EE613
Machine Learning for Engineers

NONLINEAR REGRESSION I

Sylvain Calinon
Robot Learning & Interaction Group
Idiap Research Institute
Dec. 12, 2019

Outline

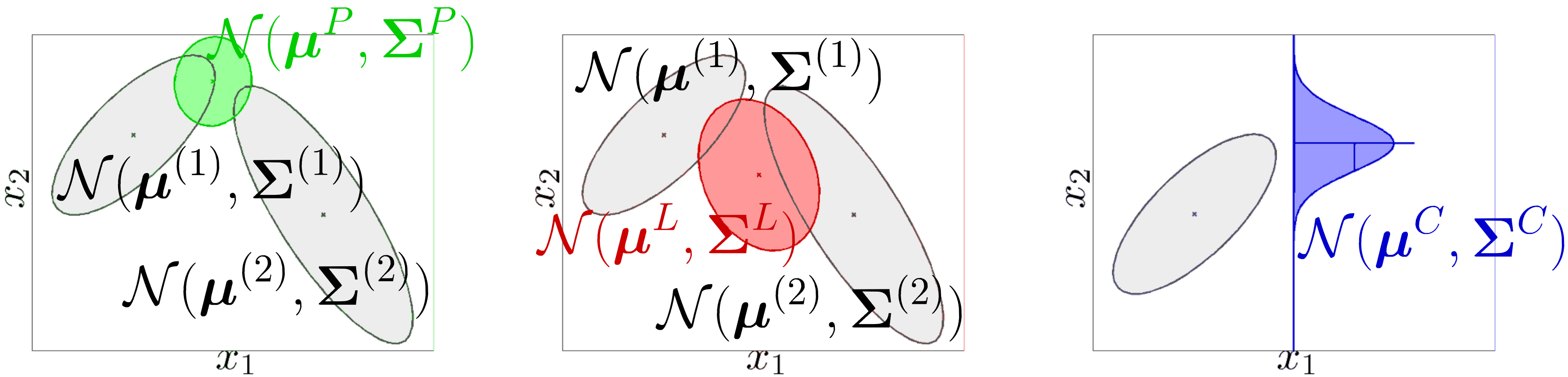
- Properties of multivariate Gaussian distributions:
  - Product of Gaussians
  - Linear transformation and combination
  - Conditional distribution
  - Gaussian estimate of a mixture of Gaussians

- Locally weighted regression (LWR)

- Gaussian mixture regression (GMR)

- Example of application:
  Dynamical movement primitives (DMP)

Some very useful properties...



Product of Gaussians:

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

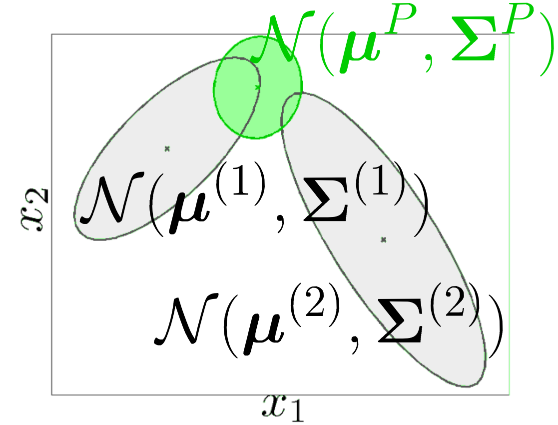
Linear transformation and combination:

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

Conditional distribution:

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

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

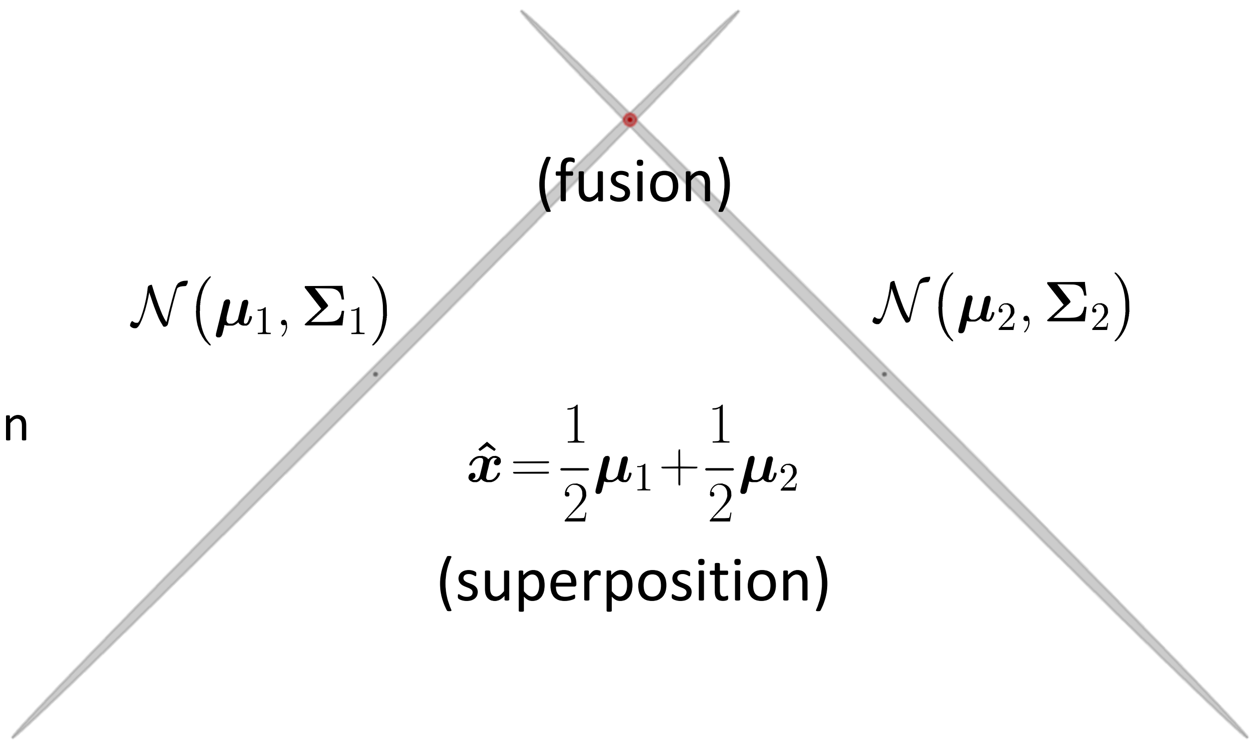
$$c\,\mathcal{N}(\boldsymbol{\mu}^P, \boldsymbol{\Sigma}^P) = \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}) \cdot \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)}),$$

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

Product of Gaussians - Motivating example

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\|\boldsymbol{\mu}_1 - \boldsymbol{x}\right\|^2_{\boldsymbol{W}_1} + \left\|\boldsymbol{\mu}_2 - \boldsymbol{x}\right\|^2_{\boldsymbol{W}_2}$$
$$= (\boldsymbol{W}_1 + \boldsymbol{W}_2)^{-1} (\boldsymbol{W}_1\boldsymbol{\mu}_1 + \boldsymbol{W}_2\boldsymbol{\mu}_2)$$
$$= \left(\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1}\right)^{-1} \left(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2\right)$$

Product of Gaussians

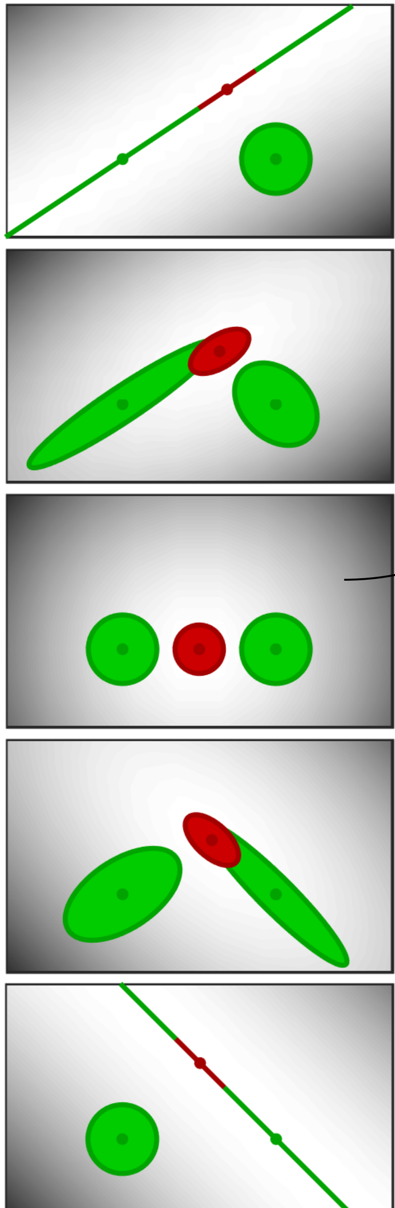


$\boldsymbol{\mu}_i$ center of the Gaussian

$\boldsymbol{\Sigma}_i$ covariance matrix

$\boldsymbol{W}_i$ precision matrix
$(\boldsymbol{W}_i = \boldsymbol{\Sigma}_i^{-1})$

Product of Gaussians - Fusion of information



$$\mathcal{N}\big(\boldsymbol{\mu}, \boldsymbol{\Sigma}\big) \propto \mathcal{N}\big(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}\big)\, \mathcal{N}\big(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)}\big)$$

Scalar superposition

Using **full weight matrices** also include the special case of using **scalar weights**

Product of Gaussians - Kalman filter

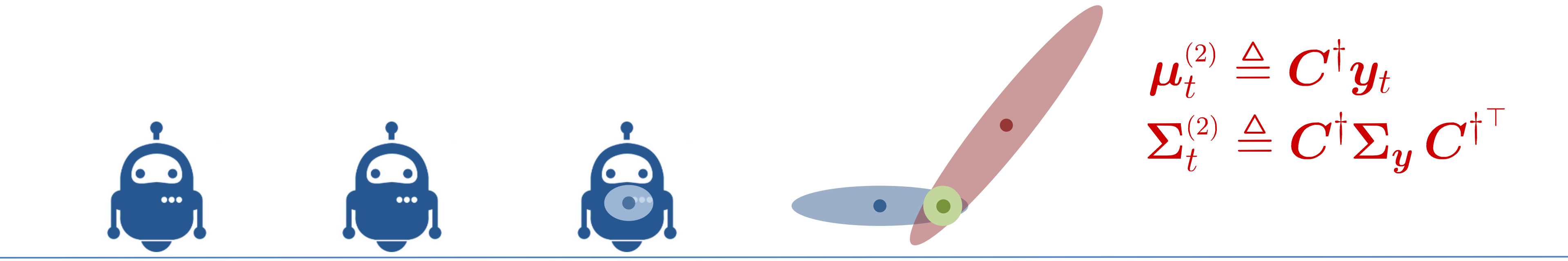
$$\boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{e}_{\boldsymbol{y}}$$

$$\boldsymbol{e}_{\boldsymbol{y}} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{y}}\right)$$

Kalman filter as product of Gaussians

$$\boldsymbol{\Sigma}_t = \left(\boldsymbol{\Sigma}_t^{(1)^{-1}} + \boldsymbol{\Sigma}_t^{(2)^{-1}}\right)^{-1}$$

$$\boldsymbol{\mu}_t = \boldsymbol{\Sigma}_t \left(\boldsymbol{\Sigma}_t^{(1)^{-1}} \boldsymbol{\mu}_t^{(1)} + \boldsymbol{\Sigma}_t^{(2)^{-1}} \boldsymbol{\mu}_t^{(2)}\right)$$



t=0 t=1 t=2

$$\boldsymbol{\mu}_t^{(2)} \triangleq \boldsymbol{C}^{\dagger}\boldsymbol{y}_t$$

$$\boldsymbol{\Sigma}_t^{(2)} \triangleq \boldsymbol{C}^{\dagger}\boldsymbol{\Sigma}_{\boldsymbol{y}}\,\boldsymbol{C}^{\dagger^{\top}}$$

$$\boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{B}\boldsymbol{u}_t + \boldsymbol{e}_{\boldsymbol{x}}$$

$$\boldsymbol{e}_{\boldsymbol{x}} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{x}}\right)$$

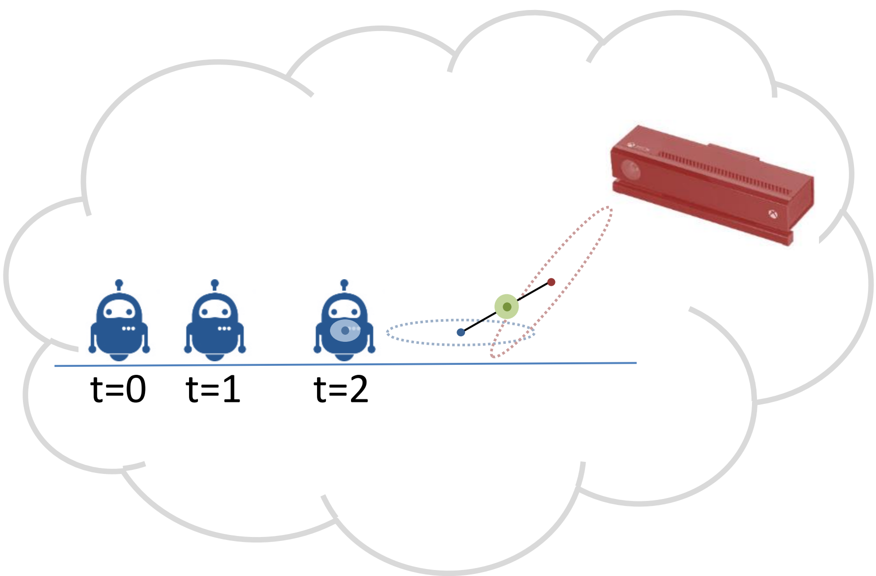
$$\boldsymbol{\mu}_t^{(1)} \triangleq \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{B}\boldsymbol{u}_t$$

