EE613
Machine Learning for Engineers

NONLINEAR REGRESSION I

Sylvain Calinon
Robot Learning & Interaction Group
Idiap Research Institute
Dec. 13, 2017

Outline

- Properties of multivariate Gaussian distributions
  Matlab codes: *demo_Gaussian_product01.m, demo_Gaussian_conditioning01.m, demo_Gaussian_conditioning_noisyInput01.m, demo_Gaussian_lawTotalCov01.m*

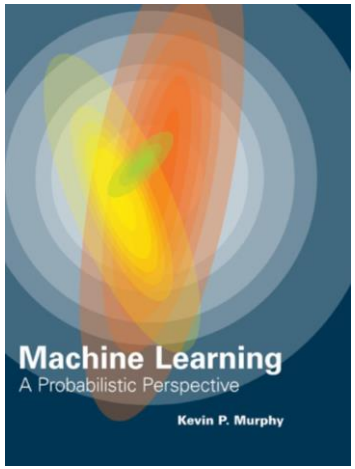
- Locally weighted regression (LWR)
  Matlab code: *demo_LWR01.m,*

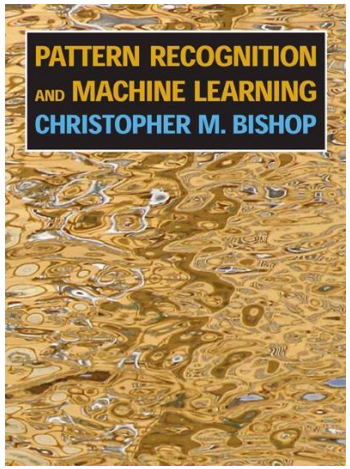
- Dynamical movement primitives (DMP)
  Matlab code: *demo_DMP01.m*

- Gaussian mixture regression (GMR)
  Matlab codes: *demo_GMR01.m, demo_GMR_polyFit01.m, demo_DMP_GMR01.m*

Properties of multivariate Gaussian distributions

Matlab codes: demo_Gaussian_product01.m, demo_Gaussian_conditioning01.m, demo_Gaussian_conditioning_noisyInput01.m, demo_Gaussian_lawTotalCov01.m

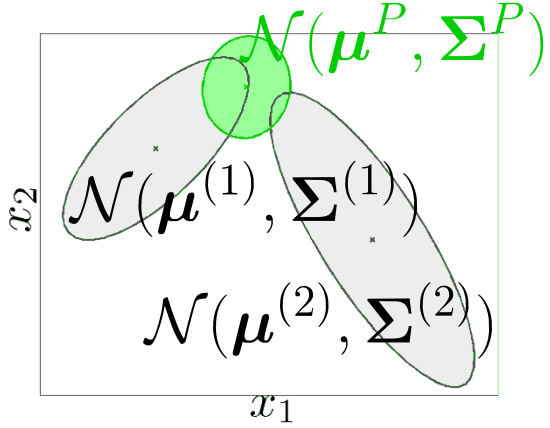


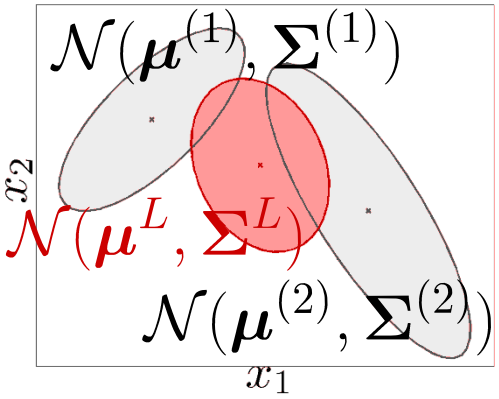


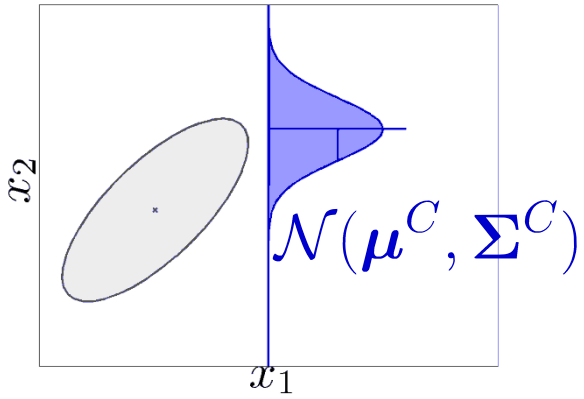
The Matrix Cookbook

Kaare Brandt Petersen
Michael Syskind Pedersen

Some very useful properties







Product of Gaussians:

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

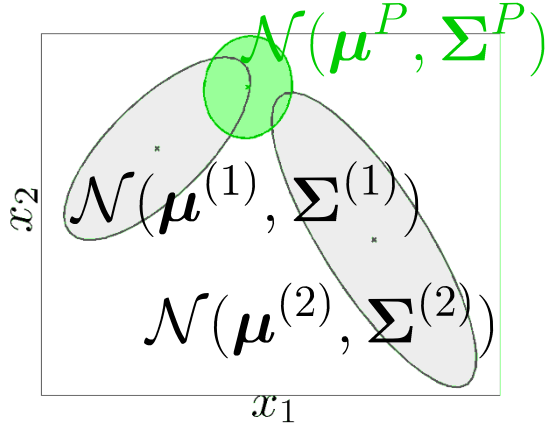
Linear combination:

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

Conditional probability:

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

Product of Gaussians

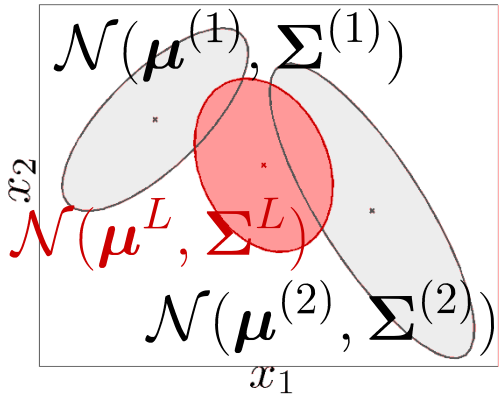


The product of two Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ and $\mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$ is defined by

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

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

Linear combination



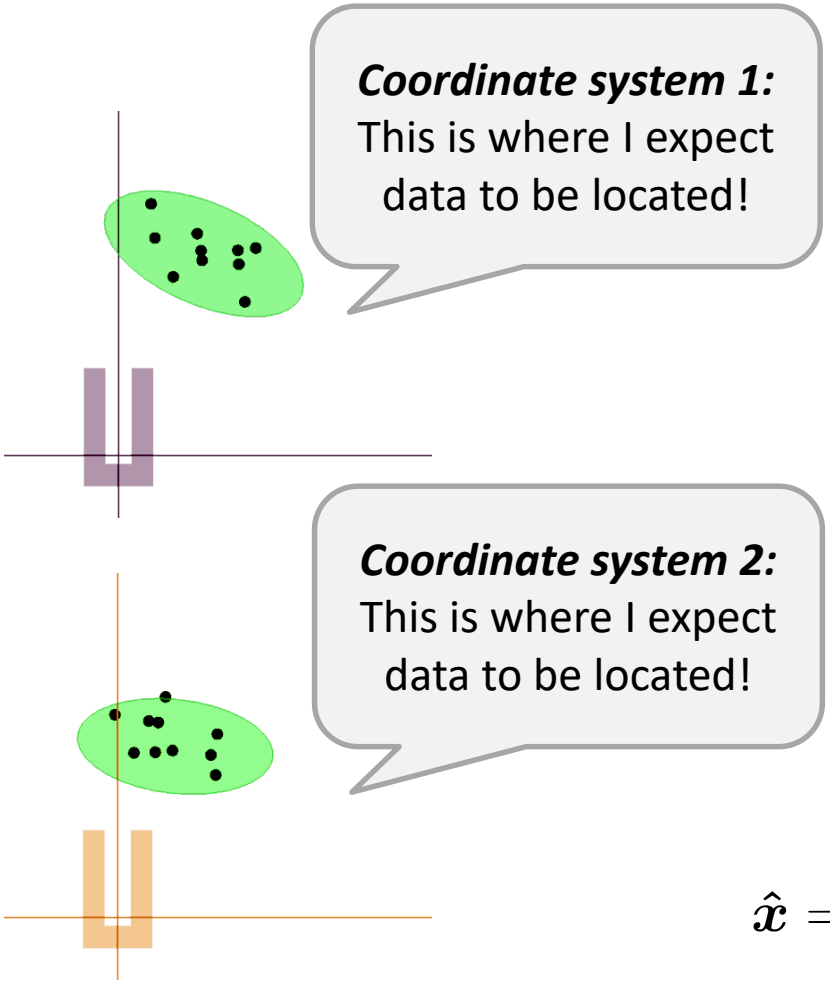
If $\boldsymbol{x}^{(1)} \sim \mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ and $\boldsymbol{x}^{(2)} \sim \mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma}^{(2)})$, the linear transformation $\boldsymbol{A}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{x}^{(2)}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

