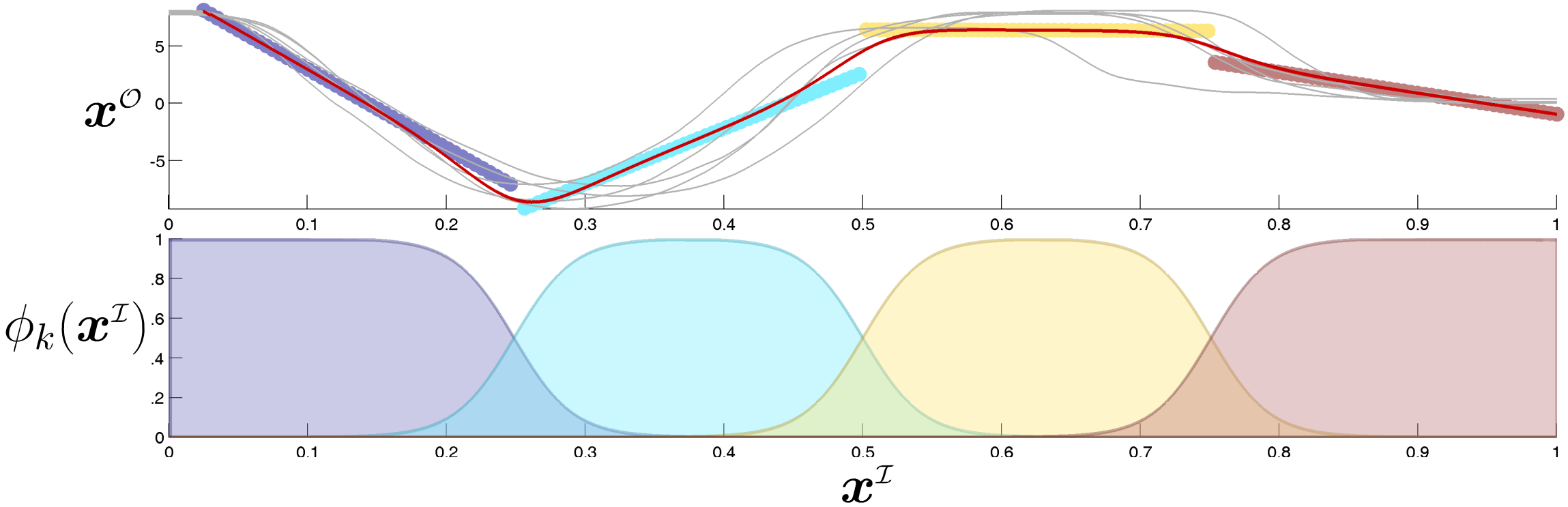


$$\hat{\boldsymbol{A}} = \arg\min_{\boldsymbol{A}} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{A})^\top \boldsymbol{W} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{A})$$
$$= (\boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{Y}$$



Locally weighted regression (LWR)

K weighted regressions with different $\boldsymbol{W}$ are performed on the same dataset $\{\boldsymbol{X}^{\mathcal{I}}, \boldsymbol{X}^{\mathcal{O}}\}$



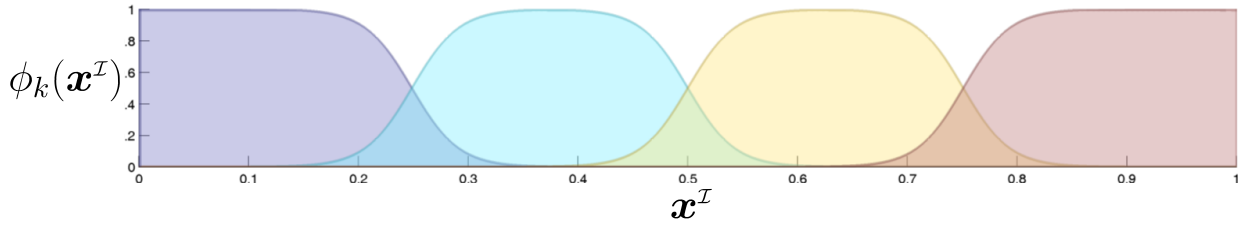
Locally weighted regression (LWR)

LWR computes K estimates $\hat{\boldsymbol{A}}_k$, by using different weighting function $\phi_k(\boldsymbol{x}_n^{\mathcal{I}})$, often defined as the **radial basis functions** (RBF)

$$\tilde{\phi}_k(\boldsymbol{x}_n^{\mathcal{I}}) = \exp\Big(-\frac{1}{2}(\boldsymbol{x}_n^{\mathcal{I}} - \boldsymbol{\mu}_k^{\mathcal{I}})^\top {\boldsymbol{\Sigma}_k^{\mathcal{I}}}^{-1}(\boldsymbol{x}_n^{\mathcal{I}} - \boldsymbol{\mu}_k^{\mathcal{I}})\Big),$$

or in a rescaled form as

$$\phi_k(\boldsymbol{x}_n^{\mathcal{I}}) = \frac{\tilde{\phi}_k(\boldsymbol{x}_n^{\mathcal{I}})}{\sum_{i=1}^{K} \tilde{\phi}_i(\boldsymbol{x}_n^{\mathcal{I}})},$$



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

$\rightarrow$ Nonlinear problem solved locally by linear regression

Locally weighted regression (LWR)

Often, the centroids $\boldsymbol{\mu}_k^{\mathcal{I}}$ are set to uniformly cover the input space, and $\boldsymbol{\Sigma}_k^{\mathcal{I}} = \boldsymbol{I}\sigma^2$ is used as a common bandwidth shared by all basis functions.

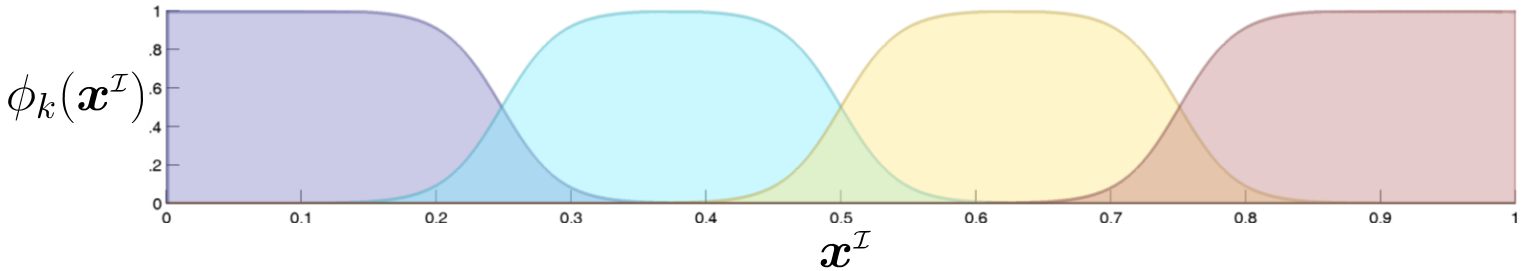
$$\boldsymbol{X}^{\mathcal{I}} = [t_1, t_2, \ldots, t_N]^\top$$
$$\hat{\boldsymbol{A}}_k = \left(\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{W}_k\boldsymbol{X}^{\mathcal{I}}\right)^{-1}\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{W}_k\,\boldsymbol{X}^{\mathcal{O}}$$

An associated diagonal matrix

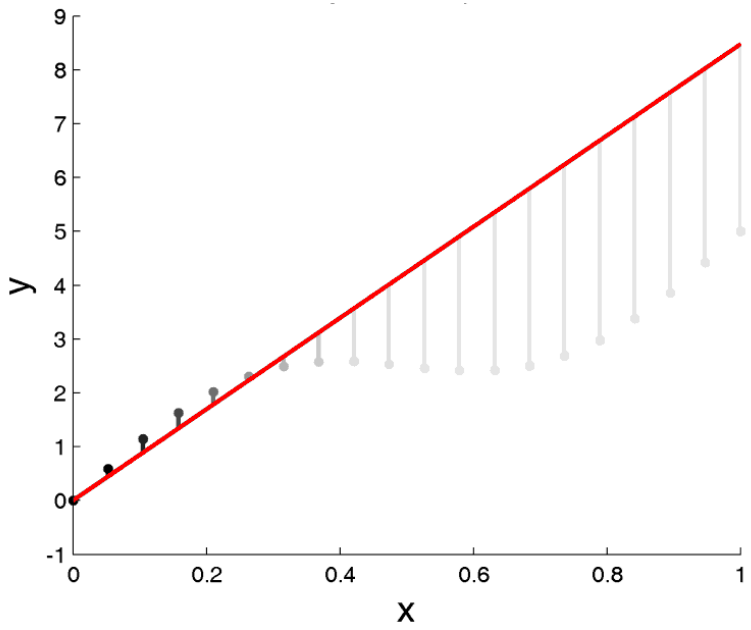
$$\boldsymbol{W}_k = \mathrm{diag}\Big(\phi_k(\boldsymbol{x}_1^{\mathcal{I}}), \phi_k(\boldsymbol{x}_2^{\mathcal{I}}), \ldots, \phi_k(\boldsymbol{x}_N^{\mathcal{I}})\Big)$$

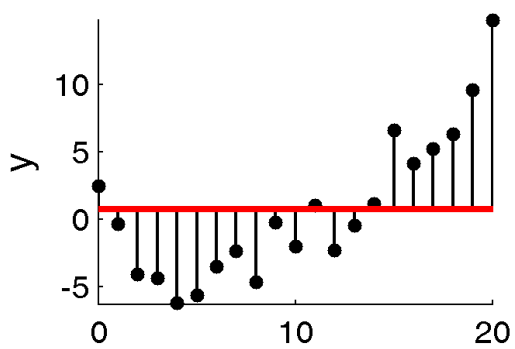
can be used to evaluate $\hat{\boldsymbol{A}}_k$. The result can then be used to compute

$$\boldsymbol{X}^{\mathcal{O}} = \sum_{k=1}^{K} \boldsymbol{W}_k \boldsymbol{X}^{\mathcal{I}} \hat{\boldsymbol{A}}_k$$

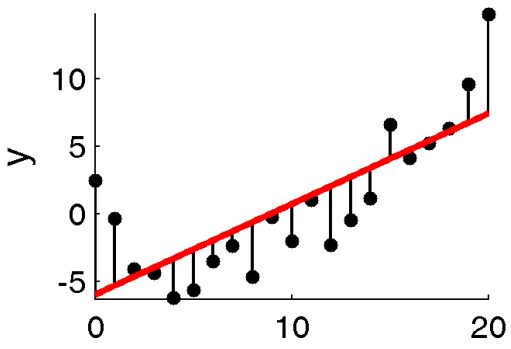


Locally weighted regression (LWR)

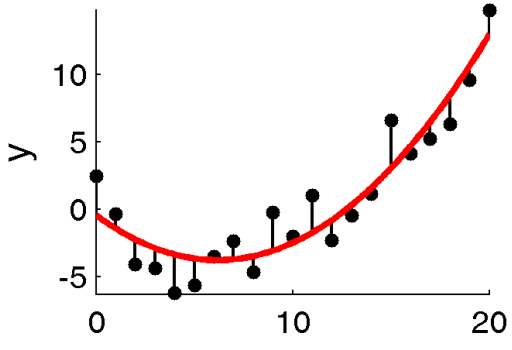




$x = 1$



$\boldsymbol{x} = [1, \mathrm{x}]$



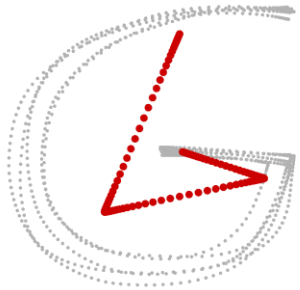
$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$

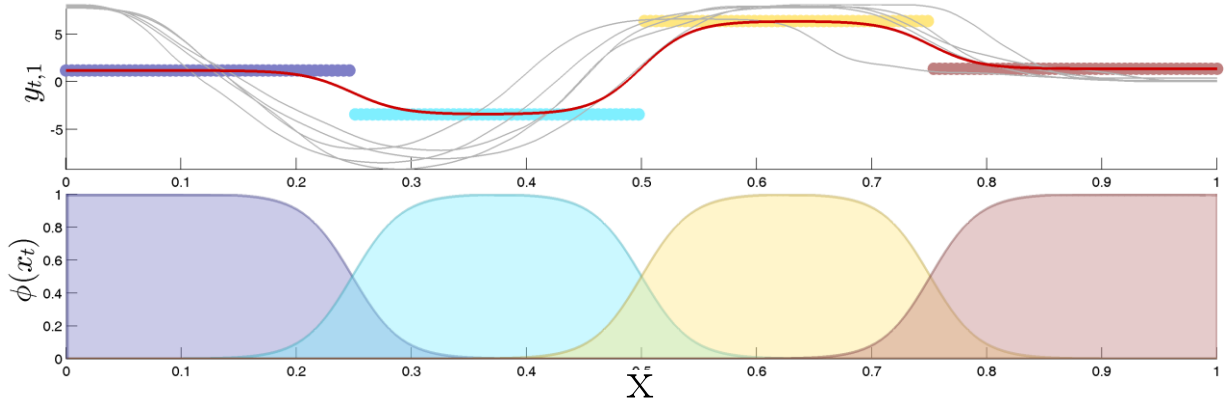
$$\hat{\boldsymbol{A}} = (\boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{Y}$$

LWR can be used for local least squares polynomial fitting by changing the definition of the inputs.