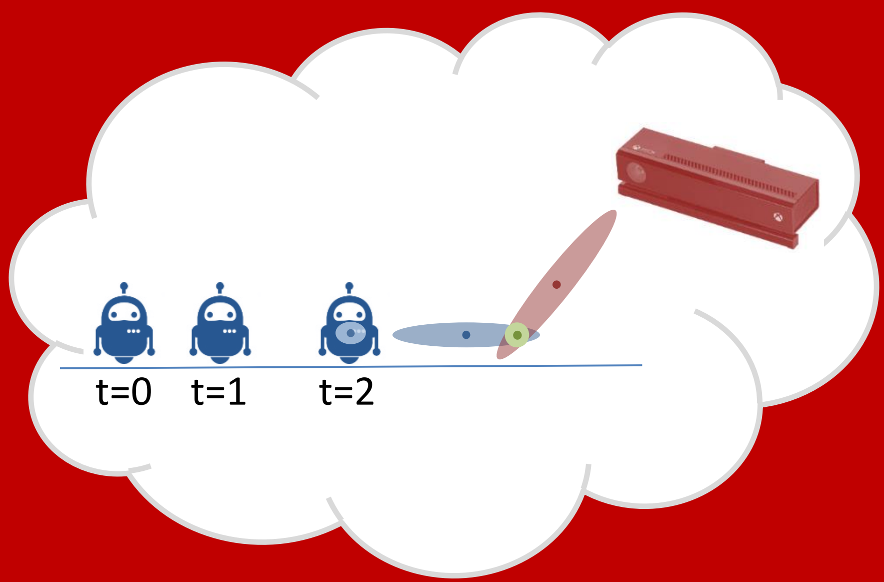
$$\boldsymbol{\Sigma}_t^{(1)} \triangleq \boldsymbol{A}\boldsymbol{\Sigma}_{t-1}\boldsymbol{A}^{\top} + \boldsymbol{\Sigma}_{\boldsymbol{x}}$$

Superposition

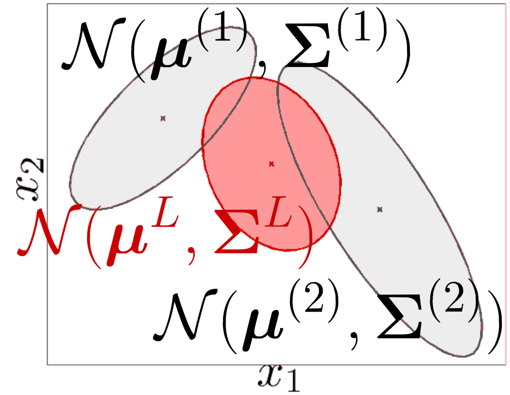
VS

Fusion





Linear transformation and combination

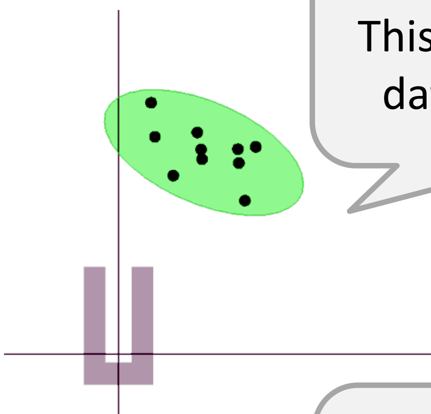


If $\boldsymbol{x}^{(1)} \sim \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ and $\boldsymbol{x}^{(2)} \sim \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$, the linear transformation $\boldsymbol{A}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{x}^{(2)}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

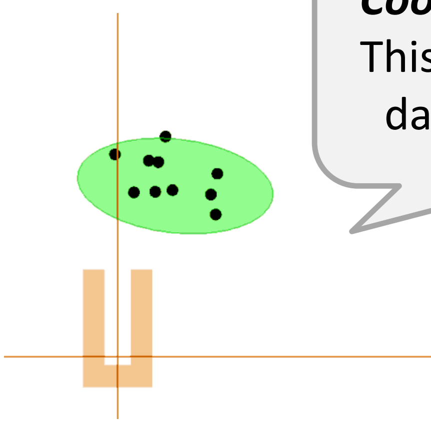
$$\boldsymbol{A}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{c} \sim \mathcal{N}(\boldsymbol{\mu}^{L}, \boldsymbol{\Sigma}^{L}),$$

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

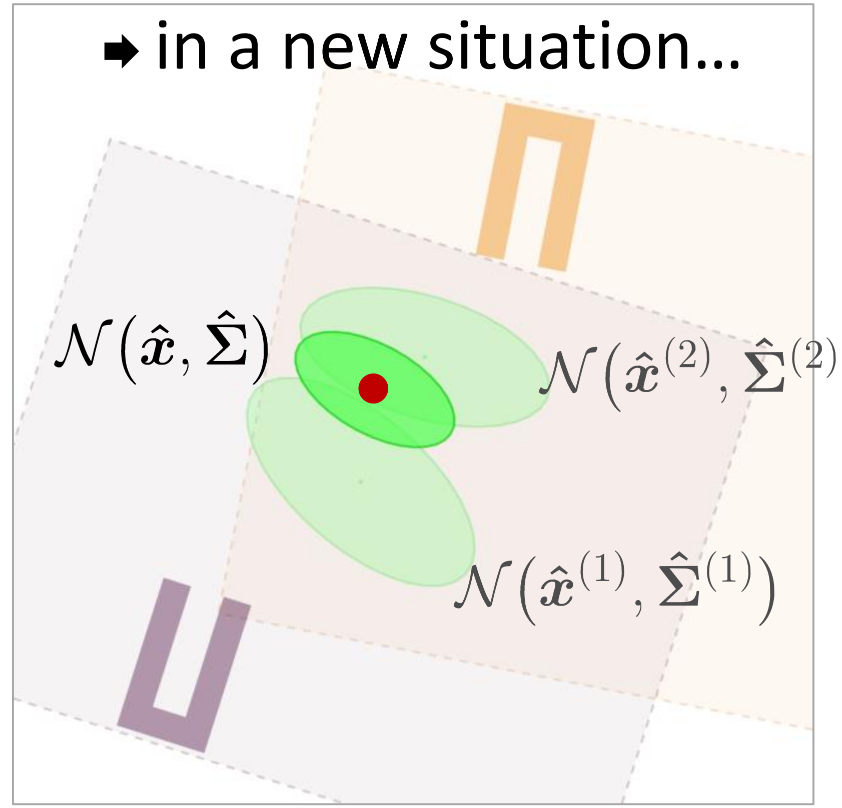
Example exploiting linear transformation and product properties



Coordinate system 1:
This is where I expect data to be located!



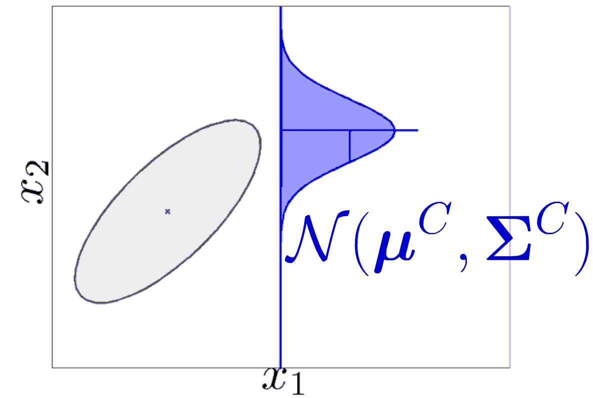
Coordinate system 2:
This is where I expect data to be located!



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

→ Product of linearly transformed Gaussians

Conditional distribution



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

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

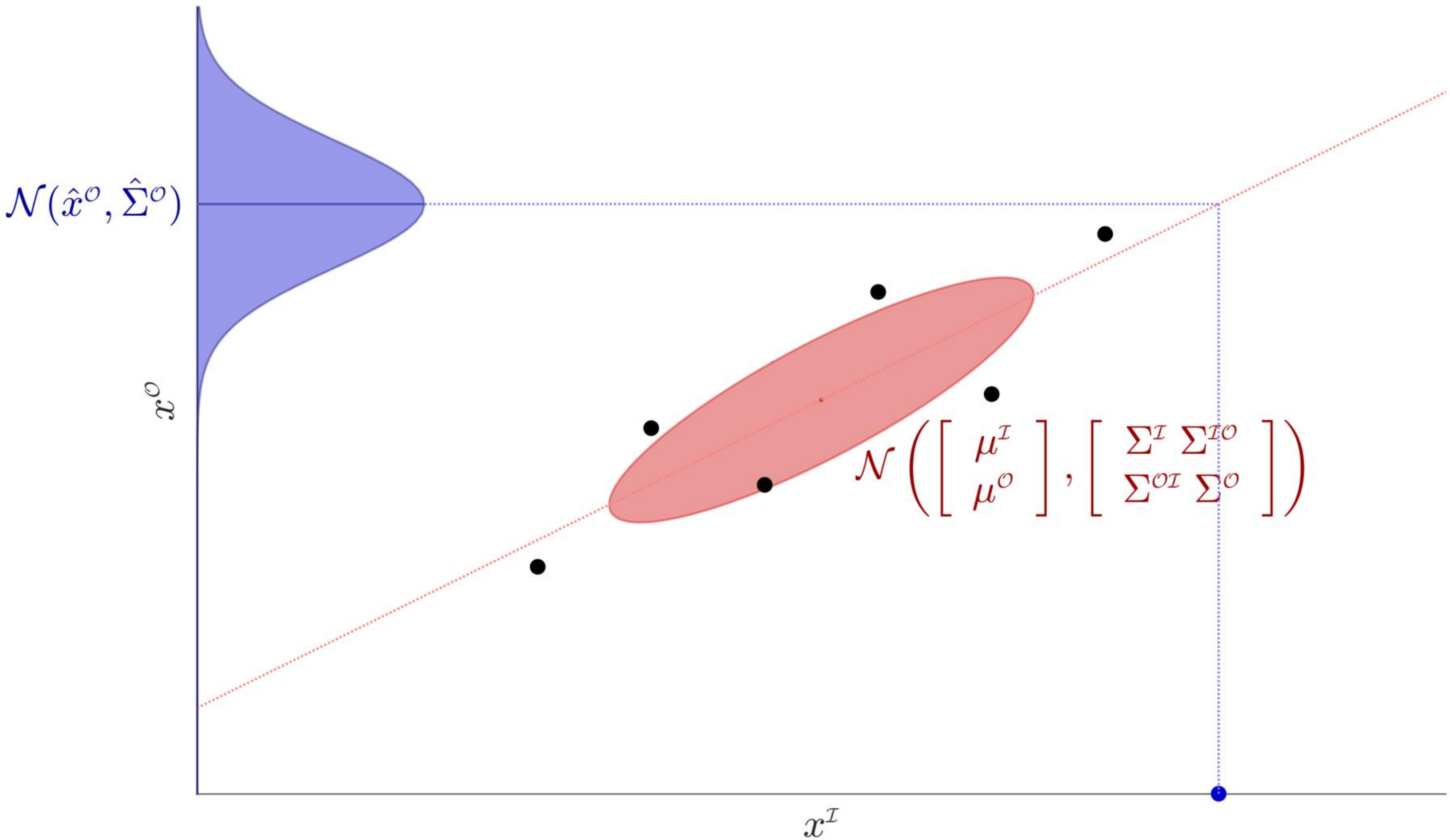
The conditional probability $\mathcal{P}(\boldsymbol{x}_2|\boldsymbol{x}_1)$ is defined by

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

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

Conditional distribution

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



→ Linear regression from joint distribution

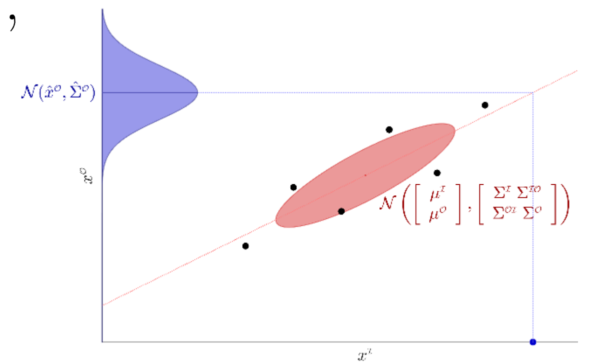
Conditional distribution

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 $\boldsymbol{x}^{\mathcal{O}} | \boldsymbol{x}^{\mathcal{I}} \sim \mathcal{N}\big(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\big)$, with parameters