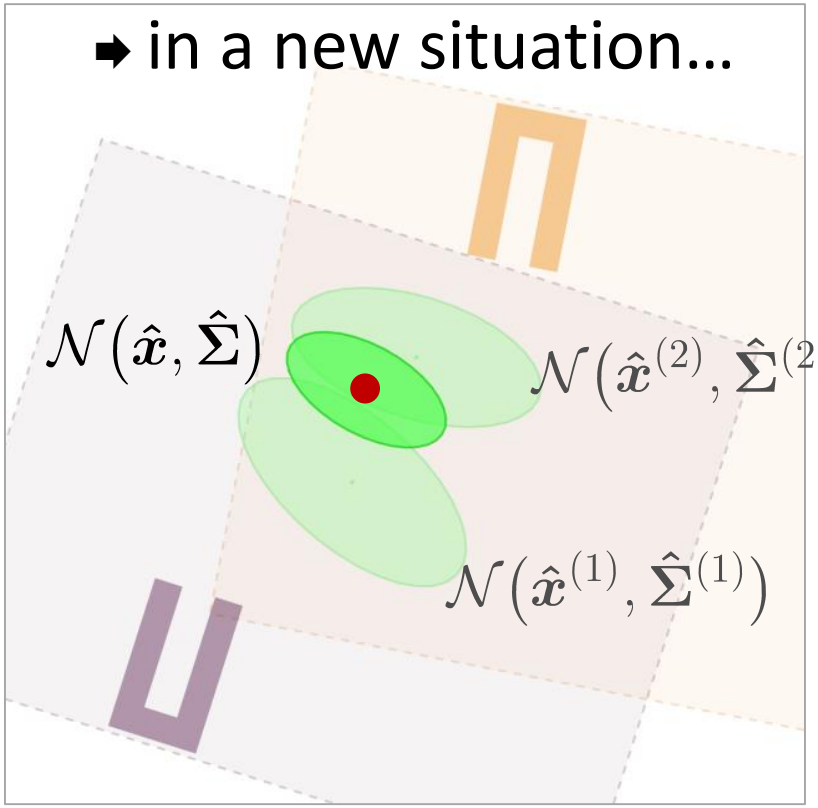
$$\boldsymbol{A}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{A}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{c} \sim \mathcal{N}(\boldsymbol{\mu}^{L}, \boldsymbol{\Sigma}^{L}),$$