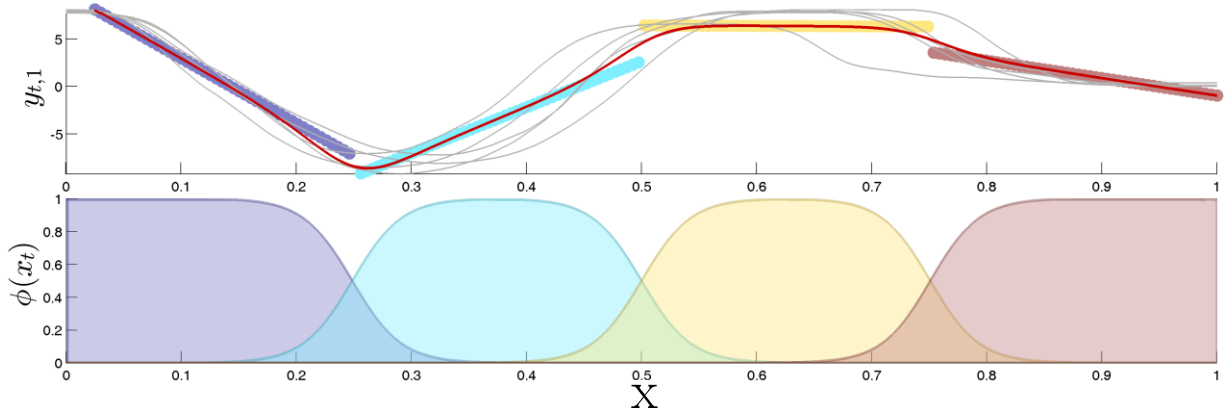
Locally weighted regression (LWR)

$x = 1$

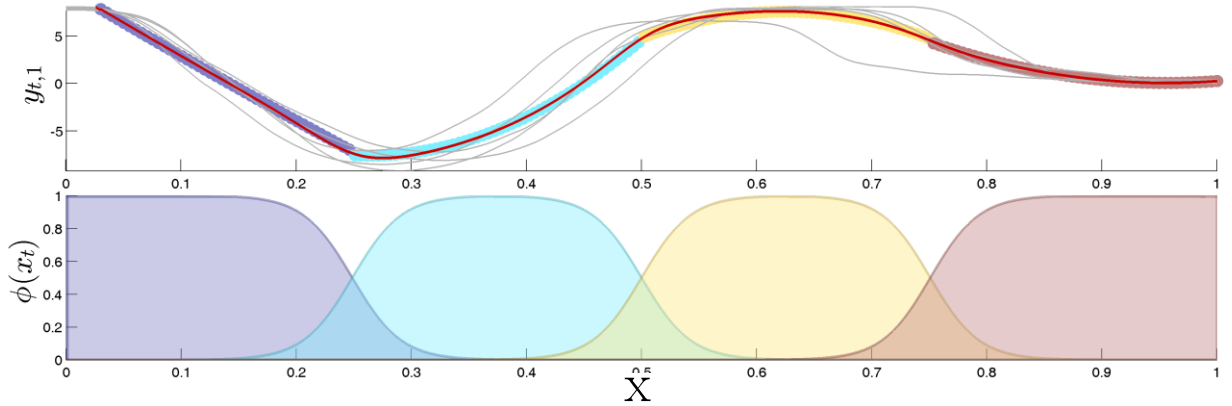




$\boldsymbol{x} = [1, \mathrm{x}]$

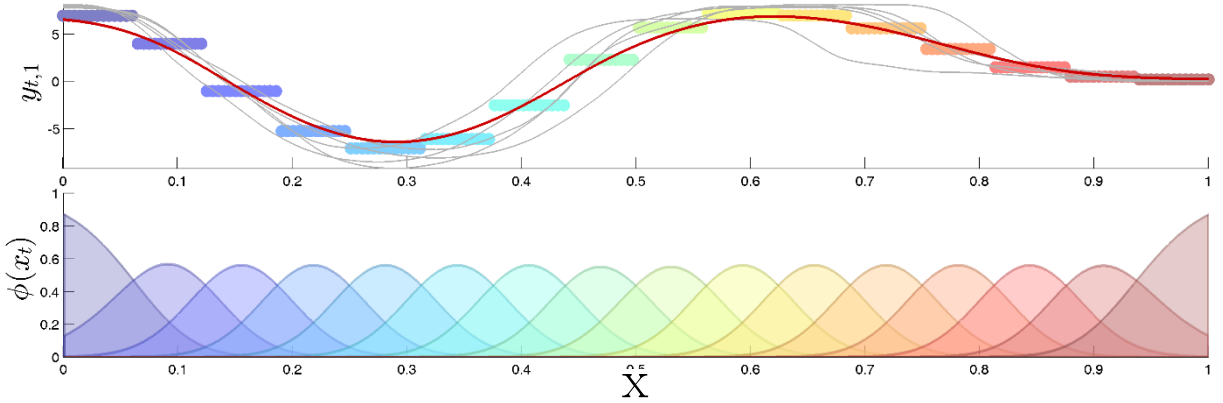


$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$

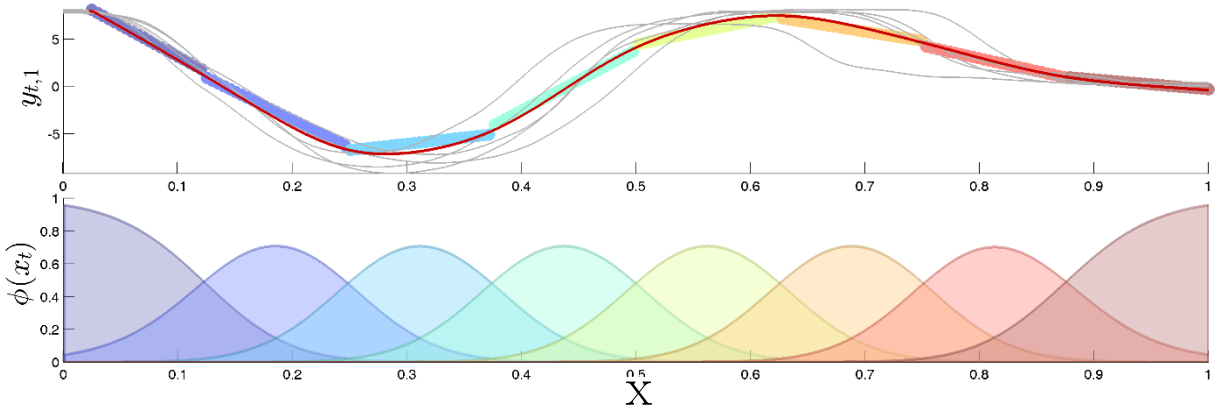


Locally weighted regression (LWR)

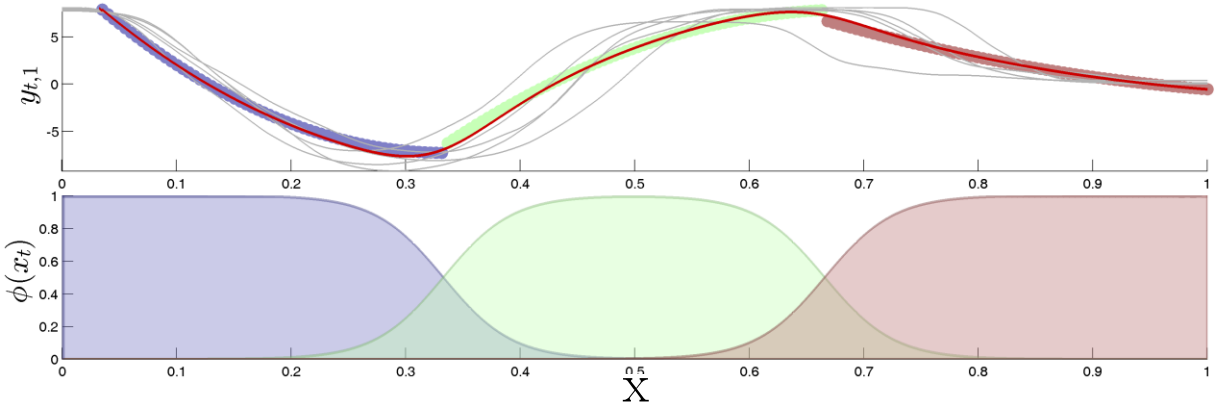
$x = 1$



$\boldsymbol{x} = [1, \mathrm{x}]$



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



Gaussian mixture regression (GMR)

Python notebooks:
demo_GMR.ipynb

Matlab codes:
demo_GMR01.m
demo_GMR_polyFit01.m

Gaussian Mixture Model (GMM)

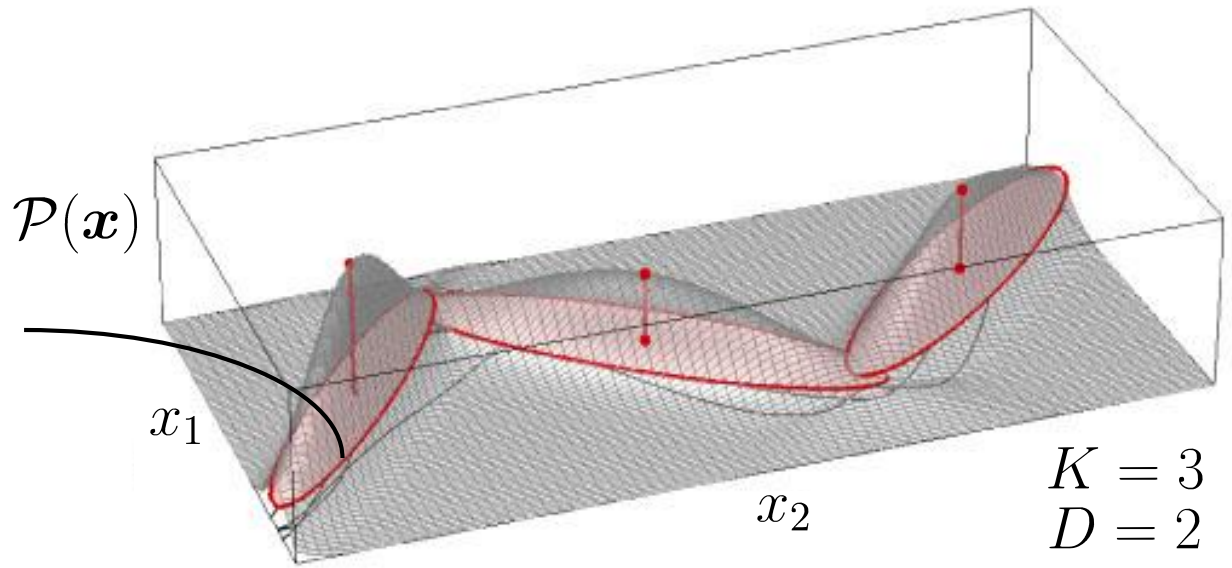
$$\boldsymbol{x}_n \sim \sum_{i=1}^{K} \pi_i \, \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$$

$$\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}_n - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_n - \boldsymbol{\mu}_i)\right)$$

K Gaussians
N datapoints of dimension *D*

$\boldsymbol{x}_n \in \mathbb{R}^D$	Datapoint
$\pi_i \in \mathbb{R}$	Mixing coefficient
$\boldsymbol{\mu}_i \in \mathbb{R}^D$	Center (or mean)
$\boldsymbol{\Sigma}_i \in \mathbb{R}^{D \times D}$	Covariance matrix

Parameters $\boldsymbol{\Theta}^{\text{GMM}} = \{\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^K$



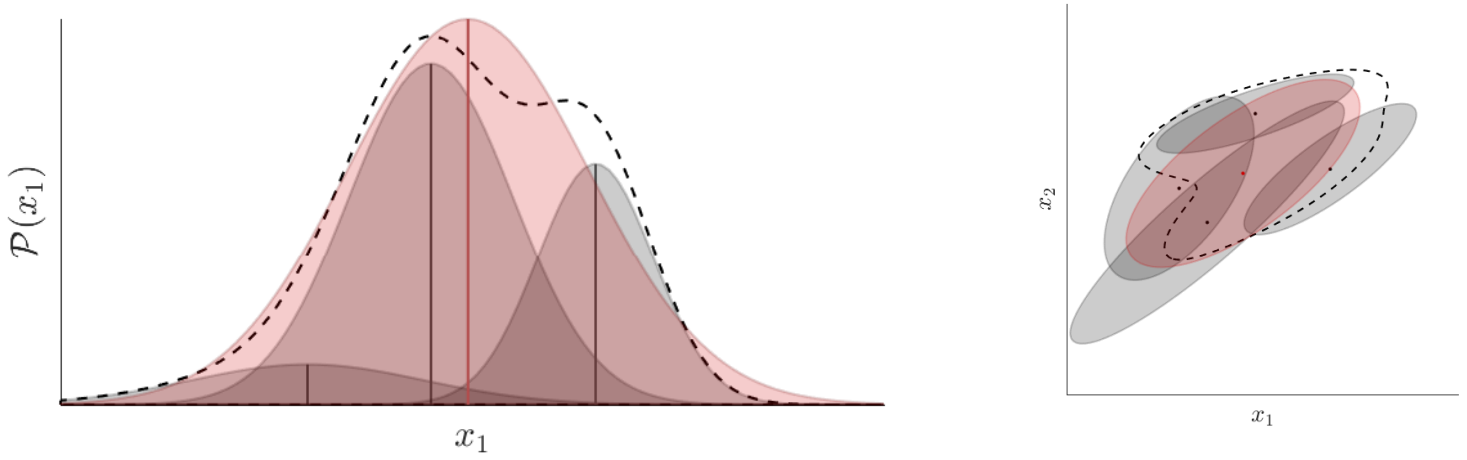
Gaussian estimate of a mixture of Gaussians