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

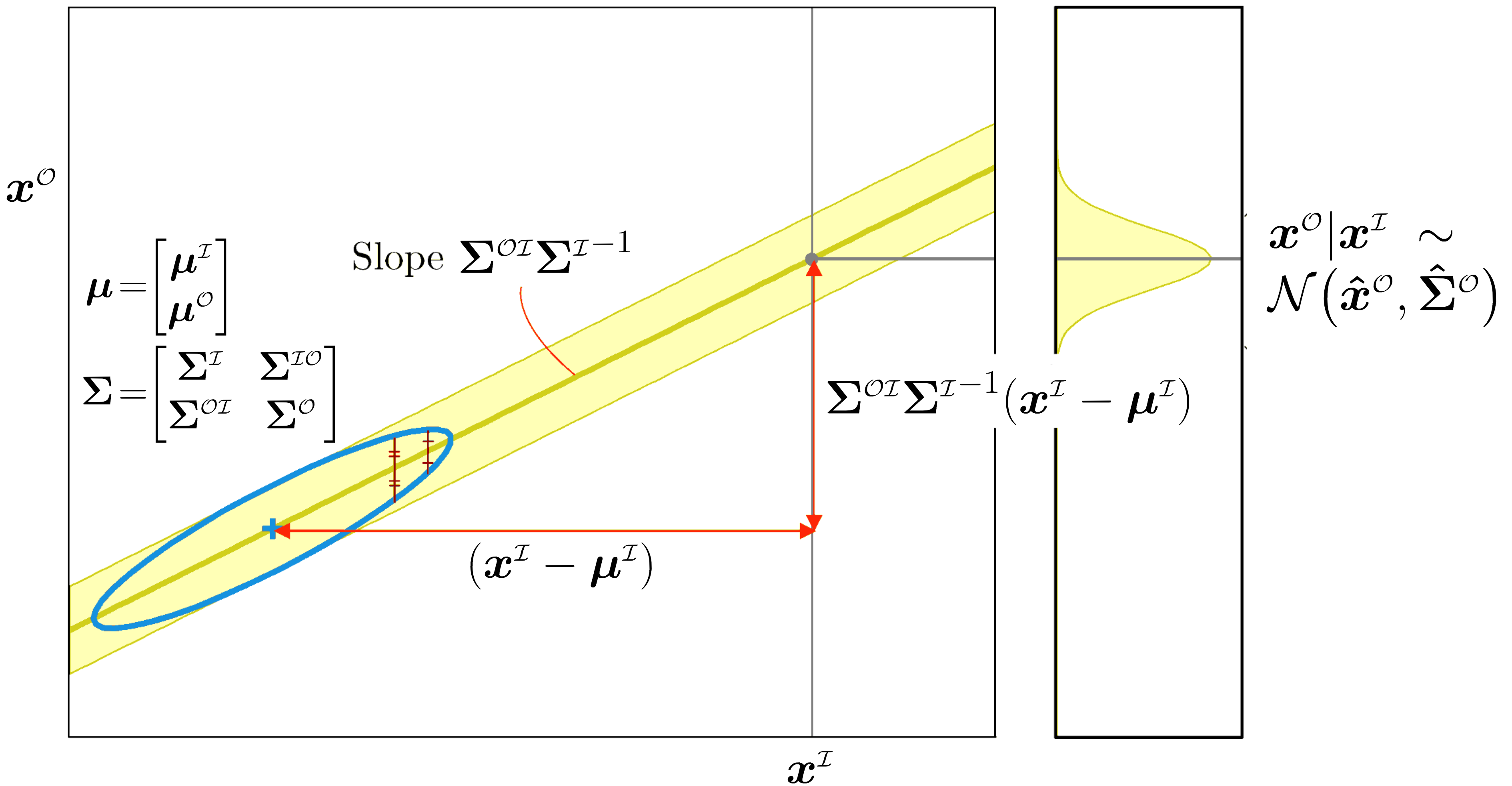


We can see that $\hat{\boldsymbol{x}}^{\mathcal{O}}$ is linearly dependent on $\boldsymbol{x}^{\mathcal{I}}$, and that $\hat{\boldsymbol{\Sigma}}^{\mathcal{O}}$ is independent of $\boldsymbol{x}^{\mathcal{I}}$.

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 - Geometric interpretation

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



Conditional distribution - Resolution

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

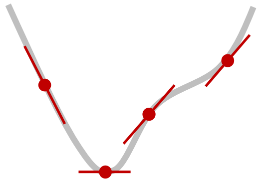
We want to find the distribution of $\boldsymbol{x}^{\mathcal{O}}$ that maximizes the log-likelihood

$$\begin{aligned} f(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \log\big(\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\big) \\ &= -\frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) - \frac{D}{2}\log(2\pi), \end{aligned}$$

when $\boldsymbol{x}^{\mathcal{I}}$ is known and acts as a constant.

This can be computed by deriving the above equation and equating to zero, namely

$$\frac{\partial f}{\partial \boldsymbol{x}^{\mathcal{O}}} = 0.$$



Conditional distribution - Resolution

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{\Gamma}^{\mathcal{I}} & \mathbf{\Gamma}^{\mathcal{IO}} \\ \mathbf{\Gamma}^{\mathcal{OI}} & \mathbf{\Gamma}^{\mathcal{O}} \end{bmatrix}$$

To do this, we first note that $\mathbf{\Sigma}^{-1}$ can be partitioned as

$$\begin{aligned}
\mathbf{\Sigma}^{-1} = \mathbf{\Gamma} &= \begin{bmatrix} \mathbf{\Gamma}^{\mathcal{I}} & \mathbf{\Gamma}^{\mathcal{IO}} \\ \mathbf{\Gamma}^{\mathcal{OI}} & \mathbf{\Gamma}^{\mathcal{O}} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{I} & -\mathbf{\Sigma}^{\mathcal{I}-1}\mathbf{\Sigma}^{\mathcal{IO}} \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}^{\mathcal{I}-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{S}^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \mathbf{0} \\ -\mathbf{\Sigma}^{\mathcal{OI}}\mathbf{\Sigma}^{\mathcal{I}-1} & \boldsymbol{I} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{\Sigma}^{\mathcal{I}-1} + \mathbf{\Sigma}^{\mathcal{I}-1}\mathbf{\Sigma}^{\mathcal{IO}}\boldsymbol{S}^{-1}\mathbf{\Sigma}^{\mathcal{OI}}\mathbf{\Sigma}^{\mathcal{I}-1} & -\mathbf{\Sigma}^{\mathcal{I}-1}\mathbf{\Sigma}^{\mathcal{IO}}\boldsymbol{S}^{-1} \\ -\boldsymbol{S}^{-1}\mathbf{\Sigma}^{\mathcal{OI}}\mathbf{\Sigma}^{\mathcal{I}-1} & \boldsymbol{S}^{-1} \end{bmatrix},
\end{aligned}$$

where $\boldsymbol{S} = \mathbf{\Sigma}^{\mathcal{O}} - \mathbf{\Sigma}^{\mathcal{OI}}\mathbf{\Sigma}^{\mathcal{I}-1}\mathbf{\Sigma}^{\mathcal{IO}}$ is the **Schur complement** of $\mathbf{\Sigma}$.

The above result can be shown by using a LDU decomposition of $\mathbf{\Sigma}$, where D is a diagonal matrix and L and U are atomic triangular matrices (lower and upper, respectively), and then computing its inverse by exploiting the inversion properties of diagonal and atomic triangular matrices.

Conditional distribution - Resolution

$$\boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{\Gamma}^{\mathcal{I}} & \boldsymbol{\Gamma}^{\mathcal{IO}} \\ \boldsymbol{\Gamma}^{\mathcal{OI}} & \boldsymbol{\Gamma}^{\mathcal{O}} \end{bmatrix}$$

With such partitioning, we can see that

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

$$\begin{aligned} -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Gamma}(\boldsymbol{x}-\boldsymbol{\mu}) = &-\frac{1}{2}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}})^{\top}\boldsymbol{\Gamma}^{\mathcal{I}}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}}) \\ &-\frac{1}{2}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}})^{\top}\boldsymbol{\Gamma}^{\mathcal{IO}}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}}) \\ &-\frac{1}{2}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}})^{\top}\boldsymbol{\Gamma}^{\mathcal{OI}}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}}) \\ &-\frac{1}{2}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}})^{\top}\boldsymbol{\Gamma}^{\mathcal{O}}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}}). \end{aligned}$$

With the symmetry of precision matrices ($\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^{\top}$), we have

$$\begin{aligned} -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Gamma}(\boldsymbol{x}-\boldsymbol{\mu}) &= -\frac{1}{2}\boldsymbol{x}^{\top}\boldsymbol{\Gamma}(\boldsymbol{x}-\boldsymbol{\mu}) + \frac{1}{2}\boldsymbol{\mu}^{\top}\boldsymbol{\Gamma}(\boldsymbol{x}-\boldsymbol{\mu}) \\ &= -\frac{1}{2}\boldsymbol{x}^{\top}\boldsymbol{\Gamma}\boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^{\top}\boldsymbol{\Gamma}\boldsymbol{\mu} + \frac{1}{2}\boldsymbol{\mu}^{\top}\boldsymbol{\Gamma}\boldsymbol{x} - \frac{1}{2}\boldsymbol{\mu}^{\top}\boldsymbol{\Gamma}\boldsymbol{\mu} \\ &= -\frac{1}{2}\boldsymbol{x}^{\top}\boldsymbol{\Gamma}\boldsymbol{x} + \boldsymbol{x}^{\top}\boldsymbol{\Gamma}\boldsymbol{\mu} - \frac{1}{2}\boldsymbol{\mu}^{\top}\boldsymbol{\Gamma}\boldsymbol{\mu}. \end{aligned}$$

Conditional distribution - Resolution

$$f(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) - \frac{D}{2}\log(2\pi)$$

By using the linear algebra relations

$$-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Gamma}(\boldsymbol{x}-\boldsymbol{\mu}) = -\frac{1}{2}\boldsymbol{x}^{\top}\boldsymbol{\Gamma}\boldsymbol{x} + \boldsymbol{x}^{\top}\boldsymbol{\Gamma}\boldsymbol{\mu} - \frac{1}{2}\boldsymbol{\mu}^{\top}\boldsymbol{\Gamma}\boldsymbol{\mu}$$

$$\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{x}^{\top}\boldsymbol{A} = \frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{A}^{\top}\boldsymbol{x} = \boldsymbol{A}, \qquad \frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{x}^{\top}\boldsymbol{A}\boldsymbol{x} = (\boldsymbol{A}+\boldsymbol{A}^{\top})\boldsymbol{x},$$

and by exploiting the derivation chain rule and the symmetry of covariances, we obtain

$$-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Gamma}(\boldsymbol{x}-\boldsymbol{\mu}) = -\frac{1}{2}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}})^{\top}\boldsymbol{\Gamma}^{\mathcal{I}}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}}) - \frac{1}{2}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}})^{\top}\boldsymbol{\Gamma}^{\mathcal{I}\mathcal{O}}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}})$$
$$-\frac{1}{2}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}})^{\top}\boldsymbol{\Gamma}^{\mathcal{O}\mathcal{I}}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}}) - \frac{1}{2}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}})^{\top}\boldsymbol{\Gamma}^{\mathcal{O}}(\boldsymbol{x}^{\mathcal{O}}-\boldsymbol{\mu}^{\mathcal{O}})$$

$$\frac{\partial f}{\partial \boldsymbol{x}^{\mathcal{O}}} = -\boldsymbol{\Gamma}^{\mathcal{O}}\boldsymbol{\mu}^{\mathcal{O}} + \boldsymbol{\Gamma}^{\mathcal{O}\mathcal{I}}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}}) + \boldsymbol{\Gamma}^{\mathcal{O}}\boldsymbol{x}^{\mathcal{O}} = 0$$
$$\iff \quad \hat{\boldsymbol{x}}^{\mathcal{O}} = \boldsymbol{\mu}^{\mathcal{O}} - \boldsymbol{\Gamma}^{\mathcal{O}^{-1}}\boldsymbol{\Gamma}^{\mathcal{O}\mathcal{I}}(\boldsymbol{x}^{\mathcal{I}}-\boldsymbol{\mu}^{\mathcal{I}}).$$

$$\boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \boldsymbol{\Sigma}^{\mathcal{I}^{-1}} + \boldsymbol{\Sigma}^{\mathcal{I}^{-1}}\boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}}\boldsymbol{S}^{-1}\boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}}\boldsymbol{\Sigma}^{\mathcal{I}^{-1}} & -\boldsymbol{\Sigma}^{\mathcal{I}^{-1}}\boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}}\boldsymbol{S}^{-1} \\ -\boldsymbol{S}^{-1}\boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}}\boldsymbol{\Sigma}^{\mathcal{I}^{-1}} & \boldsymbol{S}^{-1} \end{bmatrix}$$
$$\boldsymbol{S} = \boldsymbol{\Sigma}^{\mathcal{O}} - \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}}\boldsymbol{\Sigma}^{\mathcal{I}^{-1}}\boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}}$$

By using the Schur decomposition, we can see that

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

Conditional distribution - Resolution

$$\boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \boldsymbol{\Sigma}^{\mathcal{I}^{-1}} + \boldsymbol{\Sigma}^{\mathcal{I}^{-1}} \boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}} \boldsymbol{S}^{-1} \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}^{\mathcal{I}^{-1}} & -\boldsymbol{\Sigma}^{\mathcal{I}^{-1}} \boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}} \boldsymbol{S}^{-1} \\ -\boldsymbol{S}^{-1} \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}^{\mathcal{I}^{-1}} & \boldsymbol{S}^{-1} \end{bmatrix}$$

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

The associated covariance matrix $\hat{\boldsymbol{\Sigma}}^{\mathcal{O}}$ measuring the error of this estimate is given by the inverse of the Hessian matrix $\boldsymbol{H}$. We have

$$\boldsymbol{H} = \frac{\partial^2 f}{\partial \boldsymbol{x}^{\mathcal{O}} \boldsymbol{x}^{\mathcal{O}^{\top}}} = \boldsymbol{\Gamma}^{\mathcal{O}} \quad \Rightarrow \quad \hat{\boldsymbol{\Sigma}}^{\mathcal{O}} = \boldsymbol{\Gamma}^{\mathcal{O}^{-1}}.$$

We can then see that

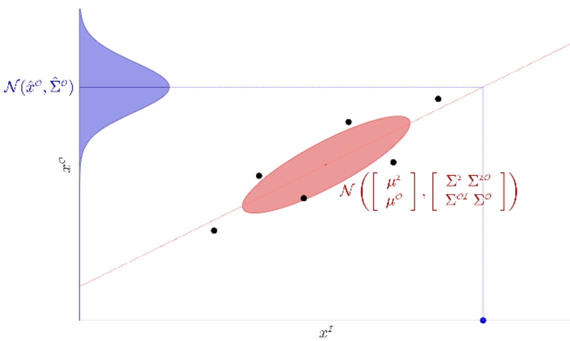
$$\hat{\boldsymbol{\Sigma}}^{\mathcal{O}} = \boldsymbol{S} = \boldsymbol{\Sigma}^{\mathcal{O}} - \boldsymbol{\Sigma}^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}^{\mathcal{I}^{-1}} \boldsymbol{\Sigma}^{\mathcal{I}\mathcal{O}}.$$

Note that in some cases, evaluating the conditional distribution with precision matrices is computationally more efficient than with covariance matrices.