We can approximate a mixture of Gaussians $\sum_{i=1}^{K} h_i \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ with a single Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, by **moment matching of the means (first moments) and covariances (second moments)** with

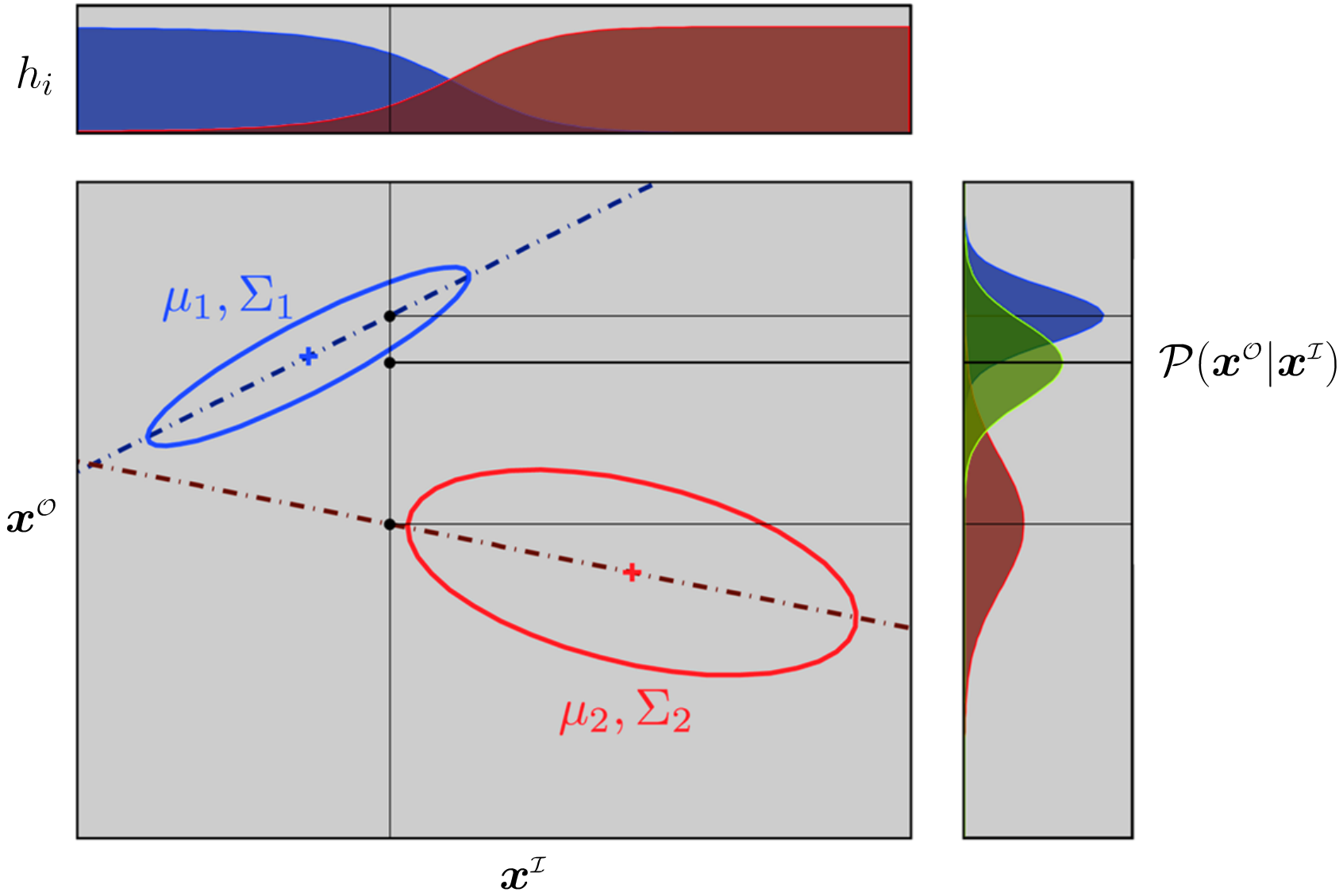
$$\boldsymbol{\mu} = \sum_{i=1}^{K} h_i \, \boldsymbol{\mu}_i,$$

$$\boldsymbol{\Sigma} = \sum_{i=1}^{K} h_i \left(\boldsymbol{\Sigma}_i + \boldsymbol{\mu}_i \boldsymbol{\mu}_i^\top \right) - \boldsymbol{\mu} \boldsymbol{\mu}^\top,$$

also referred to as the **law of total mean and (co)variance**.



Gaussian mixture regression (GMR)



Gaussian mixture regression (GMR)

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

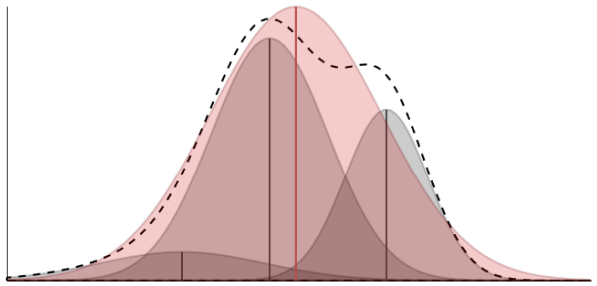
$\mathcal{P}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}})$ can be computed as the multimodal conditional distribution

$$\begin{aligned} \mathcal{P}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}}) &= \sum_{i=1}^{K} h_i \, \mathcal{N}\left(\boldsymbol{x}^{\mathcal{O}} \middle| \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}}\right), \\ \text{with} \quad \hat{\boldsymbol{\mu}}_i^{\mathcal{O}} &= \boldsymbol{\mu}_i^{\mathcal{O}} + \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}_i^{\mathcal{I}-1} (\boldsymbol{x}^{\mathcal{I}} - \boldsymbol{\mu}_i^{\mathcal{I}}), \\ \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} &= \boldsymbol{\Sigma}_i^{\mathcal{O}} - \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}_i^{\mathcal{I}-1} \boldsymbol{\Sigma}_i^{\mathcal{I}\mathcal{O}} \\ \text{and} \quad h_i &= \frac{\pi_i \, \mathcal{N}(\boldsymbol{x}^{\mathcal{I}} | \, \boldsymbol{\mu}_i^{\mathcal{I}}, \boldsymbol{\Sigma}_i^{\mathcal{I}})}{\sum_k^K \pi_k \, \mathcal{N}(\boldsymbol{x}^{\mathcal{I}} | \, \boldsymbol{\mu}_k^{\mathcal{I}}, \boldsymbol{\Sigma}_k^{\mathcal{I}})}, \end{aligned}$$

computed with the marginal

$$\mathcal{N}(\boldsymbol{x}^{\mathcal{I}} | \, \boldsymbol{\mu}_i^{\mathcal{I}}, \boldsymbol{\Sigma}_i^{\mathcal{I}}) = (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}_i^{\mathcal{I}}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}^{\mathcal{I}} - \boldsymbol{\mu}_i^{\mathcal{I}})^{\top} \boldsymbol{\Sigma}_i^{\mathcal{I}-1} (\boldsymbol{x}^{\mathcal{I}} - \boldsymbol{\mu}_i^{\mathcal{I}})\right).$$

Gaussian mixture regression (GMR)



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

$$\begin{aligned} \hat{\boldsymbol{\mu}}_i^{\mathcal{O}} &= \boldsymbol{\mu}_i^{\mathcal{O}} + \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}_i^{\mathcal{I}-1} (\boldsymbol{x}^{\mathcal{I}} - \boldsymbol{\mu}_i^{\mathcal{I}}) \\ \hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} &= \boldsymbol{\Sigma}_i^{\mathcal{O}} - \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}_i^{\mathcal{I}-1} \boldsymbol{\Sigma}_i^{\mathcal{I}\mathcal{O}} \end{aligned}$$

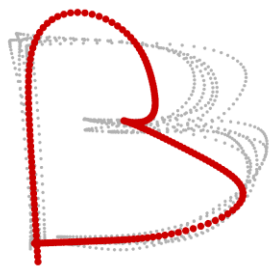
In GMR, an output distribution as a single multivariate Gaussian can be evaluated by moment matching of the means and covariances. The resulting Gaussian distribution $\mathcal{N}(\hat{\boldsymbol{\mu}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}})$ has parameters

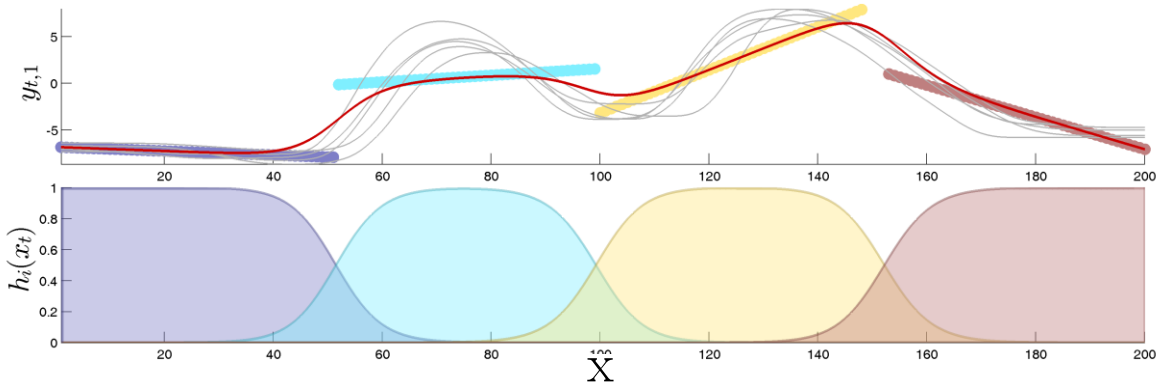
$$\hat{\boldsymbol{\mu}}^{\mathcal{O}} = \sum_{i=1}^{K} h_i \, \hat{\boldsymbol{\mu}}_i^{\mathcal{O}},$$

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

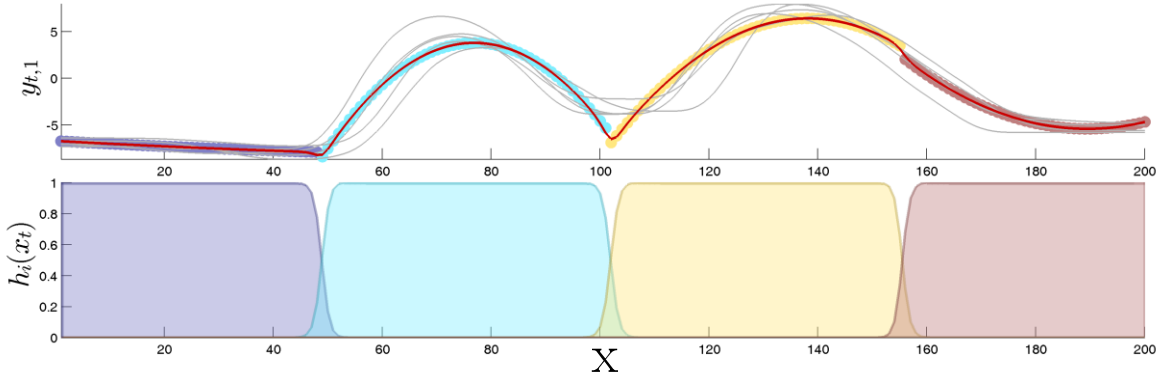
GMR for piecewise polynomial fitting

$x = 1$

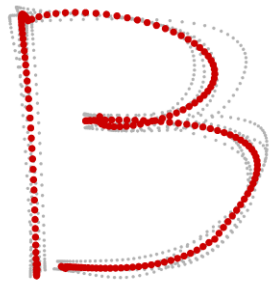


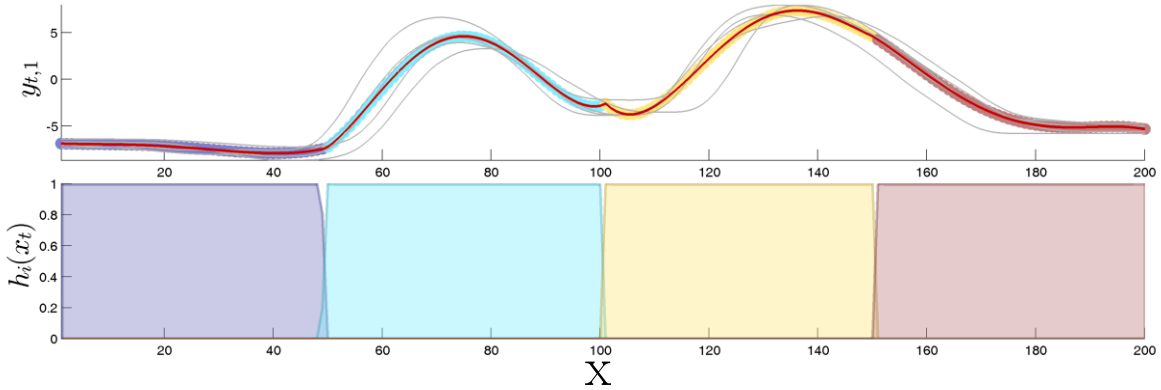


$\boldsymbol{x} = [1, \mathrm{x}]$

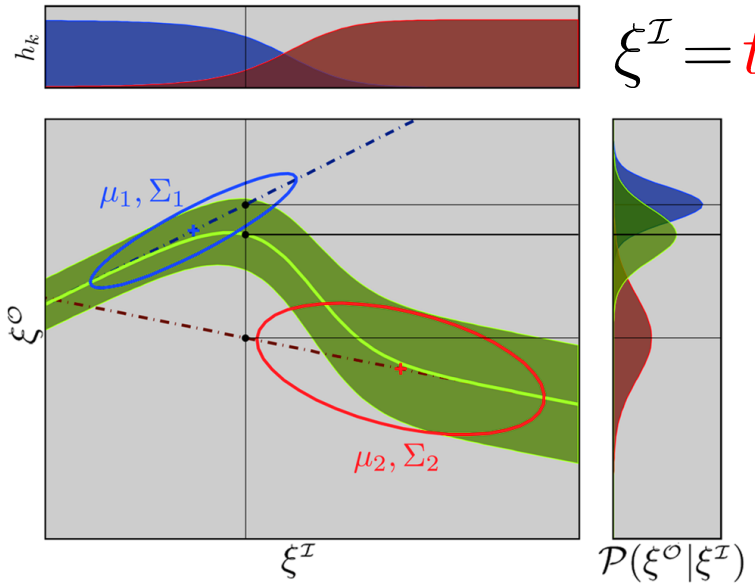


$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$



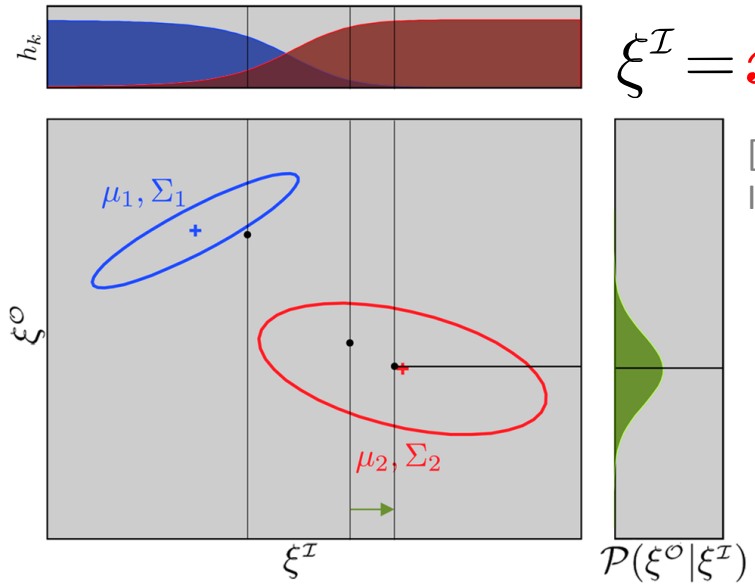


Gaussian mixture regression (GMR)



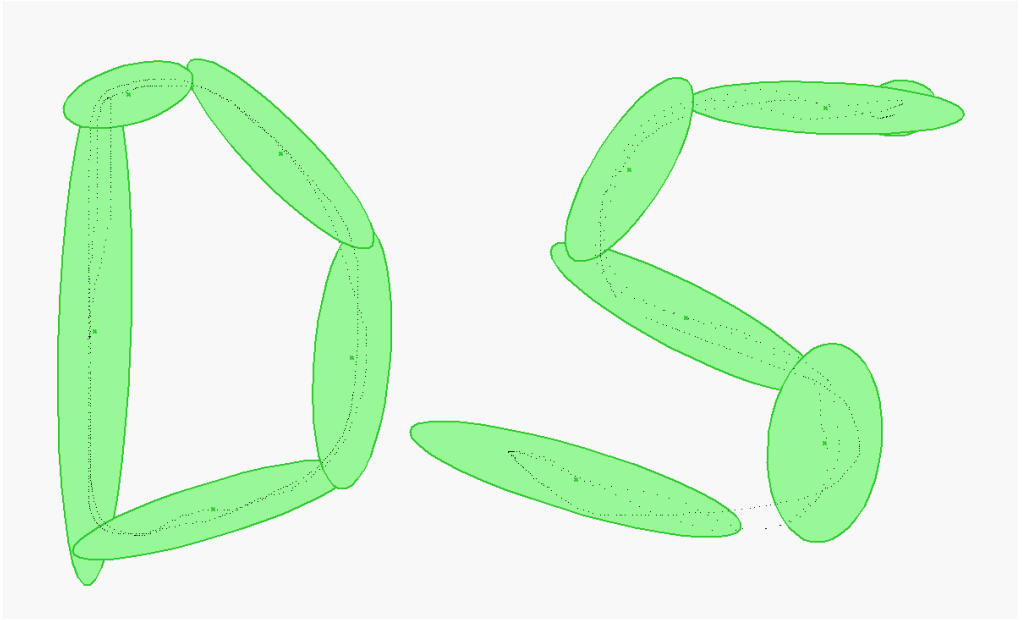
$$\xi^{\mathcal{I}} = t, \;\; \xi^{\mathcal{O}} = \boldsymbol{x}$$

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



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

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



Gaussian process regression (GPR)

Python notebook:
demo_GPR.ipynb

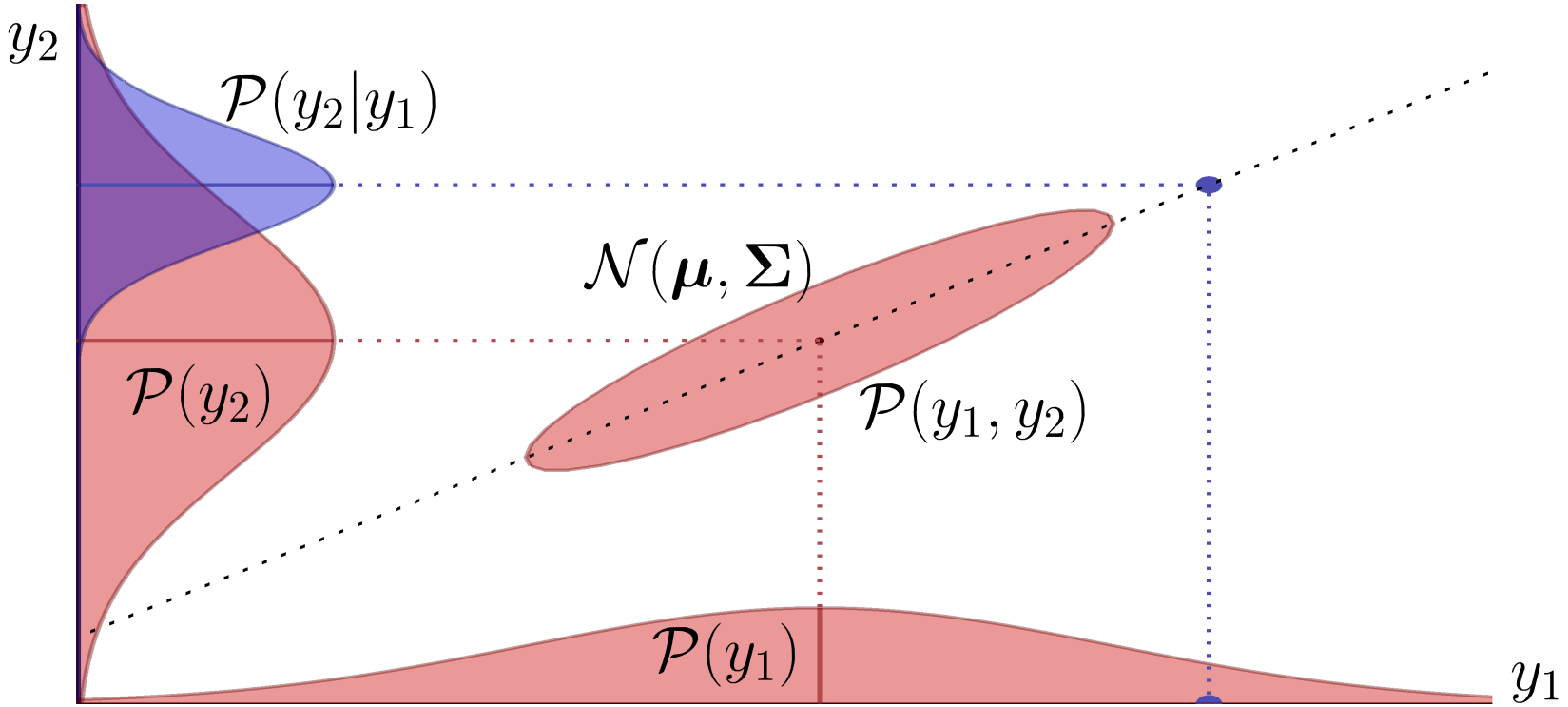
Matlab code:
demo_GPR01.m

Gaussian process – 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.

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

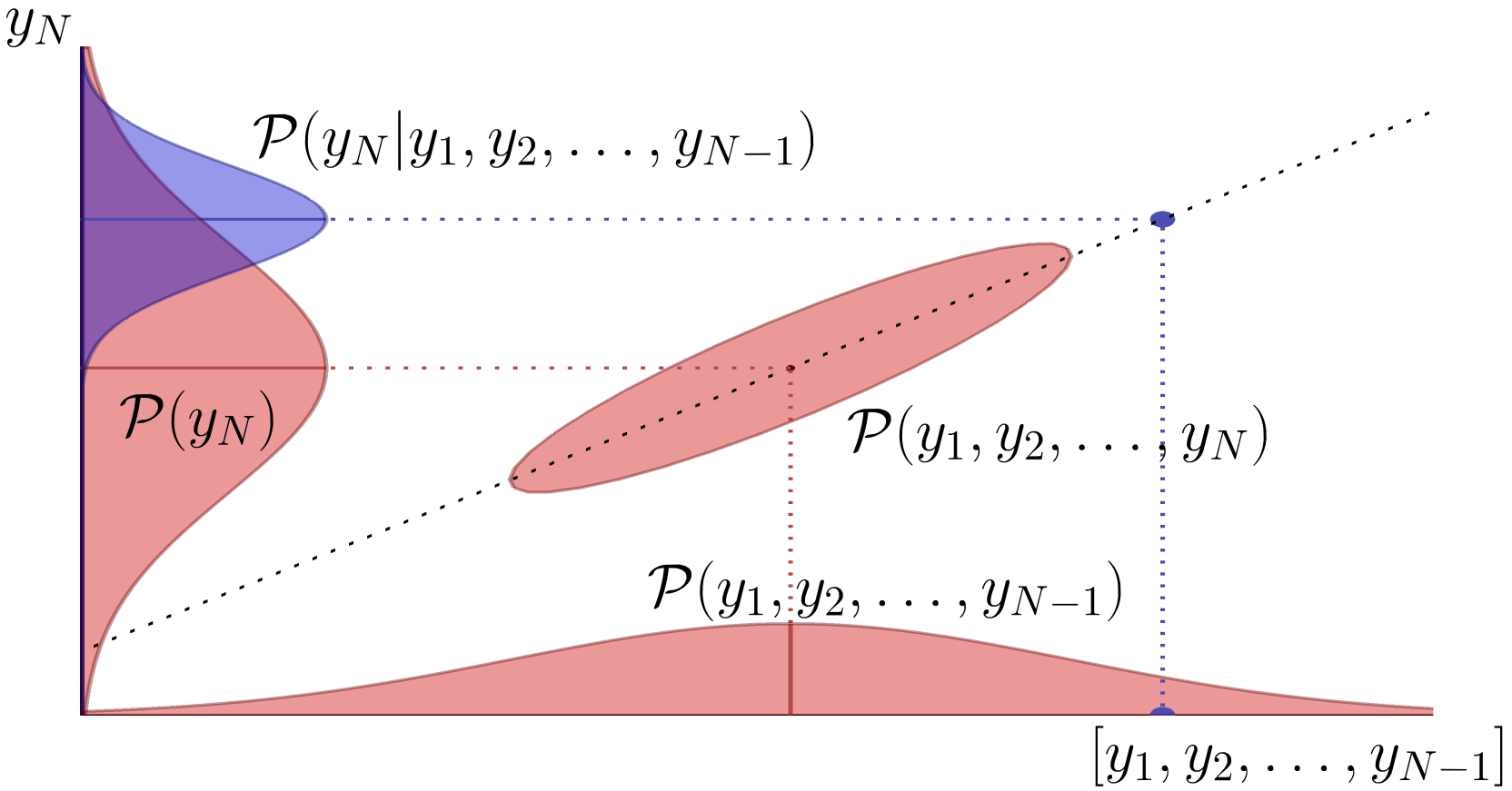
$$\hat{\Sigma}_{22} = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$$