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

Conditional distribution - Summary

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



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

If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Gamma}^{-1})$, we have that $\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Gamma}}^{\mathcal{O}^{-1}}\right)$, with parameters

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

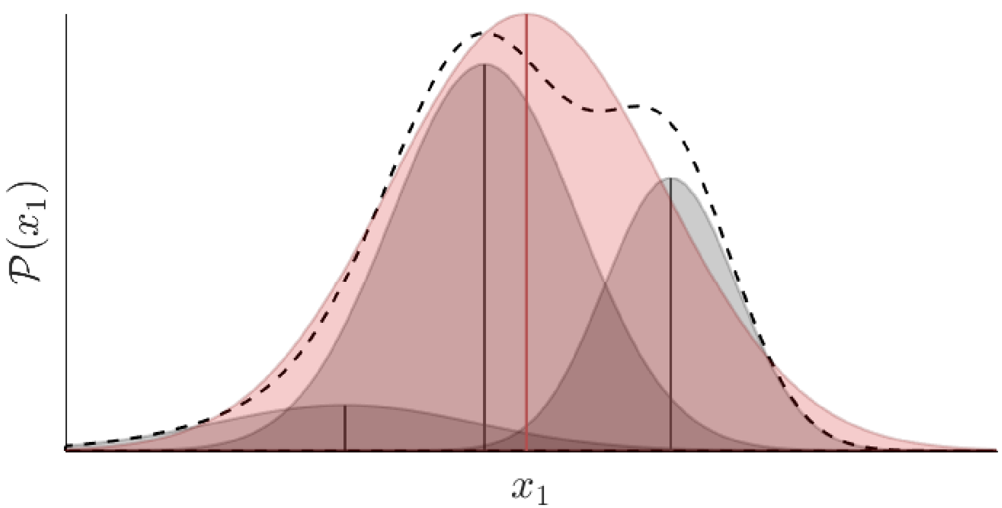
Gaussian estimate of a mixture of Gaussians

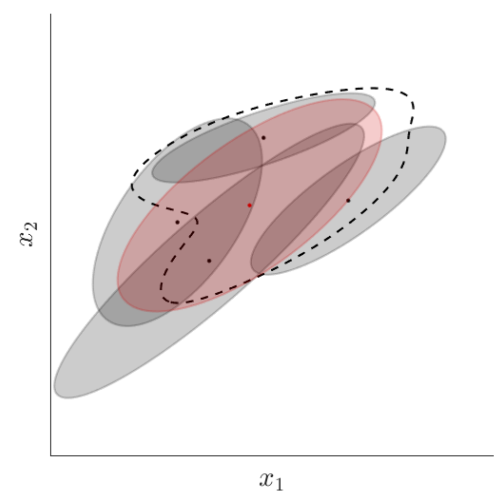
We can approximate a mixture of Gaussians $\sum_{i=1}^{K} h_i \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ with a single Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, by **moment matching of the means (first moments) and covariances (second moments)** with

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

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

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





Gaussian estimate of a mixture of Gaussians

The result can be shown by developing the expressions

$$\mathbb{E}(\boldsymbol{x}) = \boldsymbol{\mu}, \quad \boldsymbol{\Sigma} = \text{cov}(\boldsymbol{x}) = \mathbb{E}(\boldsymbol{x}\boldsymbol{x}^{\top}) - \mathbb{E}(\boldsymbol{x})\mathbb{E}(\boldsymbol{x}^{\top}) = \mathbb{E}(\boldsymbol{x}\boldsymbol{x}^{\top}) - \boldsymbol{\mu}\boldsymbol{\mu}^{\top}$$
$$\iff \mathbb{E}(\boldsymbol{x}\boldsymbol{x}^{\top}) = \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^{\top}.$$

By considering datapoints $\boldsymbol{x}$ distributed with a mixture of Gaussians

$$\mathcal{P}(\boldsymbol{x}) = \sum_{i=1}^{K} \mathcal{P}(z_i)\mathcal{P}(\boldsymbol{x}|z_i) = \sum_{i=1}^{K} h_i \, \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \, ,$$

the mean is computed as

$$\begin{aligned}
\boldsymbol{\mu} = \mathbb{E}(\boldsymbol{x}) = \int \boldsymbol{x} \, \mathcal{P}(\boldsymbol{x}) \, d\boldsymbol{x} &= \int \boldsymbol{x} \sum_{i=1}^{K} h_i \, \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \, d\boldsymbol{x} \\
&= \sum_{i=1}^{K} h_i \int \boldsymbol{x} \, \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \, d\boldsymbol{x} \\
&= \sum_{i=1}^{K} h_i \, \boldsymbol{\mu}_i.
\end{aligned}$$

Gaussian estimate of a mixture of Gaussians

By noting that

$\mathbb{E}(\boldsymbol{x}\boldsymbol{x}^\top) = \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^\top$

$$\begin{aligned}
\mathbb{E}(\boldsymbol{x}\boldsymbol{x}^\top) &= \int \boldsymbol{x}\boldsymbol{x}^\top \mathcal{P}(\boldsymbol{x})\ d\boldsymbol{x} \\
&= \int \sum_{i=1}^{K} h_i\ \boldsymbol{x}\boldsymbol{x}^\top \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\, d\boldsymbol{x} \\
&= \sum_{i=1}^{K} h_i \int \boldsymbol{x}\boldsymbol{x}^\top \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\, d\boldsymbol{x} \\
&= \sum_{i=1}^{K} h_i \Big(\boldsymbol{\Sigma}_i + \boldsymbol{\mu}_i\boldsymbol{\mu}_i^\top\Big),
\end{aligned}$$

the covariance is then computed as

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

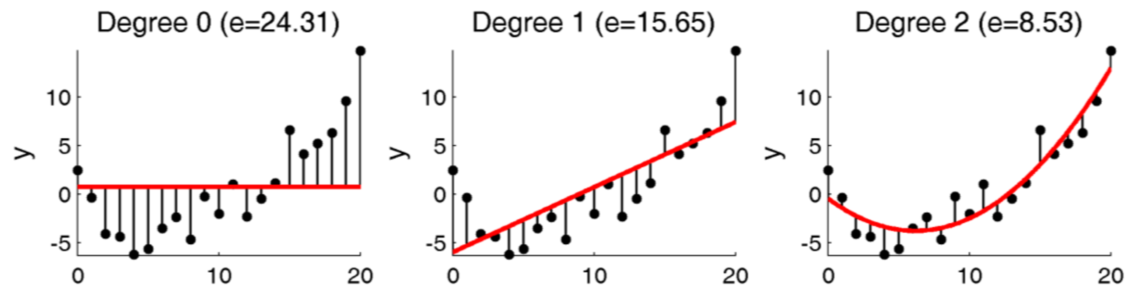
Locally weighted regression (LWR)

Python notebooks:
demo_LWR.ipynb

Matlab codes:
demo_LWR01.m

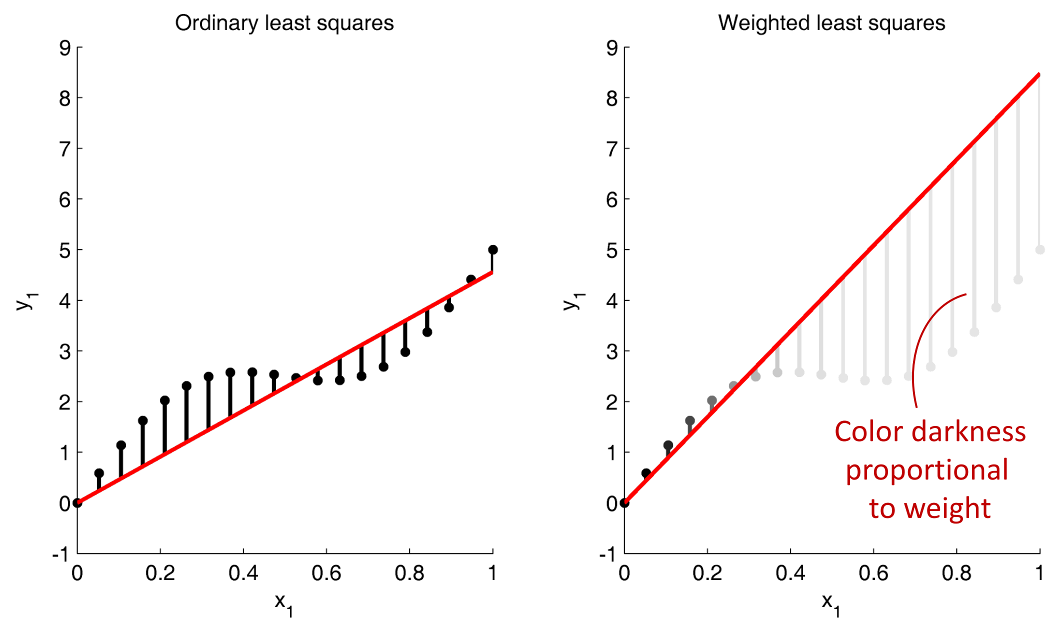
Previous lecture on linear regression

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



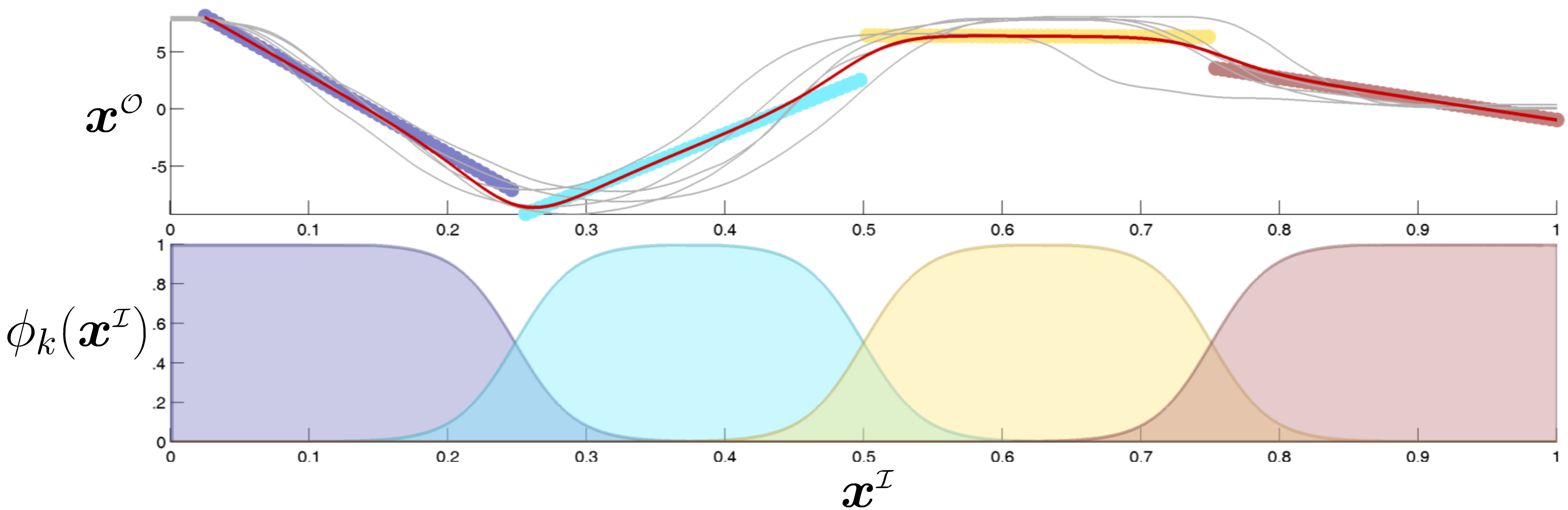
$\boldsymbol{X} = 1$ $\quad$ $\boldsymbol{X} = [\boldsymbol{x}, 1]$ $\quad$ $\boldsymbol{X} = [\boldsymbol{x}^2, \boldsymbol{x}, 1]$

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



Locally weighted regression (LWR)

Locally weighted regression (LWR) is a direct extension of the weighted least squares formulation in which K weighted regressions are performed on the same dataset $\{\boldsymbol{X}^{\mathcal{I}}, \boldsymbol{X}^{\mathcal{O}}\}$.

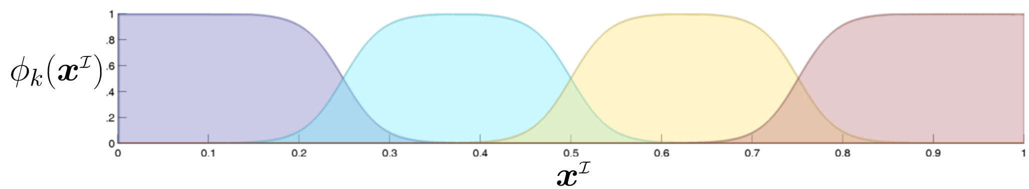


Locally weighted regression (LWR)

LWR computes K estimates $\hat{\boldsymbol{A}}_k$, each with a 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 its 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.

→ K **weighted regressions on the same dataset** $\{\boldsymbol{X}^{\mathcal{I}}, \boldsymbol{X}^{\mathcal{O}}\}$

→ **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.

$$\boldsymbol{X}^{\mathcal{I}} = [t_1, t_2, \ldots, t_N]^\top$$

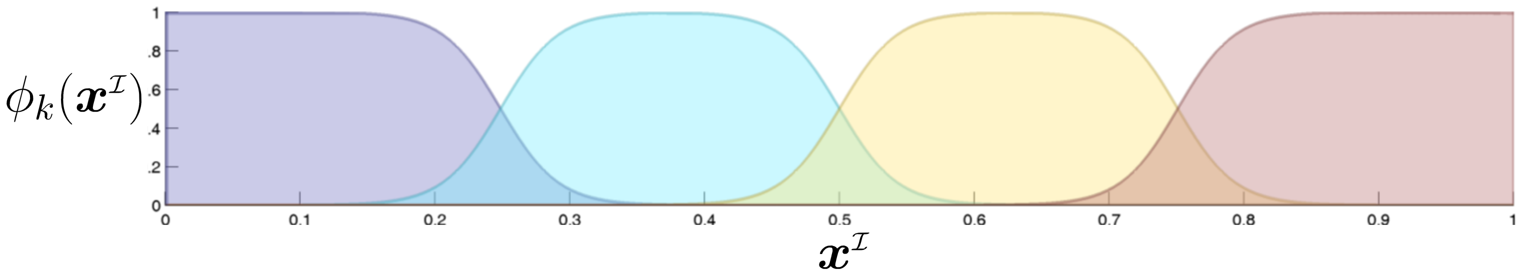
$$\hat{\boldsymbol{A}}_k = (\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{W}_k\boldsymbol{X}^{\mathcal{I}})^{-1}\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{W}_k\,\boldsymbol{X}^{\mathcal{O}}$$

An associated diagonal matrix

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

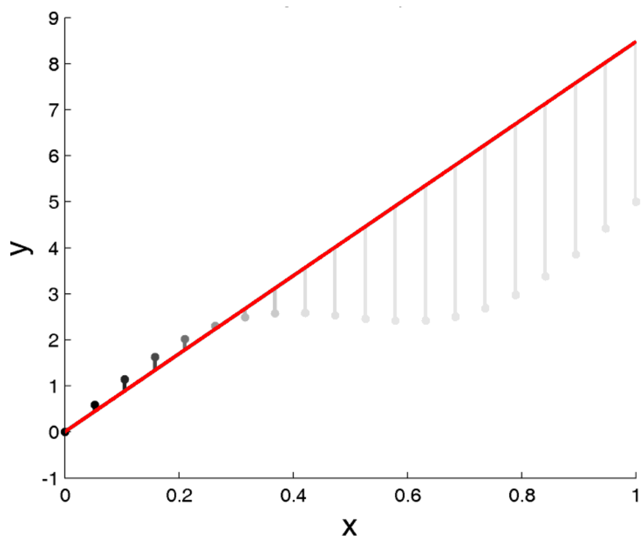
can be used to evaluate $\hat{\boldsymbol{A}}_k$. The result can then be used to compute

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

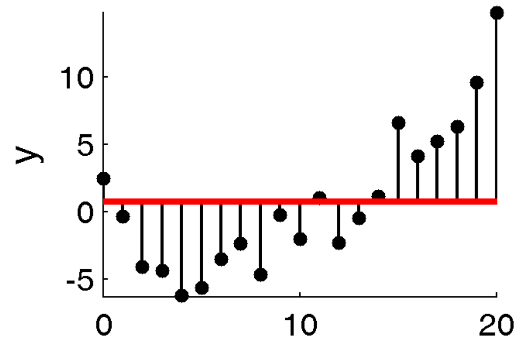


Locally weighted regression (LWR)

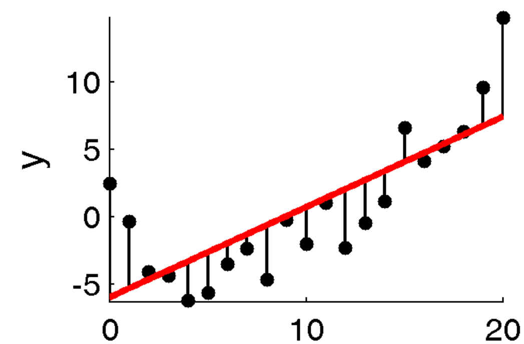
$$\hat{A} = (X^\top W X)^{-1} X^\top W Y$$



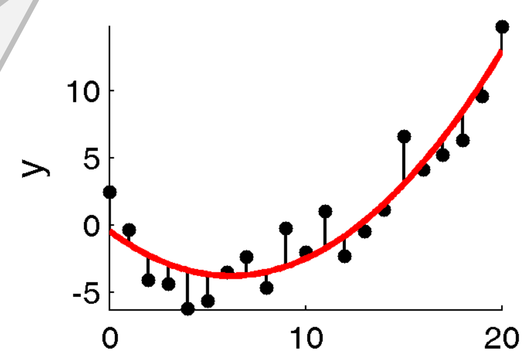
LWR can be used for local least squares polynomial fitting by changing the definition of the inputs.



$$x = 1$$

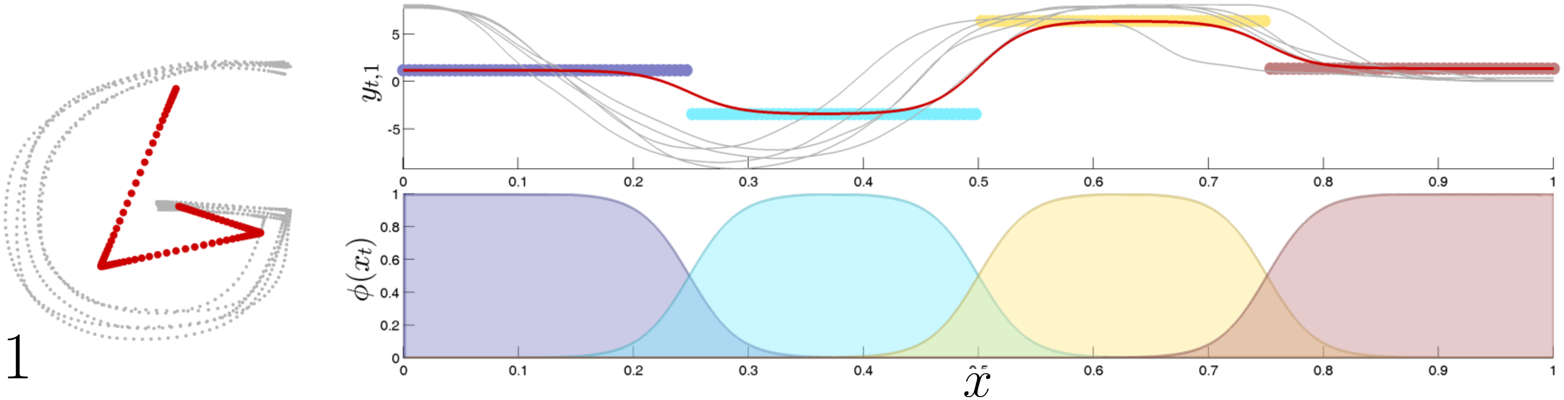


$$\boldsymbol{x} = [1, \mathrm{x}]$$

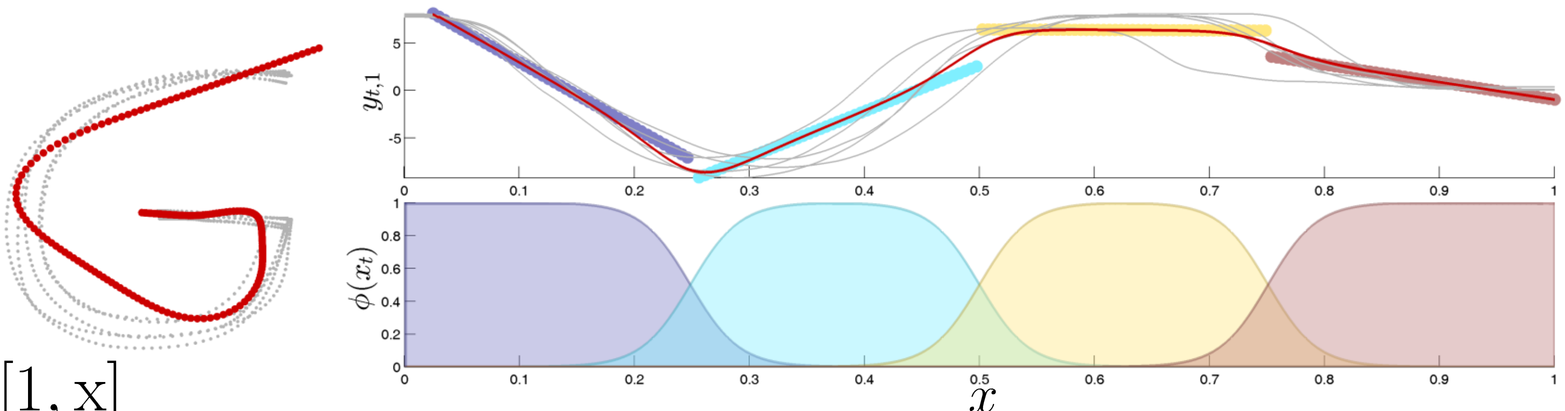


$$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$$

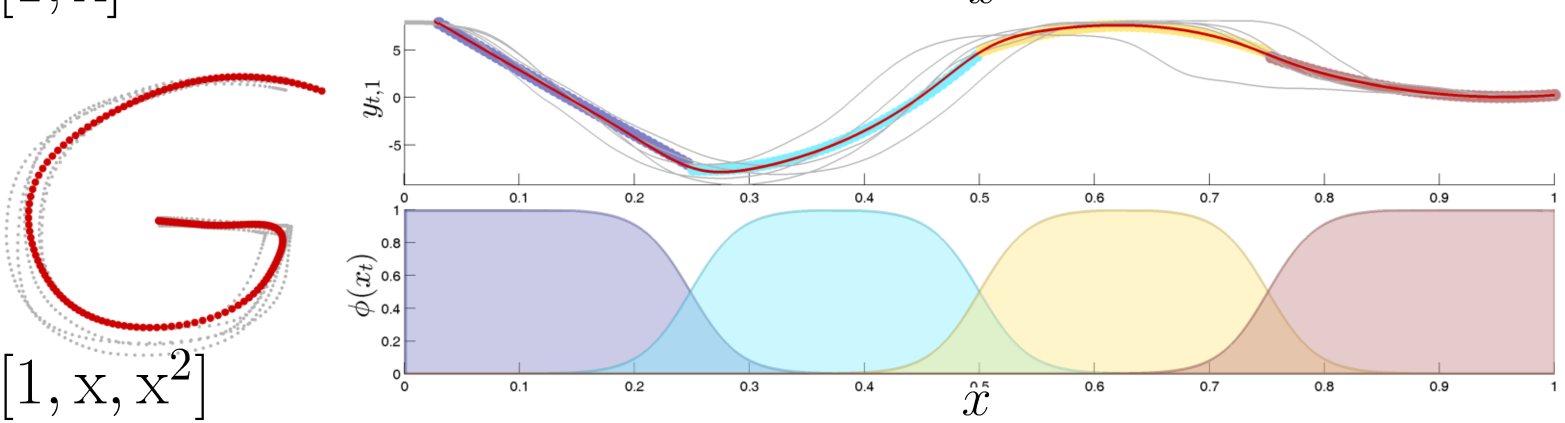
Locally weighted regression (LWR)



$x = 1$

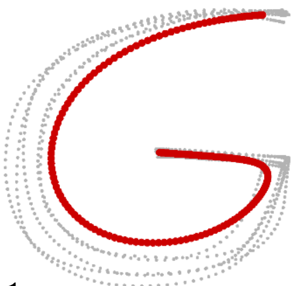


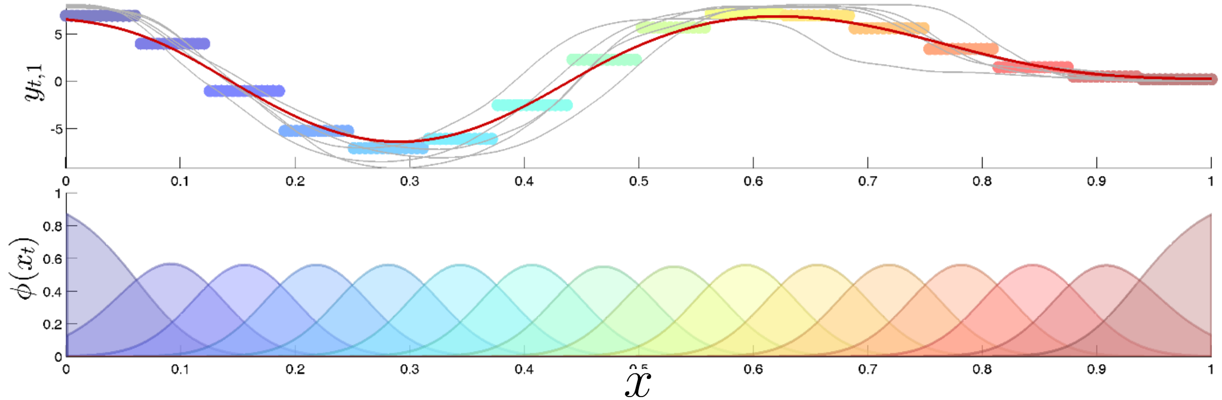
$\boldsymbol{x} = [1, \mathrm{x}]$



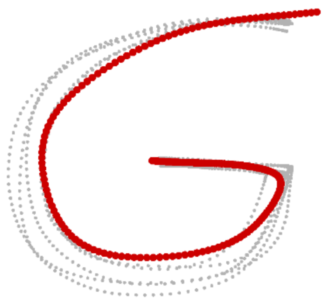
$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$