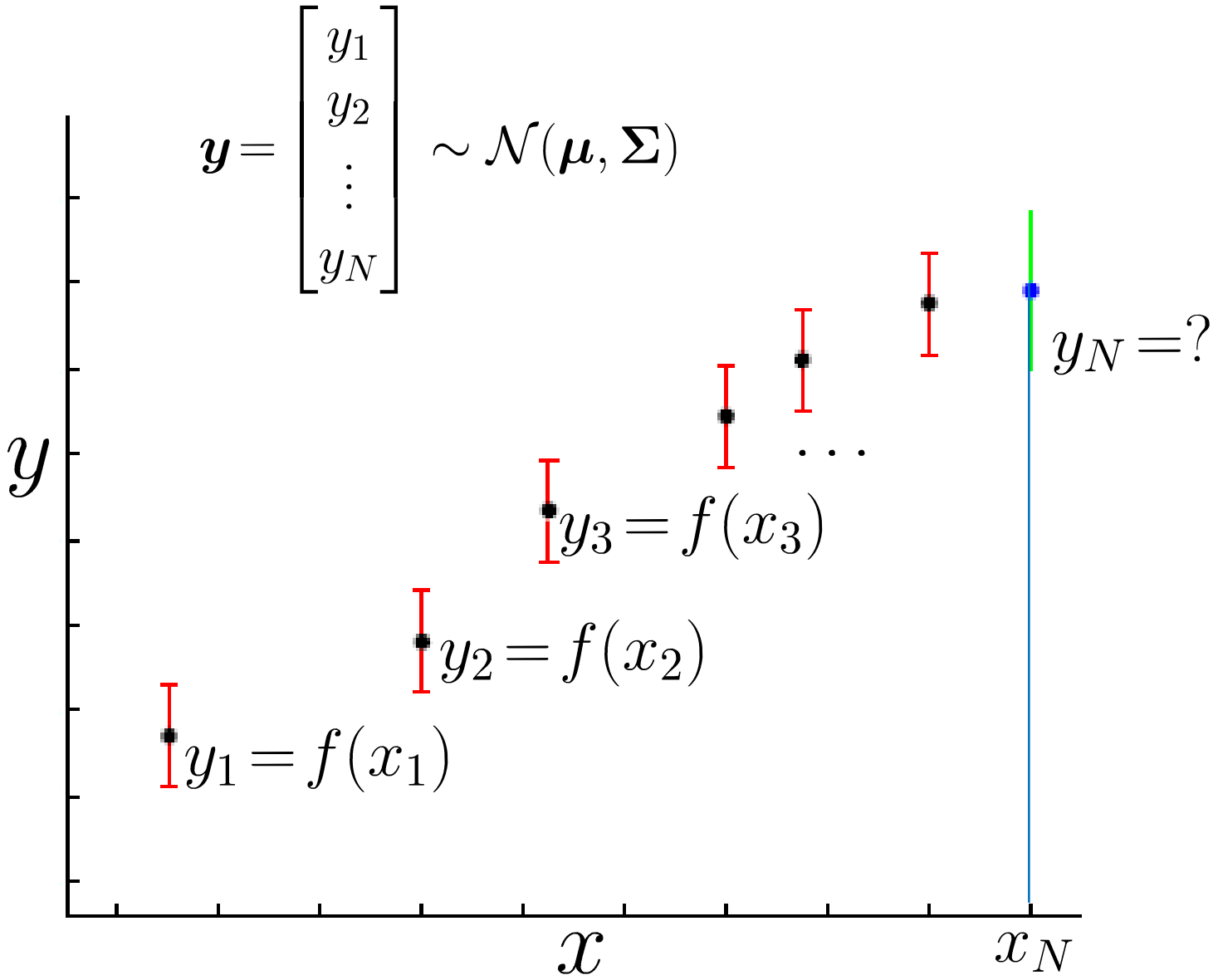
Gaussian process – Interpretation

- 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.

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



Gaussian process - Interpretation

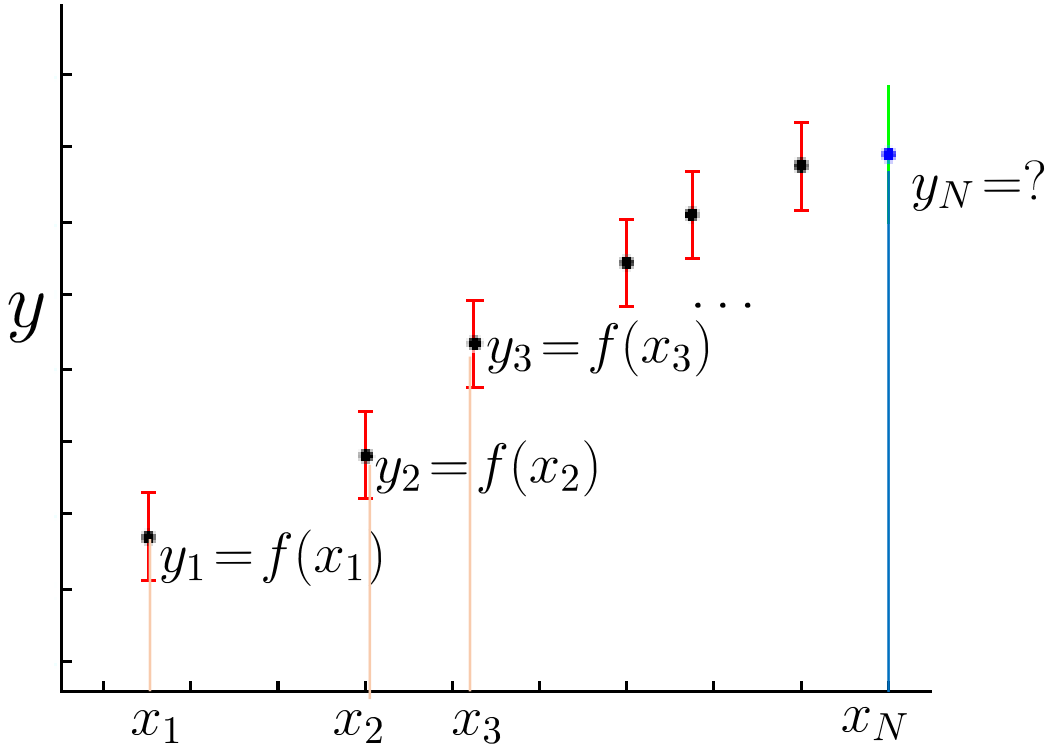


How to construct this joint distribution in GP?

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

Gaussian process – Interpretation

- 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.



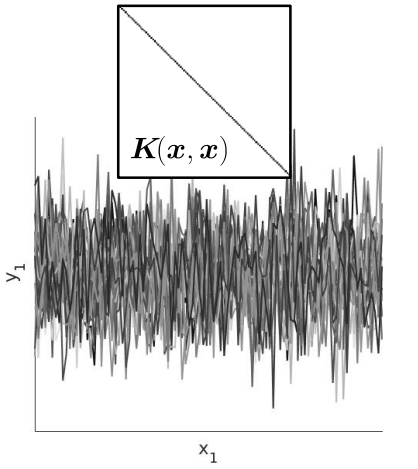
If x_N is similar to x_3, we also expect y_N to be similar to y_3.

Gaussian process (GP)

For a set of input 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

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

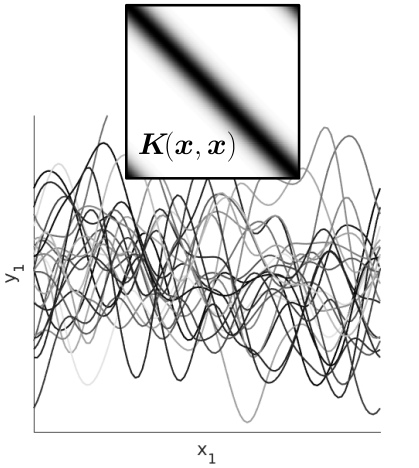
$$\boldsymbol{y} \sim \mathcal{N}\Big(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})\Big)$$



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**.



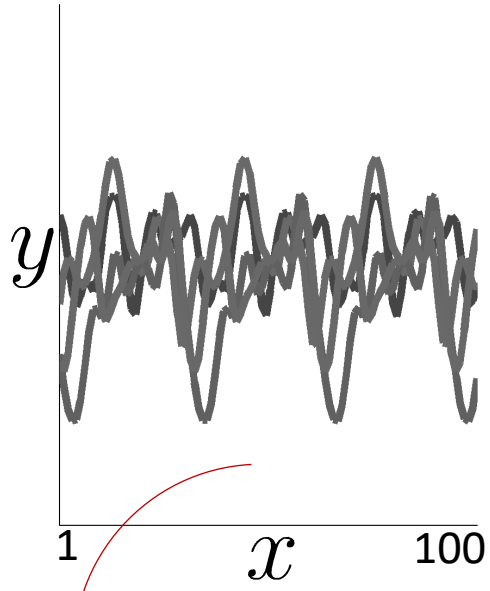
How to choose $k(x_i,x_j)$?

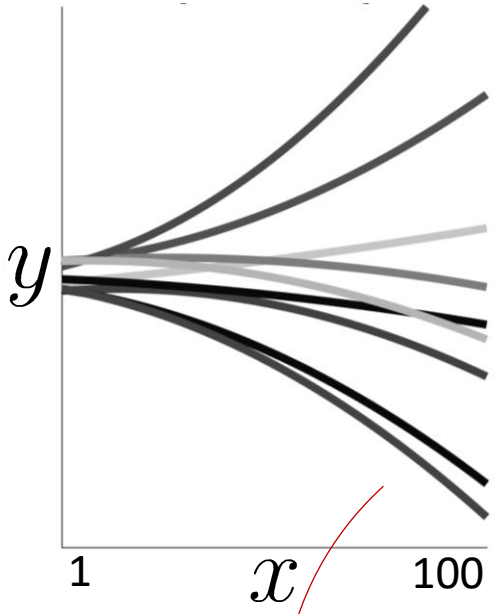
- We may know that our observations are samples from a process that is **smooth**, that is **continuous**, that has **typical amplitude**, or that the variations in the function take place within a **typical dynamic range.**
- These models require hyperparameters to be inferred, but **these hyperparameters 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**.

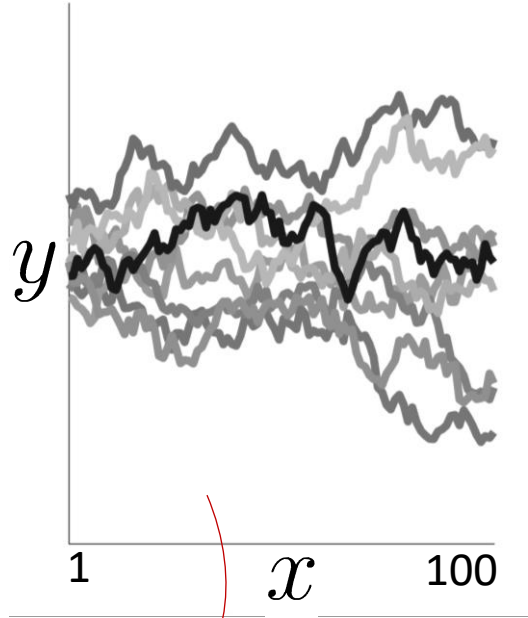
How to choose k(xi,xj) ?

$$\boldsymbol{x} = [1, 2, \ldots, 100]^\top$$

$$\boldsymbol{y} \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})\big)$$







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

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

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

k(xi,xj) as squared exponential function

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**).

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.

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

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

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

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

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

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

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

Gaussian process regression (GPR)

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 computed faster

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

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

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

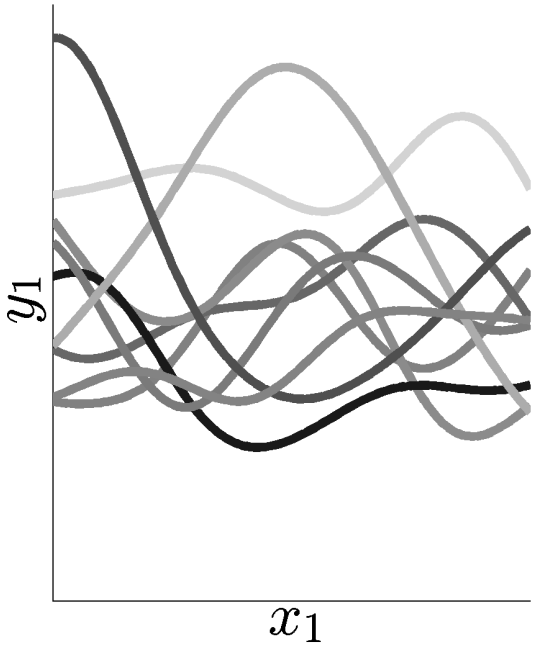
$$\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & k(\boldsymbol{x}_1, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & k(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & k(\boldsymbol{x}_2,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

$k(x_i, x_j)$ as squared exponential function

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

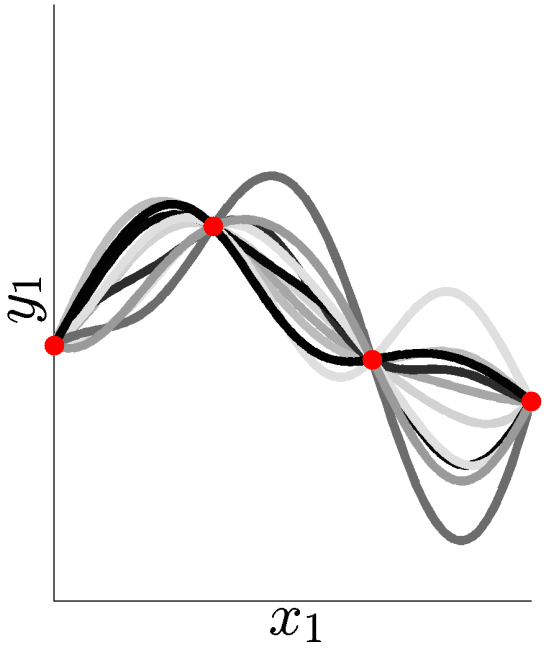
$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$

Samples from prior



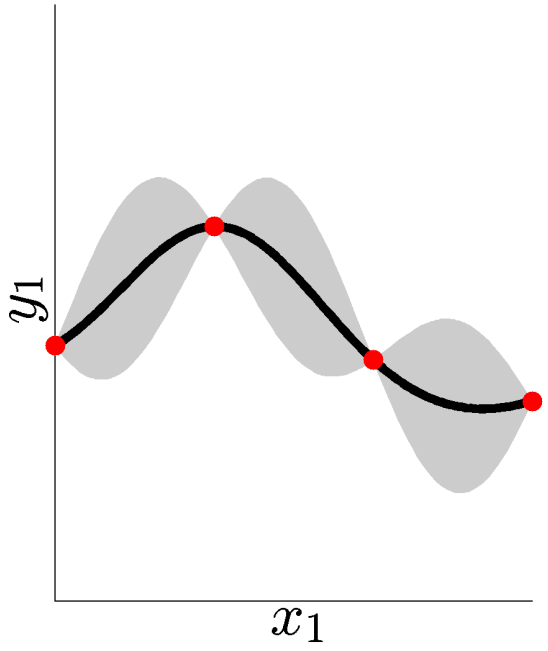
$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Samples from posterior



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

Trajectory distribution



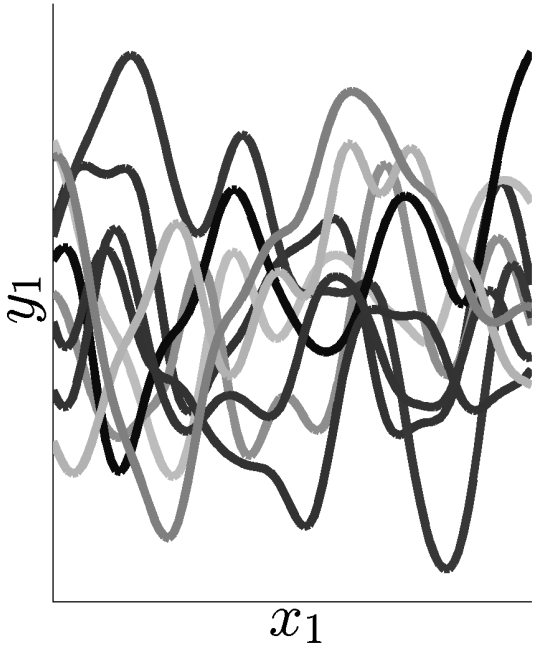
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}}\delta_{i,j}$$

$k(x_i, x_j)$ as squared exponential function

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

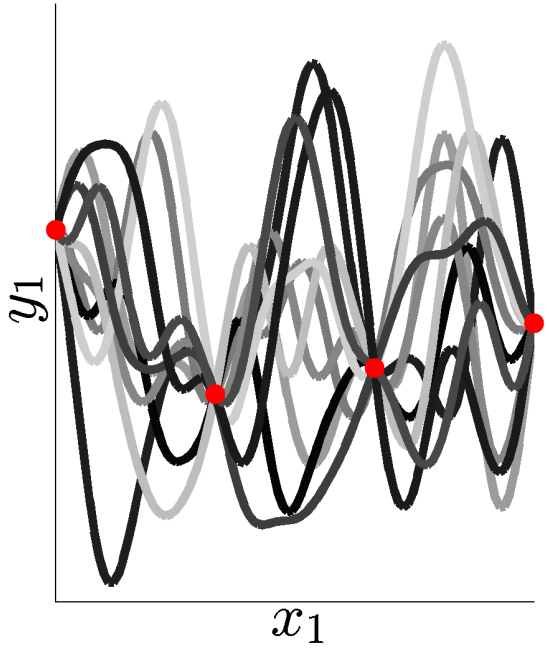
$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$

Samples from prior



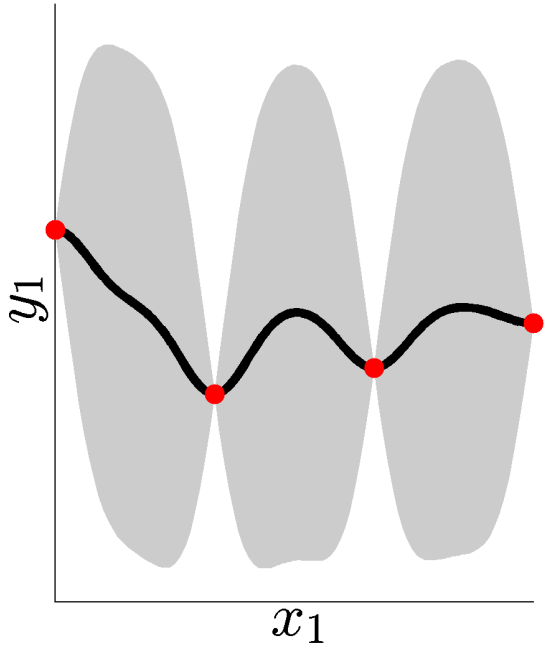
$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Samples from posterior



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

Trajectory distribution

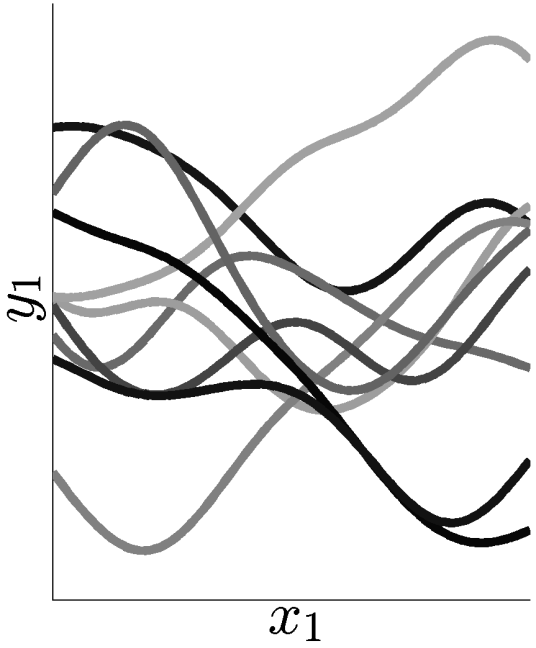


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\text{GP}}\delta_{i,j}$$

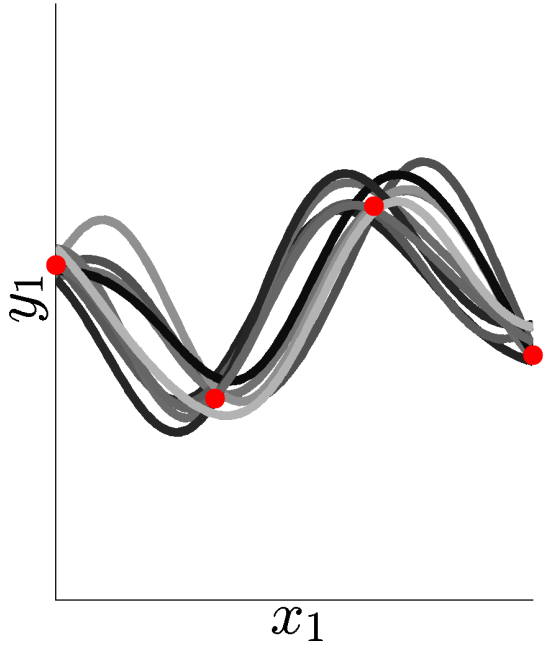
$k(x_i, x_j)$ as squared exponential function

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

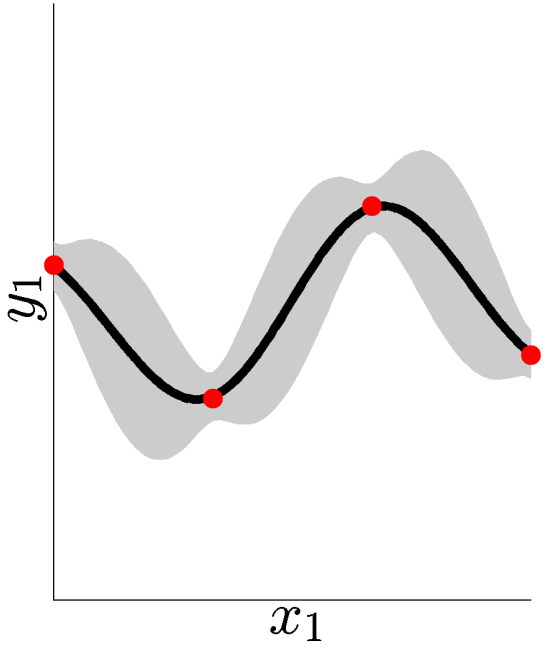
$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$ Samples from prior



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



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

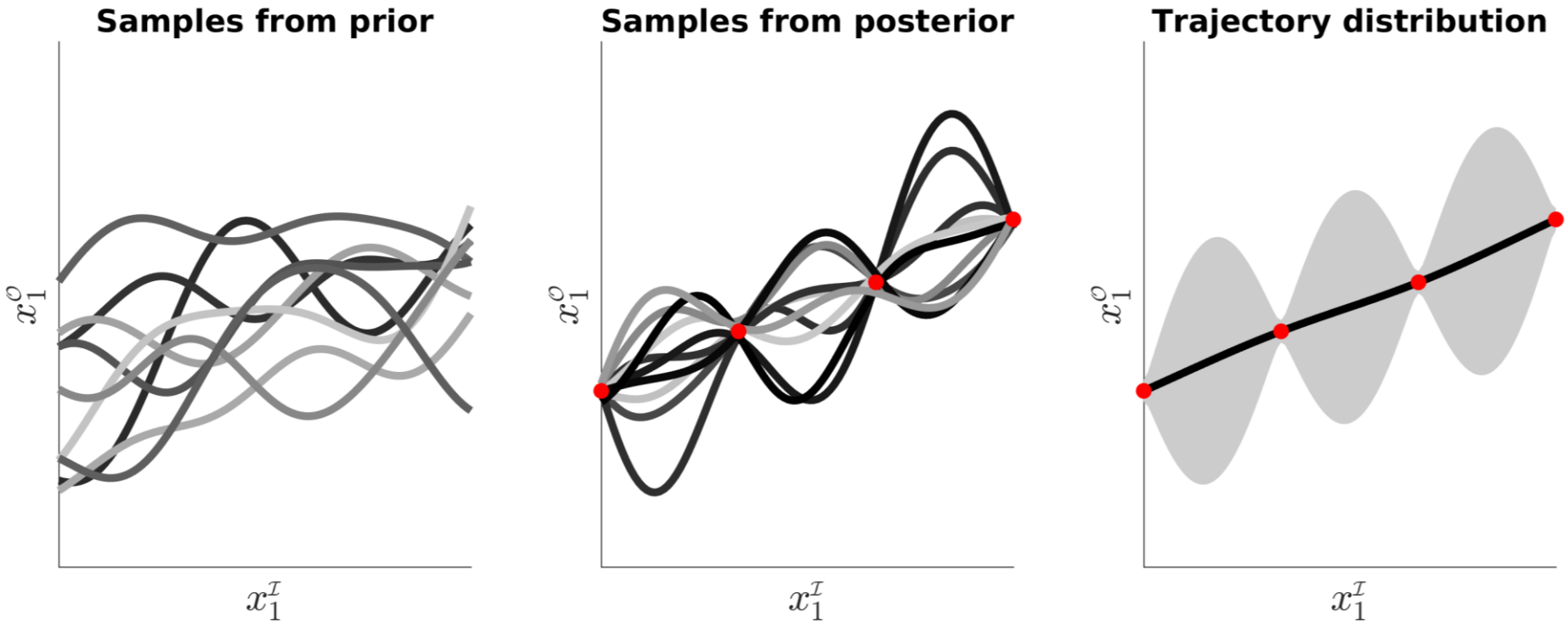


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\text{GP}} \exp\left(-\frac{1}{\Theta_2^{\text{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\text{GP}} \delta_{i,j}$$