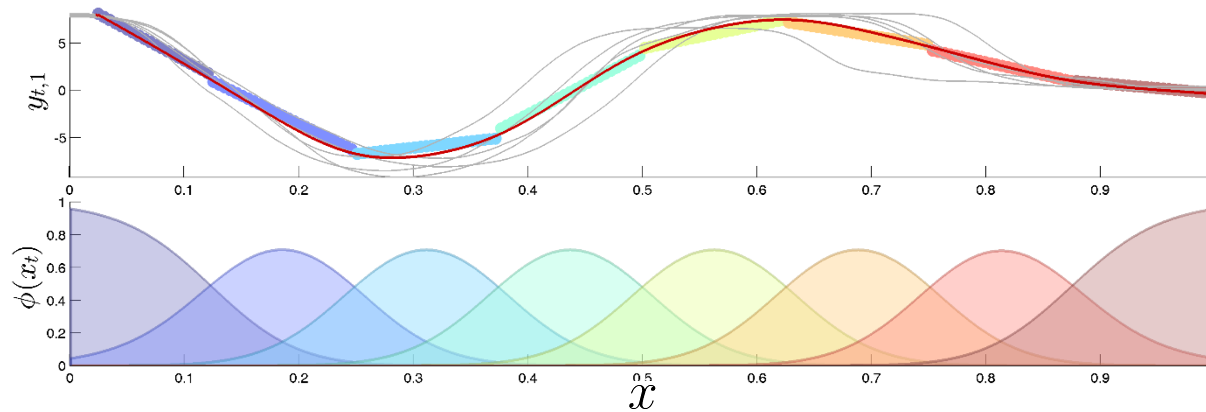
Locally weighted regression (LWR)



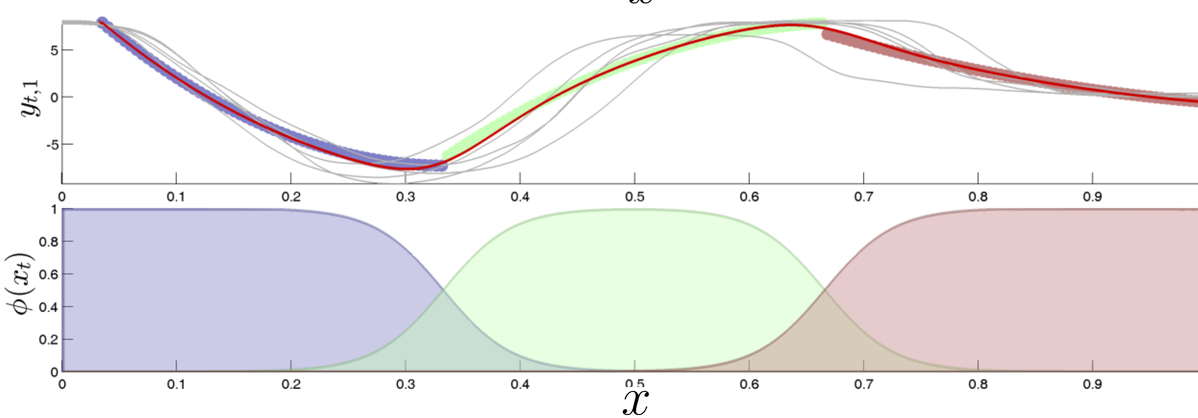


$x = 1$





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



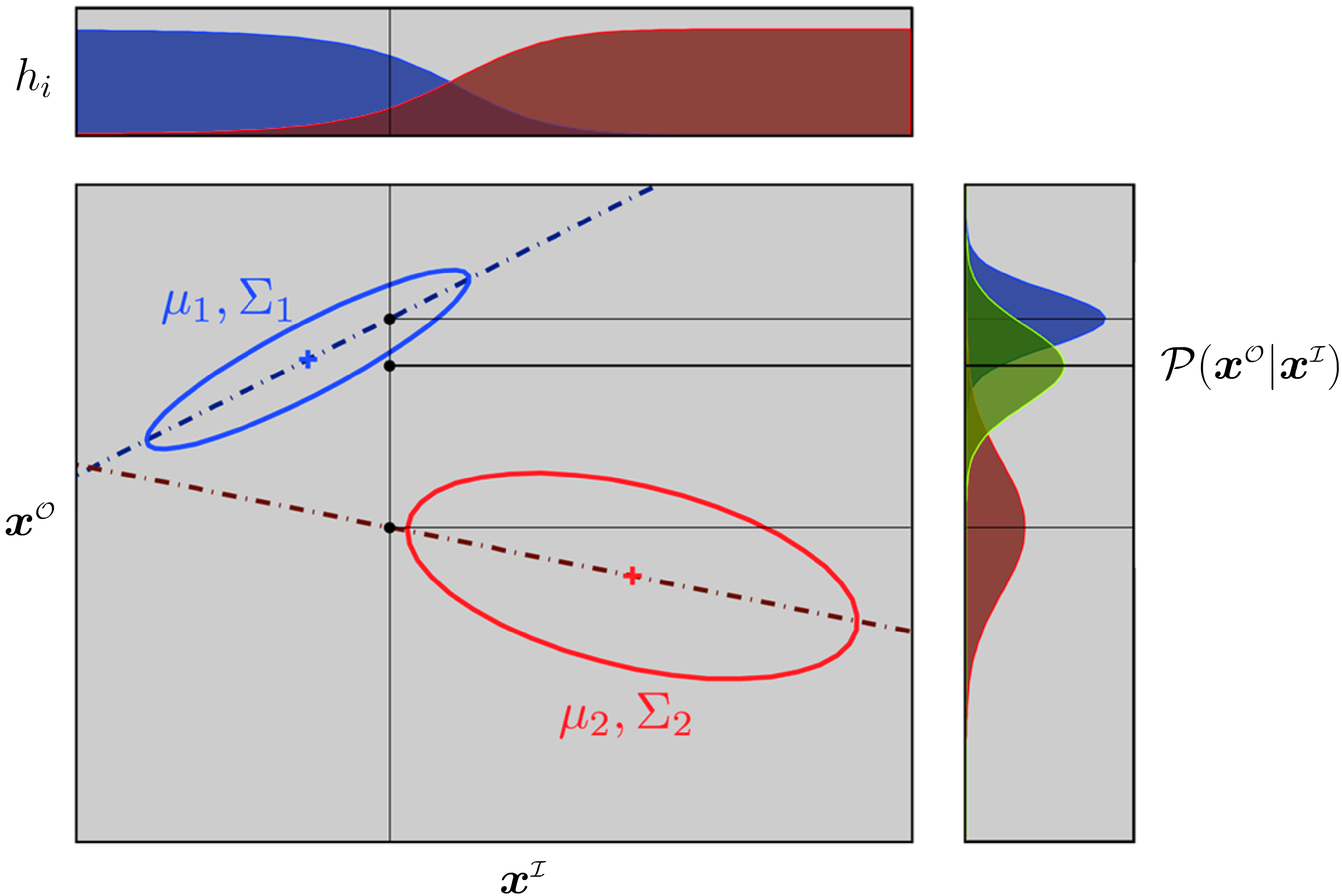
$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$

Gaussian mixture regression (GMR)

Python notebooks:
demo_GMR.ipynb

Matlab codes:
demo_GMR01.m
demo_GMR_polyFit01.m

Gaussian mixture regression (GMR)



Gaussian mixture regression (GMR)

- Gaussian mixture regression (GMR) is a nonlinear regression technique that does not model the regression function directly, but instead first models the **joint probability density of input-output data** in the form of a Gaussian mixture model (GMM).
- The computation relies on **linear transformation and conditioning properties** of multivariate normal distributions.
- GMR provides a regression approach in which **multivariate output distributions can be computed in an online manner**, with a computation time **independent of the number of datapoints** used to train the model, by exploiting the learned joint density model.
- In GMR, **both input and output variables can be multivariate**, and after learning, **any subset of input-output dimensions can be selected** for regression. This can for example be exploited to handle different sources of missing data, where expectations on the remaining dimensions can be computed as a multivariate distribution.

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

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

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

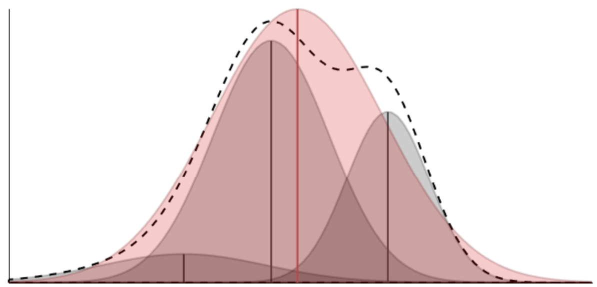
$$\hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} = \boldsymbol{\Sigma}_i^{\mathcal{O}} - \boldsymbol{\Sigma}_i^{\mathcal{O}\mathcal{I}} \boldsymbol{\Sigma}_i^{\mathcal{I}\,-1} \boldsymbol{\Sigma}_i^{\mathcal{I}\mathcal{O}}$$

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

computed with the marginal

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

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

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

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

Gaussian mixture regression (GMR)

This can be shown by computing

$$\hat{\boldsymbol{\mu}}^{\mathcal{O}} = \mathbb{E}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}}),$$
$$\hat{\boldsymbol{\Sigma}}^{\mathcal{O}} = \text{cov}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}}) = \mathbb{E}(\boldsymbol{x}^{\mathcal{O}}\boldsymbol{x}^{\mathcal{O}\top}|\boldsymbol{x}^{\mathcal{I}}) - \mathbb{E}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}})\mathbb{E}(\boldsymbol{x}^{\mathcal{O}\top}|\boldsymbol{x}^{\mathcal{I}}).$$

The conditional mean can be computed as

$$\hat{\boldsymbol{\mu}}^{\mathcal{O}} = \mathbb{E}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}}) = \sum_{i=1}^{K} h_i \; \hat{\boldsymbol{\mu}}_i^{\mathcal{O}}.$$

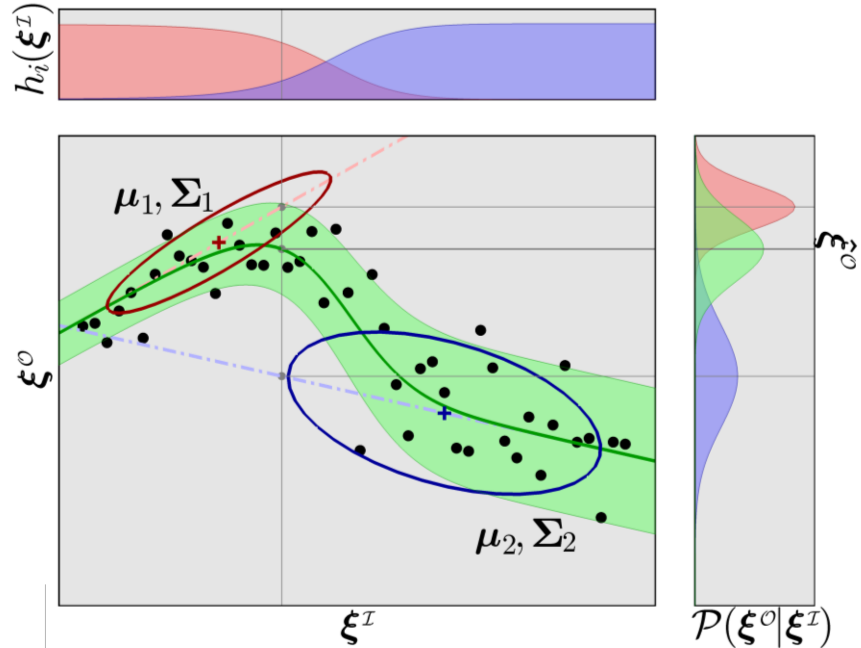
In order to evaluate the covariance, we first note that

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

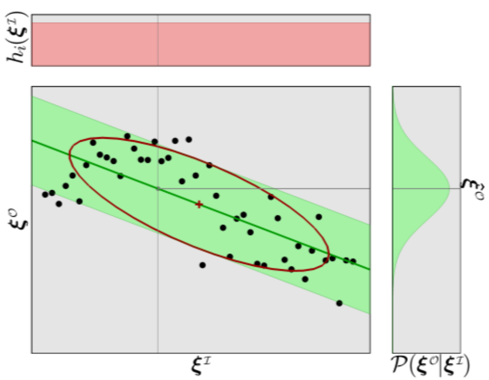
We then have

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

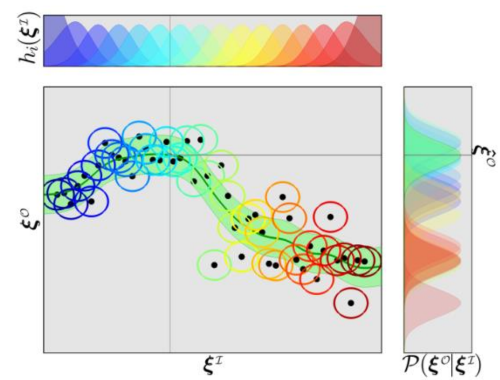
Gaussian mixture regression (GMR)