$k(x_i, x_j)$ as squared exponential function

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

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



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

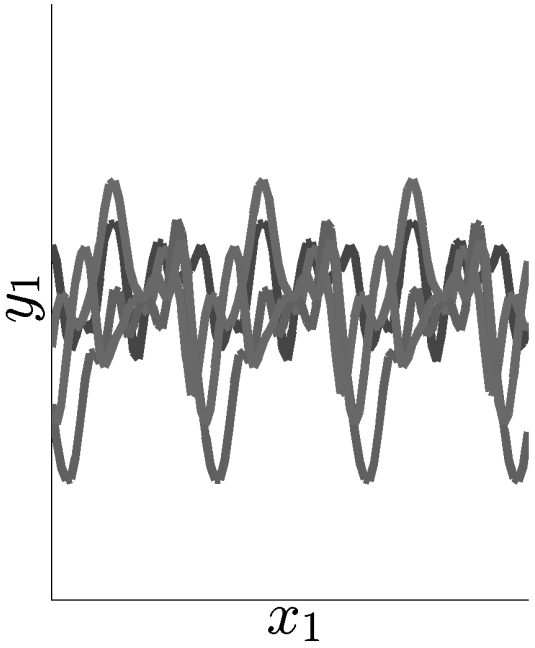
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \exp\left(-\frac{1}{\Theta_2^{\mathrm{GP}}}(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)\right) + \Theta_3^{\mathrm{GP}} \delta_{i,j}$$

k(xi,xj) as periodic covariance function

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

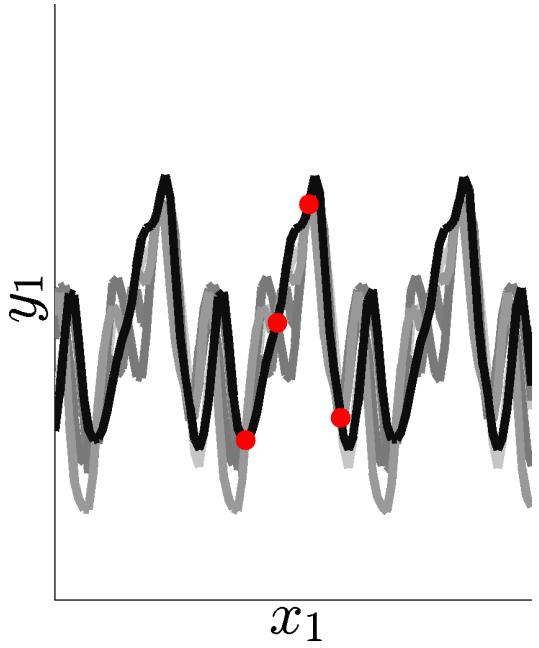
$\boldsymbol{y}^* \sim \mathcal{N}\big(\boldsymbol{\mu}(\boldsymbol{x}^*), \boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)\big)$

Samples from prior



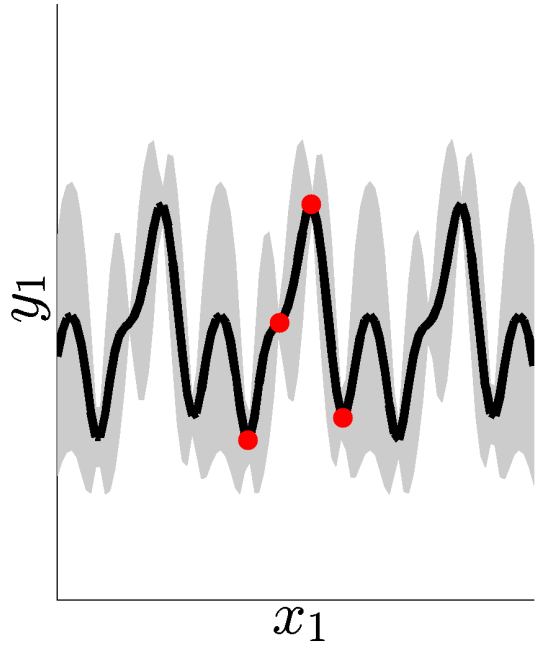
$\boldsymbol{y}^* | \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Samples from posterior

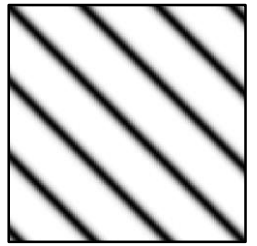


$\mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$

Trajectory distribution



$\boldsymbol{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)$



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

k(xi,xj) as Matérn 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.

$k(x_i, x_j)$ as Matérn 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 $\nu = 3/2$ $(p = 1)$: $C_{3/2}(d) = \sigma^2 \left(1 + \frac{\sqrt{3}d}{\rho}\right) \exp\left(-\frac{\sqrt{3}d}{\rho}\right)$

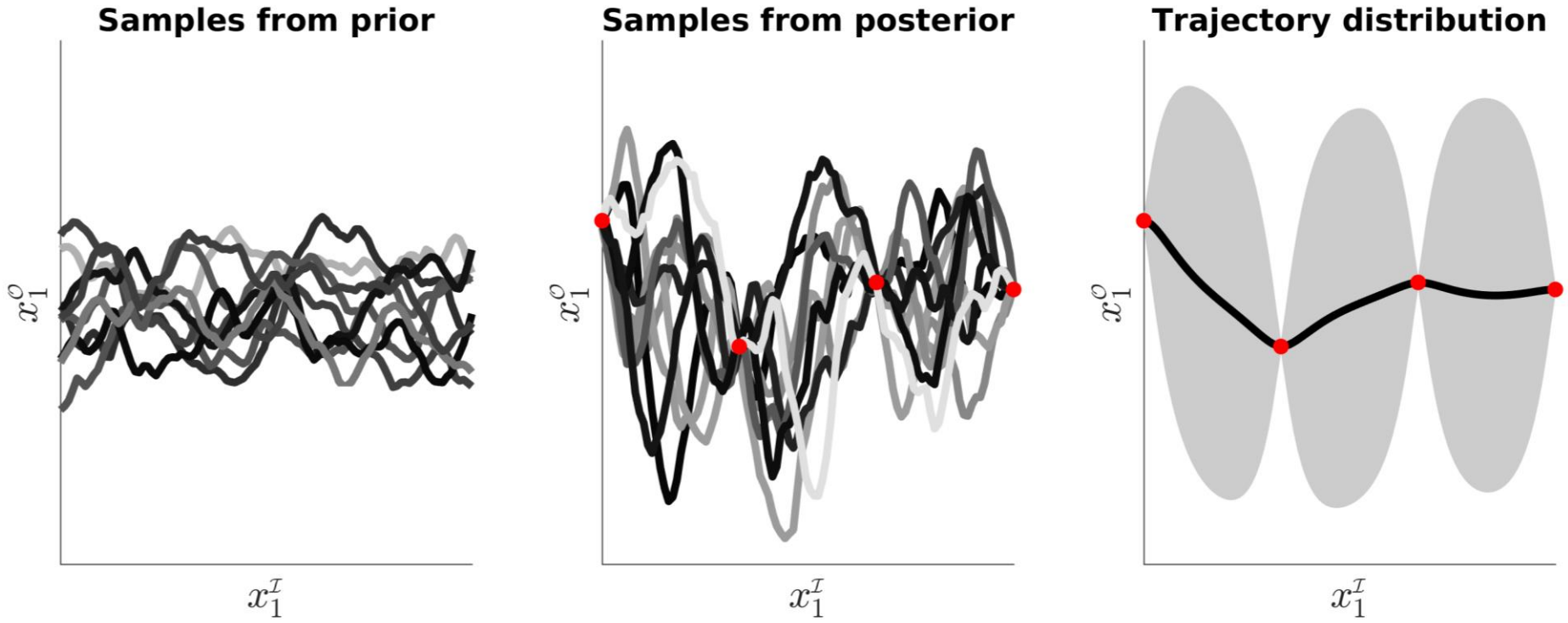
For $\nu = 5/2$ $(p = 2)$: $C_{5/2}(d) = \sigma^2 \left(1 + \frac{\sqrt{5}d}{\rho} + \frac{5d^2}{3\rho^2}\right) \exp\left(-\frac{\sqrt{5}d}{\rho}\right)$

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

k(xi,xj) as Matérn covariance function

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

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

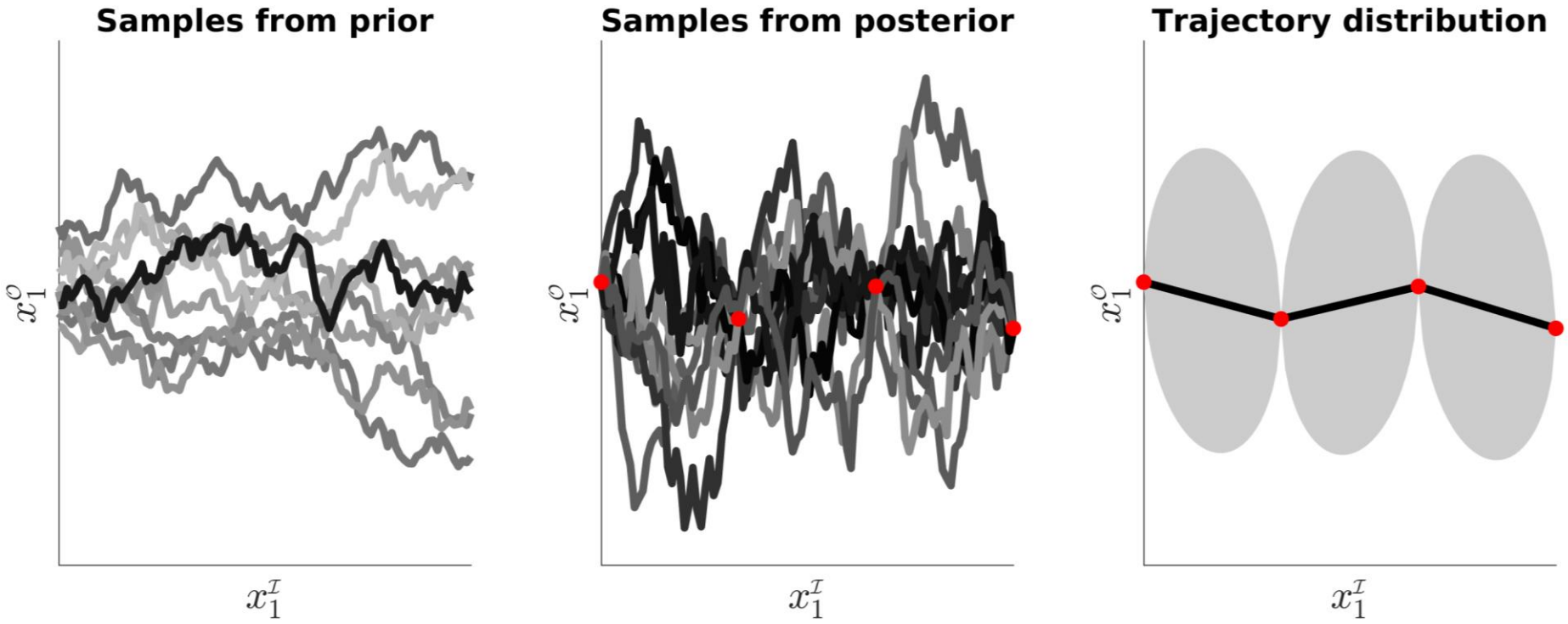


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Theta_1^{\mathrm{GP}} \left(1 + \frac{\sqrt{3}\,\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{\Theta_2^{\mathrm{GP}}}\right) \exp\left(-\frac{\sqrt{3}\,\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{\Theta_2^{\mathrm{GP}}}\right)$$

$k(x_i,x_j)$ as Brownian motion covariance function

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

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

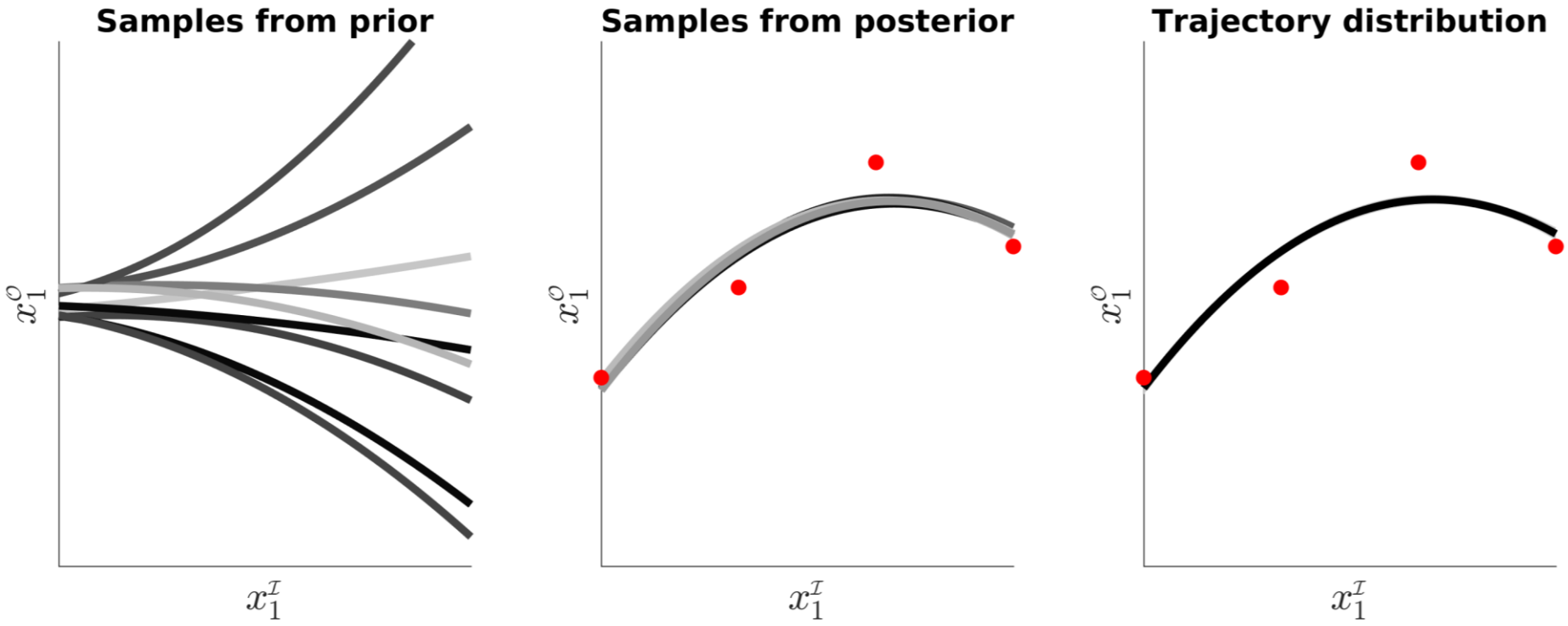


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \min(\boldsymbol{x}_i, \boldsymbol{x}_j) + \Theta_1^{\mathrm{GP}} \qquad \Theta_1^{\mathrm{GP}} = 0.1$$