Least squares
linear regression

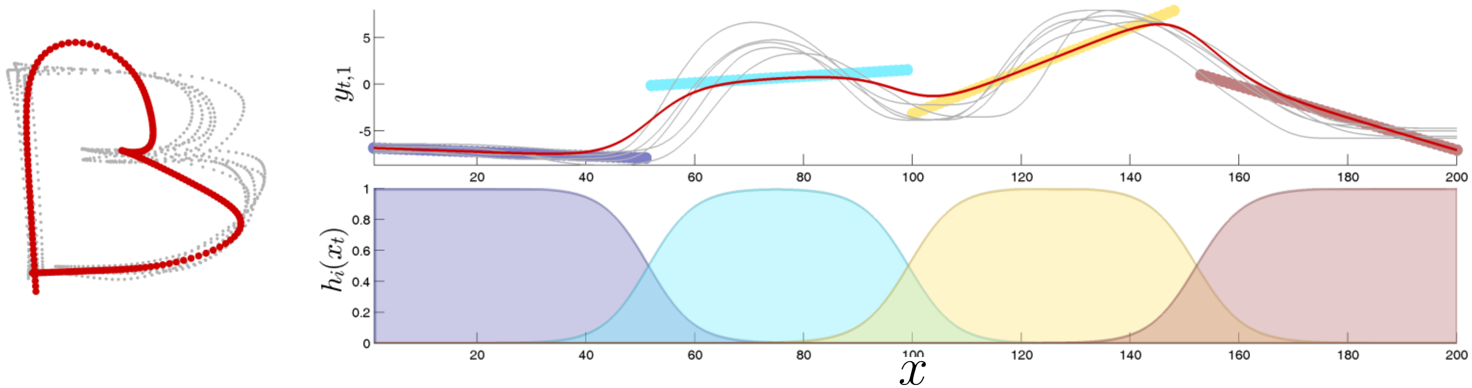


Nadaraya-Watson
kernel regression

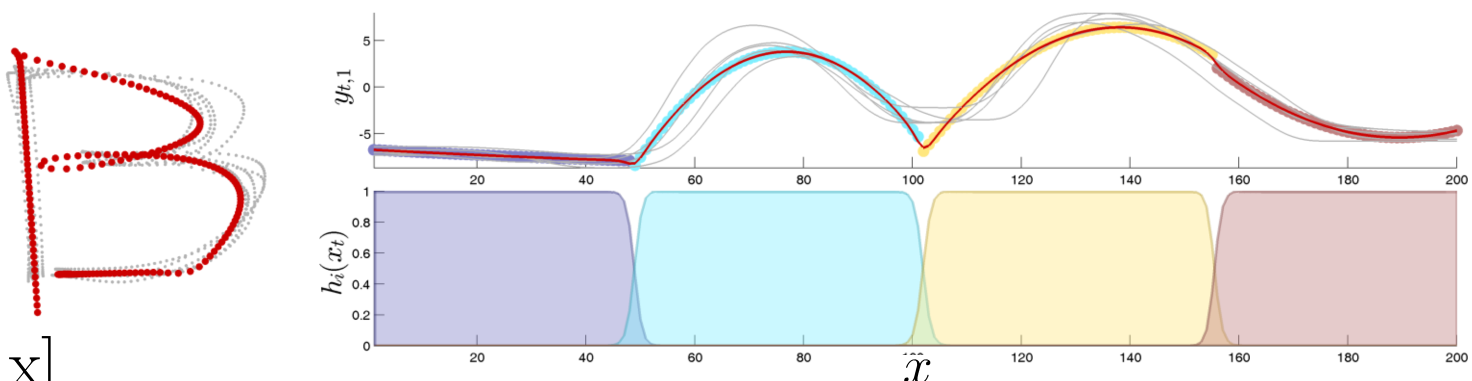


GMR can cover a large range
of regression approaches!

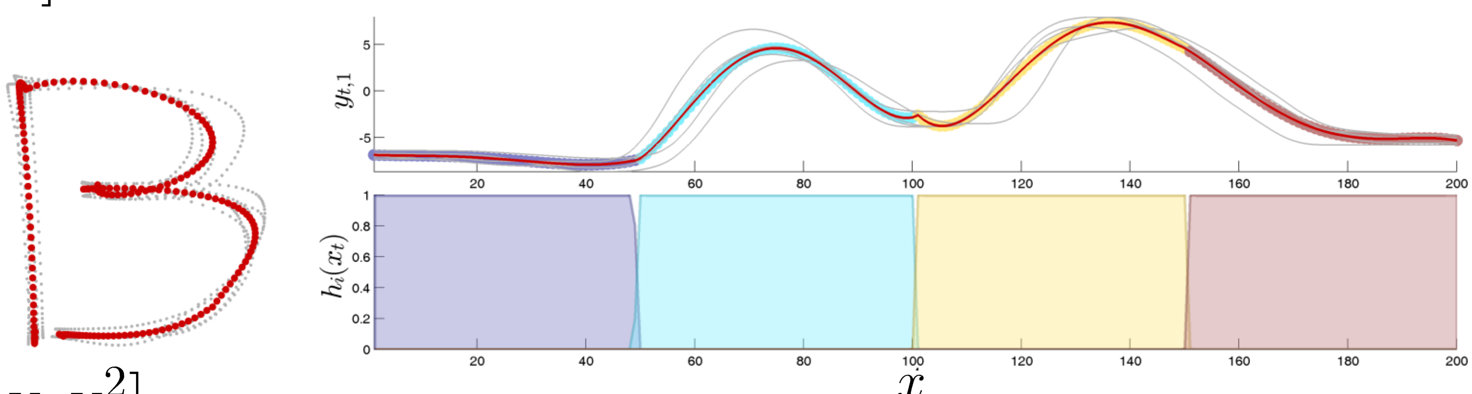
GMR for smooth piecewise polynomial fitting



$x = 1$

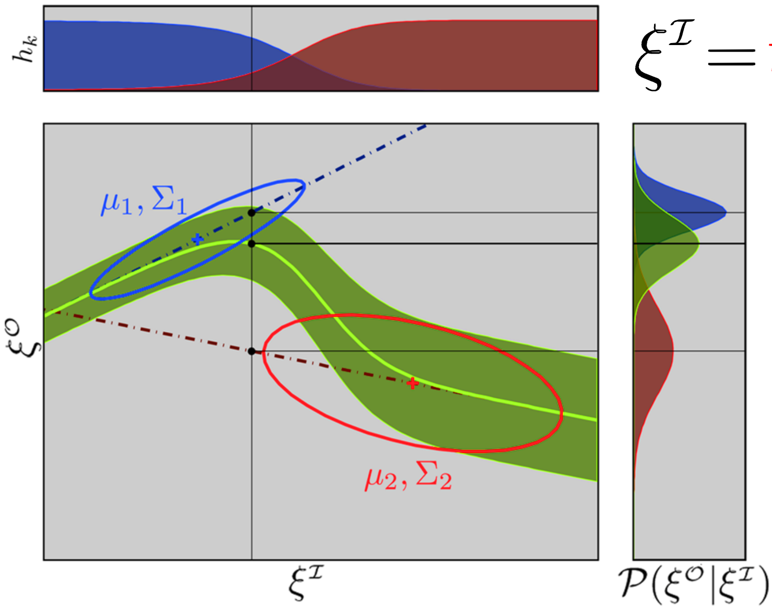


$\boldsymbol{x} = [1, \mathrm{x}]$



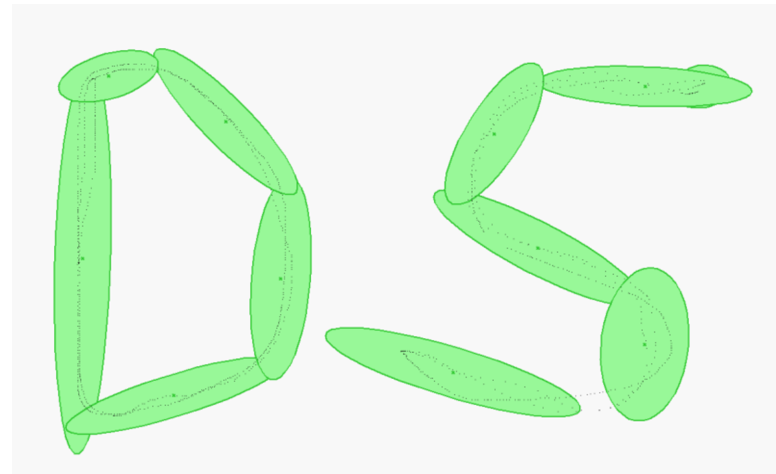
$\boldsymbol{x} = [1, \mathrm{x}, \mathrm{x}^2]$

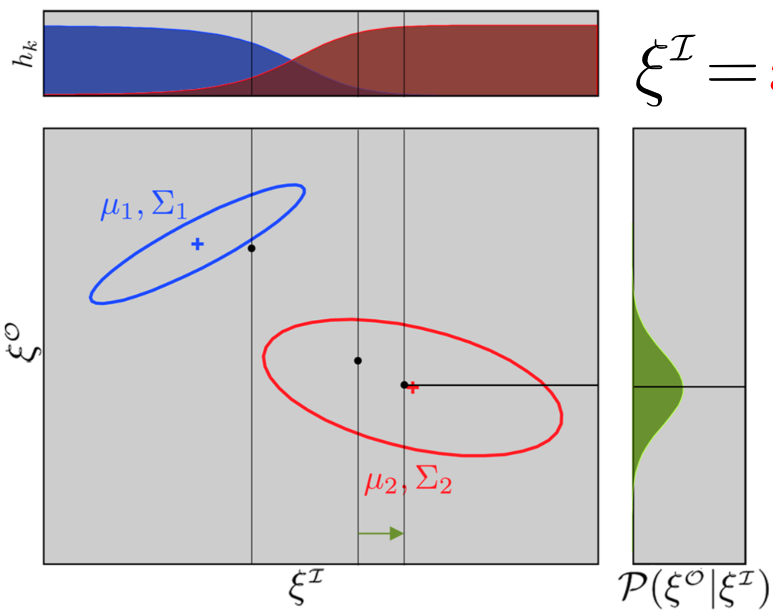
Gaussian mixture regression - Examples



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

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





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

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]

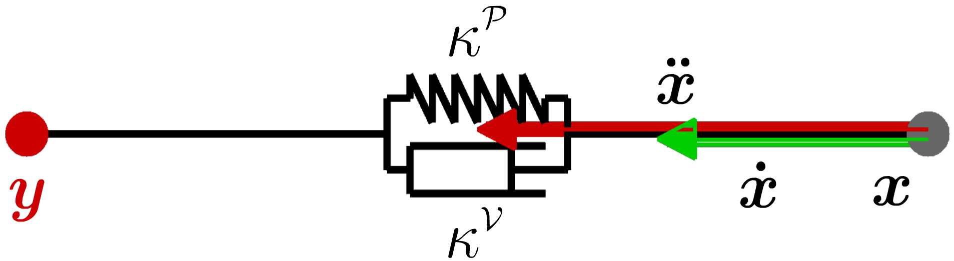
Example of application:
Dynamical movement primitives (DMP)

Python notebooks:
demo_DMP.ipynb
demo_DMP_GMR.ipynb

Matlab codes:
demo_DMP01.m
demo_DMP_GMR01.m

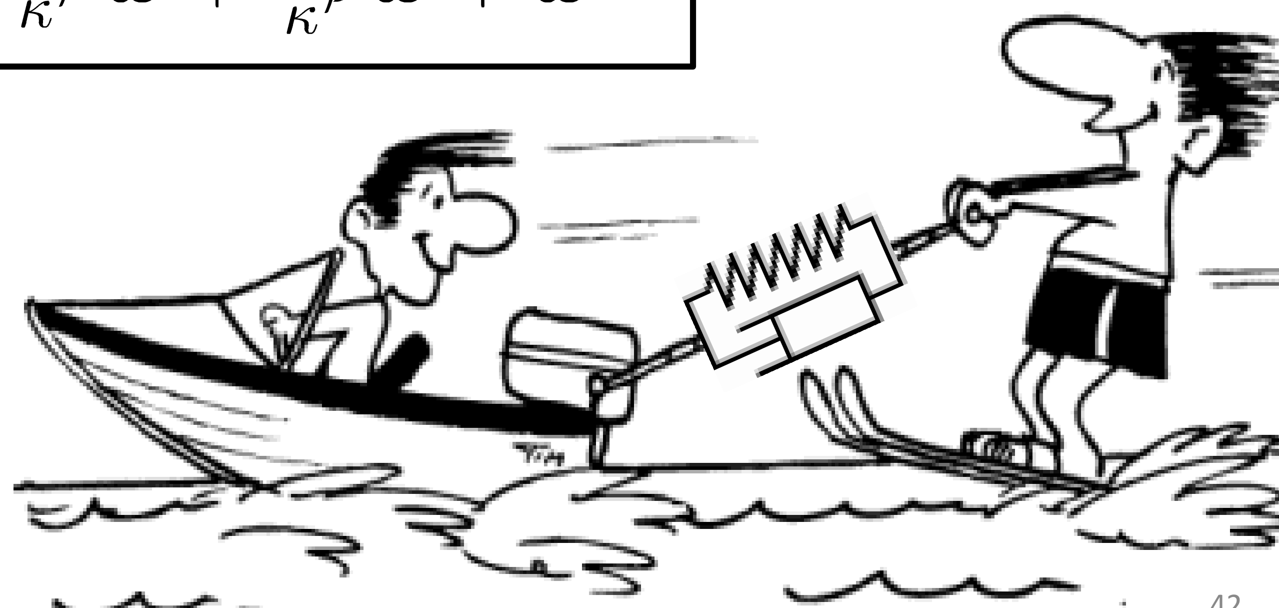
Dynamical movement primitives (DMP)

Spring-damper system



$$\ddot{\boldsymbol{x}} = \kappa^{\mathcal{P}}[\boldsymbol{y} - \boldsymbol{x}] - \kappa^{\mathcal{V}}\dot{\boldsymbol{x}}$$

$$\Rightarrow \boldsymbol{y} = \frac{1}{\kappa^{\mathcal{P}}}\ddot{\boldsymbol{x}} + \frac{\kappa^{\mathcal{V}}}{\kappa^{\mathcal{P}}}\dot{\boldsymbol{x}} + \boldsymbol{x}$$



Dynamical movement primitives (DMP)

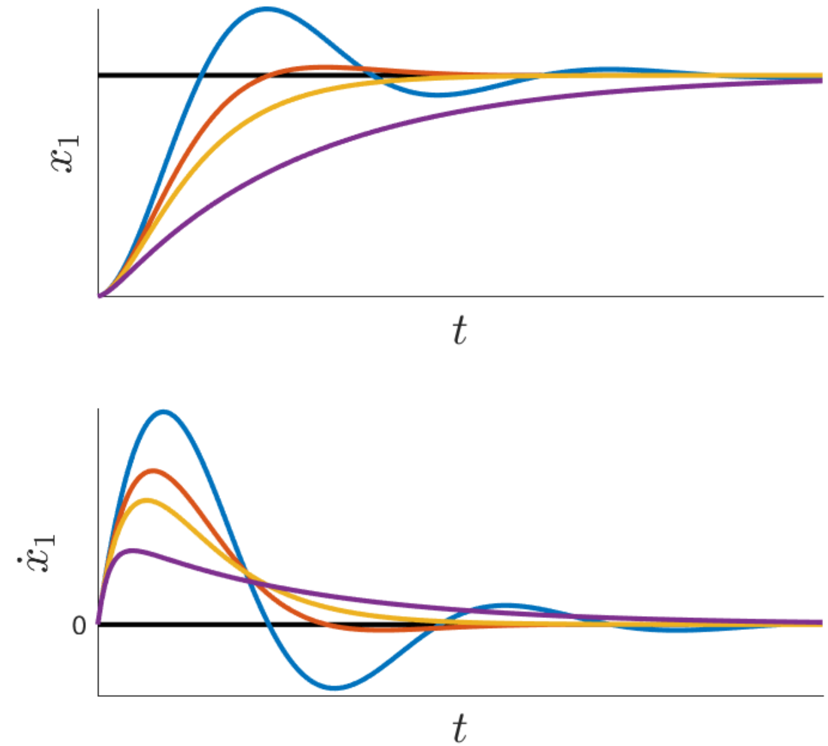
$$\ddot{\boldsymbol{x}} = k^{\mathcal{P}}(\hat{\boldsymbol{x}} - \boldsymbol{x}) - k^{\mathcal{V}}\dot{\boldsymbol{x}}$$

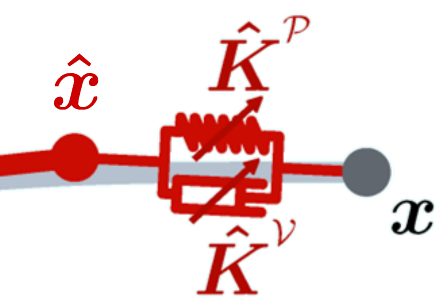
$k^{\mathcal{V}} = \frac{1}{2}\sqrt{2k^{\mathcal{P}}}$ (underdamped)

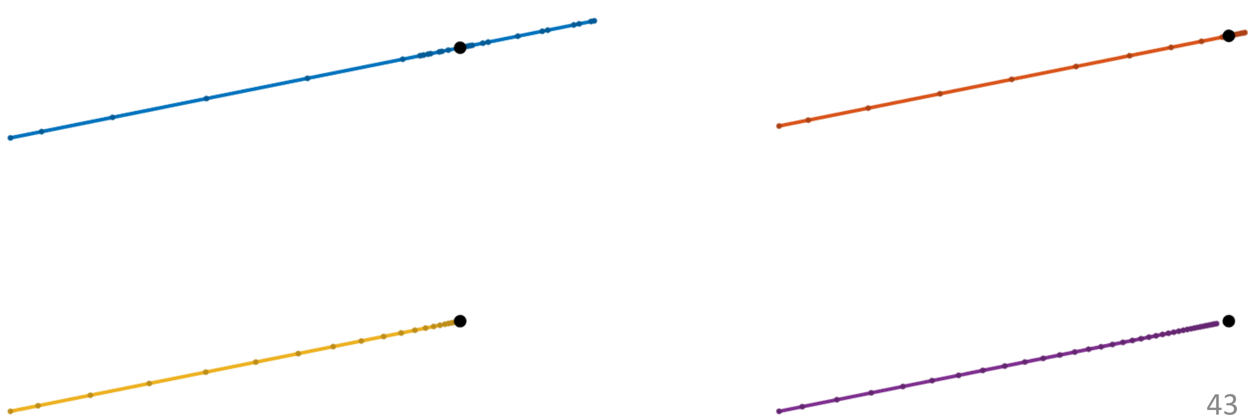
$k^{\mathcal{V}} = \sqrt{2k^{\mathcal{P}}}$ (ideally damped)

$k^{\mathcal{V}} = 2\sqrt{k^{\mathcal{P}}}$ (critically damped)

$k^{\mathcal{V}} = 4\sqrt{k^{\mathcal{P}}}$ (overdamped)



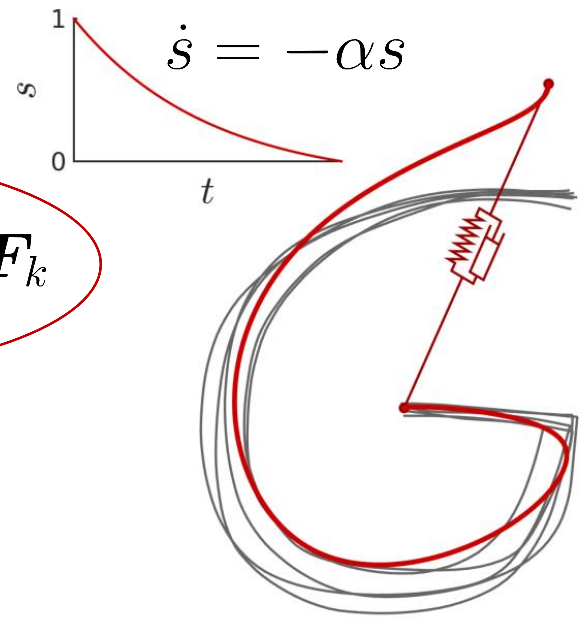




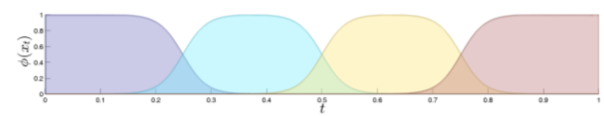
Dynamical movement primitives (DMP)

$$\ddot{\boldsymbol{x}} = k^{\mathcal{P}}(\boldsymbol{\mu}_T - \boldsymbol{x}) - k^{\mathcal{V}}\dot{\boldsymbol{x}} \ + \ \boldsymbol{f}(s)$$

$$\dot{s} = -\alpha s$$



$$\boldsymbol{f}(s) = \ s \sum_{k=1}^{K} \phi_k(s)\, \boldsymbol{F}_k$$



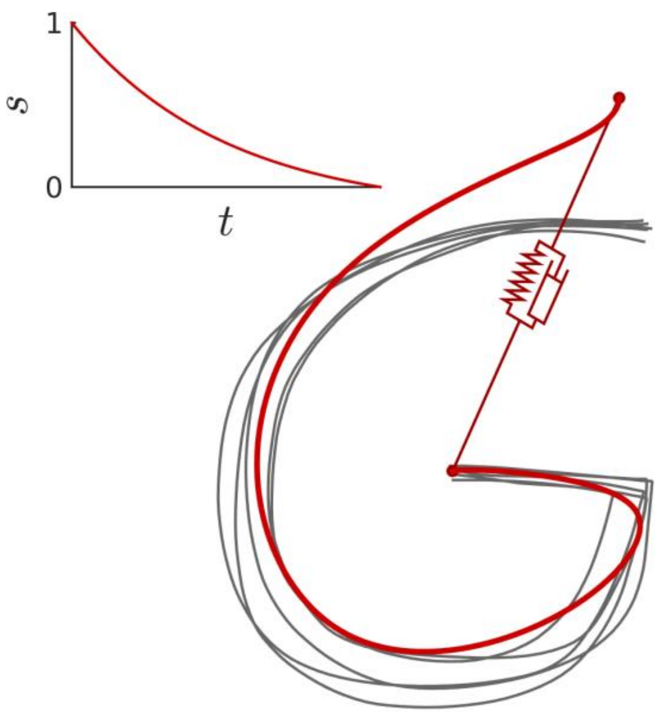
$$\boldsymbol{X}^{\mathcal{O}} = \begin{bmatrix} \ddot{\boldsymbol{x}}_1 - k^{\mathcal{P}}(\boldsymbol{\mu}_T - \boldsymbol{x}_1) + k^{\mathcal{V}}\dot{\boldsymbol{x}}_1 \\ \ddot{\boldsymbol{x}}_2 - k^{\mathcal{P}}(\boldsymbol{\mu}_T - \boldsymbol{x}_2) + k^{\mathcal{V}}\dot{\boldsymbol{x}}_2 \\ \vdots \\ \ddot{\boldsymbol{x}}_T - k^{\mathcal{P}}(\boldsymbol{\mu}_T - \boldsymbol{x}_T) + k^{\mathcal{V}}\dot{\boldsymbol{x}}_T \end{bmatrix} \quad \boldsymbol{X}^{\mathcal{I}} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_T \end{bmatrix}$$

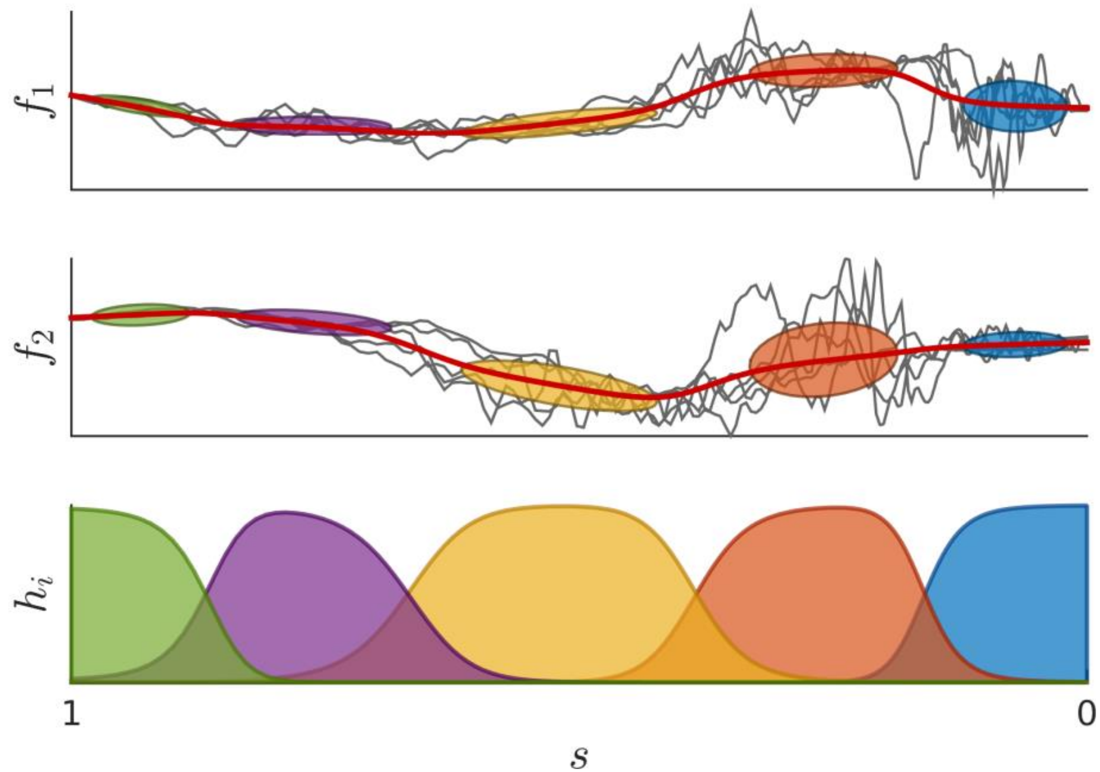
$$\boldsymbol{W}_k = \mathrm{diag}\Big(\phi_k(s_1), \phi_k(s_2), \dots, \phi_k(s_T)\Big)$$

$$\hat{\boldsymbol{F}}_k = (\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{W}_k\boldsymbol{X}^{\mathcal{I}})^{-1}\boldsymbol{X}^{\mathcal{I}\top}\boldsymbol{W}_k\,\boldsymbol{X}^{\mathcal{O}}$$

Dynamical movement primitives with GMR

Learning of $\mathcal{P}(s, \boldsymbol{x})$ and retrieval of $\mathcal{P}(\boldsymbol{x}|s)$





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

DMP

A. Ijspeert, J. Nakanishi and S. Schaal. Learning Control Policies For Movement Imitation and Movement recognition. NIPS'2003

A. Ijspeert, J. Nakanishi, P. Pastor, H. Hoffmann, and S. Schaal. Dynamical movement primitives: Learning attractor models for motor behaviors. Neural Computation, 25(2):328–373, 2013