$k(x_i,x_j)$ as quadratic covariance function

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

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

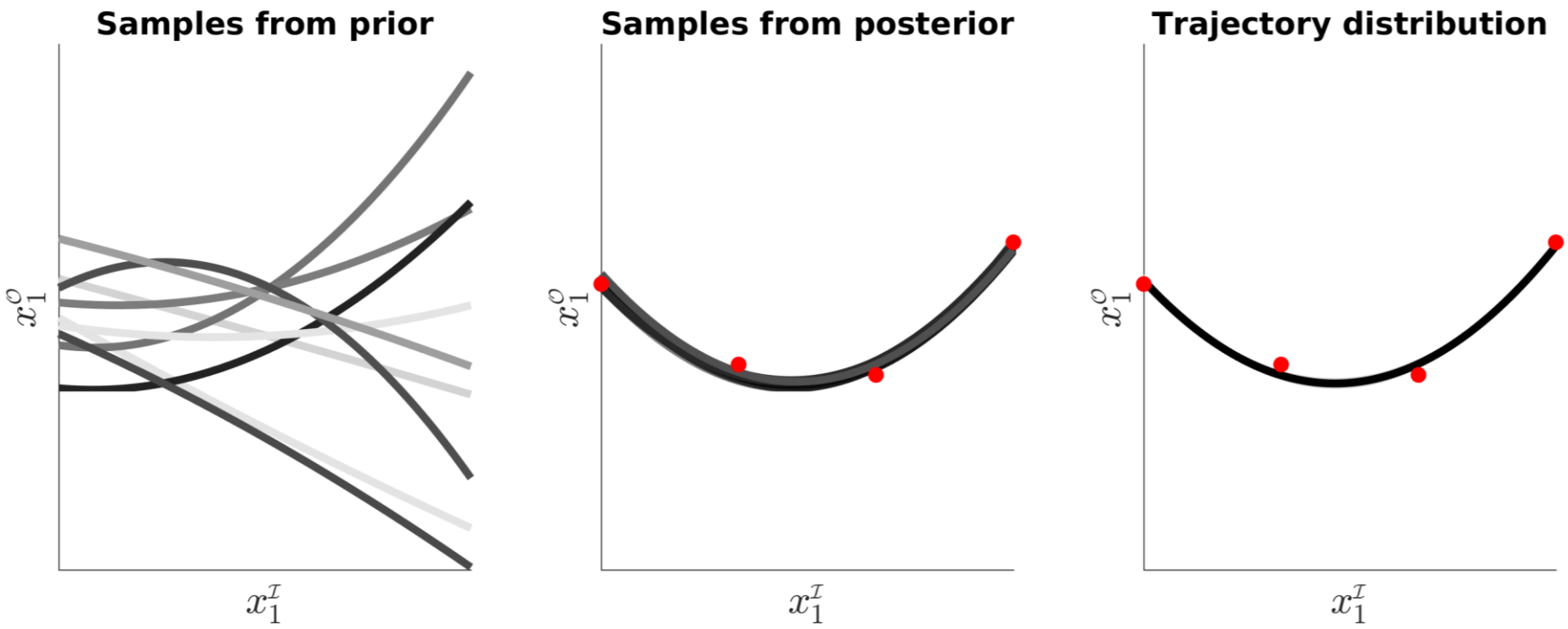


$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j + \Theta_1^{\text{GP}})^2 \qquad \Theta_1^{\text{GP}} = 0.1$$

$k(x_i, x_j)$ as polynomial covariance function

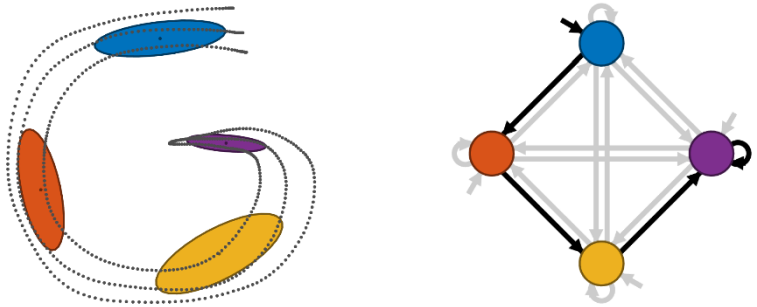
$$\Theta_1^{\mathrm{GP}} = 0.1$$

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



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

$k(x_i,x_j)$ 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 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).

$k(x_i, x_j)$ as weighted sum of covariance 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

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

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

Some extensions of Gaussian processes

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

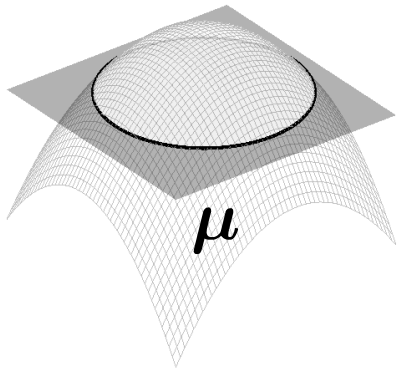
- **Sparse GP:**
  A known bottleneck in Gaussian process prediction is that the computational complexity of prediction is $O(N^3)$
  → not feasible for large data sets!

  Sparse Gaussian processes circumvent this issue by building a representative set for the given process $y = f(\mathbf{x})$.

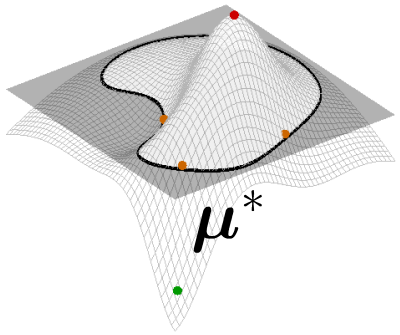
- **Gaussian process latent variable models (GPLVM):**
  GPLVM is a probabilistic dimensionality reduction method that uses GPs to find a lower dimensional non-linear embedding of high dimensional data.

Example: Gaussian Process Implicit Surface (GPIS)

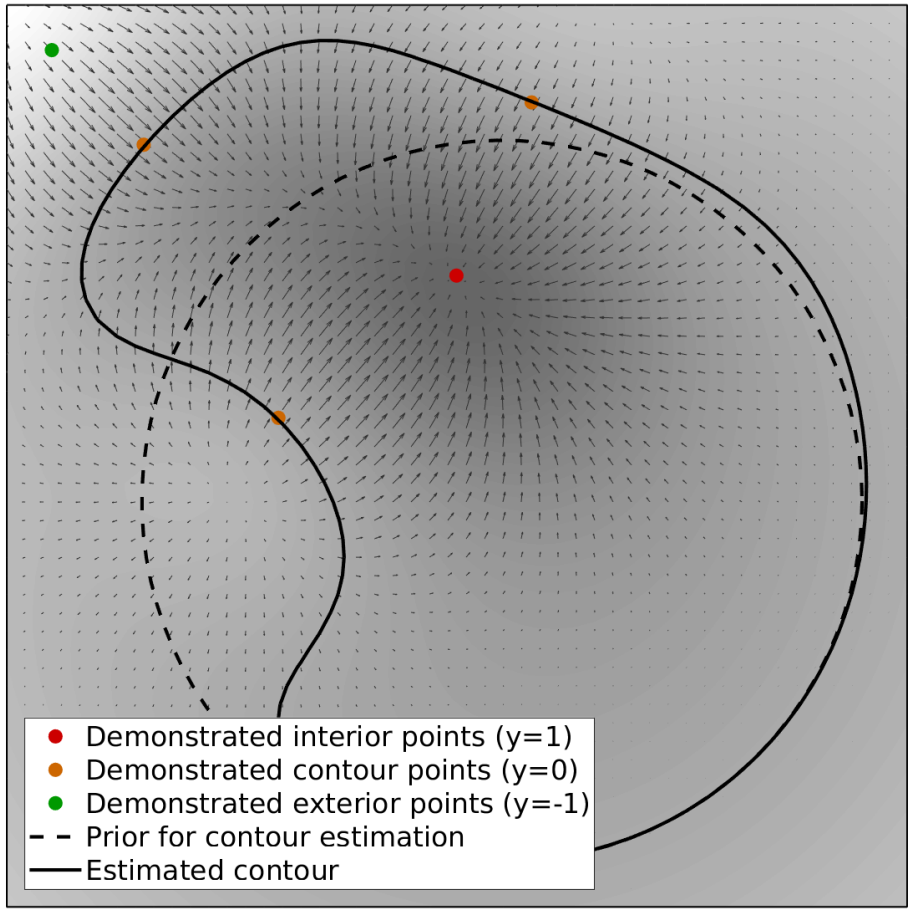
Prior (circular contour)



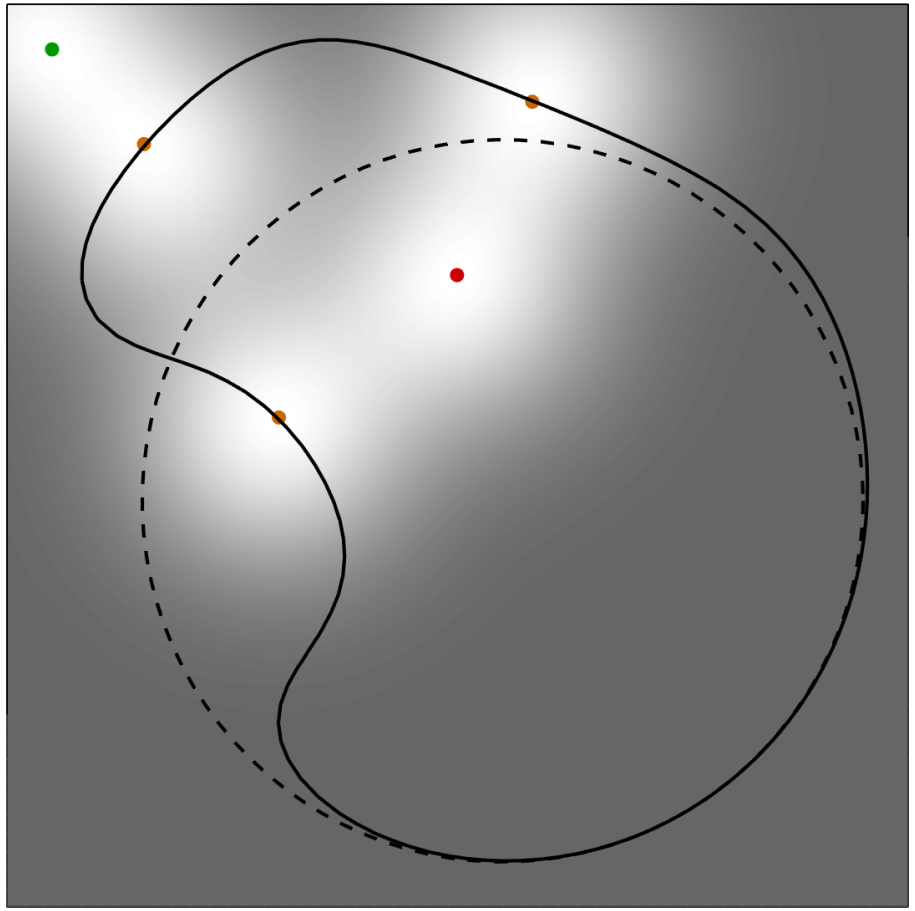
Posterior



Distance to contour and gradient



Uncertainty



Summary of today's lecture

Properties of multivariate Gaussian distributions:

- Product of Gaussians
- Linear transformation and combination
- Conditional distribution

Three nonlinear regression models:

- Locally weighted regression (LWR)
- Gaussian mixture regression (GMR)
- Gaussian process regression (GPR)

Modeling possible co-variations
(a.k.a. aleatoric uncertainty)

Modeling uncertainty of the estimate
(a.k.a. epistemic uncertainty)

References

LWR

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

W.S. Cleveland. Robust locally weighted regression and smoothing scatterplots. American Statistical Association 74(368):829–836, 1979

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

S. Calinon. Mixture models for the analysis, edition, and synthesis of continuous time series. Mixture Models and Applications, Springer, 2019

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

GPIS

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