with

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

Example: Fusion of sensor/control information





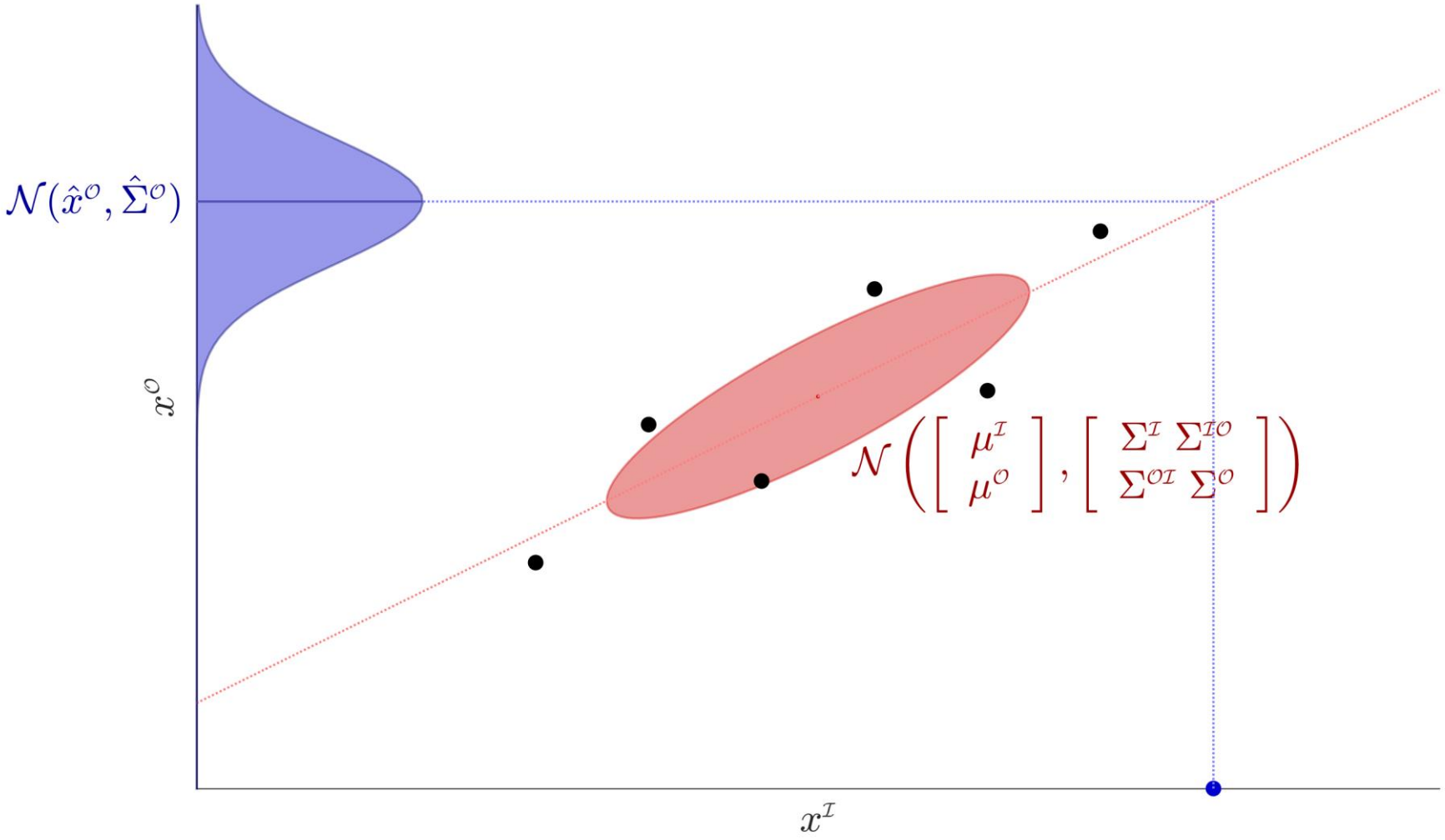
$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \sum_{j=1}^{2} \left(\boldsymbol{x} - \hat{\boldsymbol{x}}^{(j)}\right)^{\top} \hat{\boldsymbol{\Sigma}}^{(j)-1} \left(\boldsymbol{x} - \hat{\boldsymbol{x}}^{(j)}\right)$$

→ Product of linearly transformed Gaussians

[Calinon, S. (2016). A Tutorial on Task-Parameterized Movement Learning and Retrieval. Intelligent Service Robotics 9(1):1–29]

Conditional probability

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



→ Linear regression from joint distribution

Conditional probability

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

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

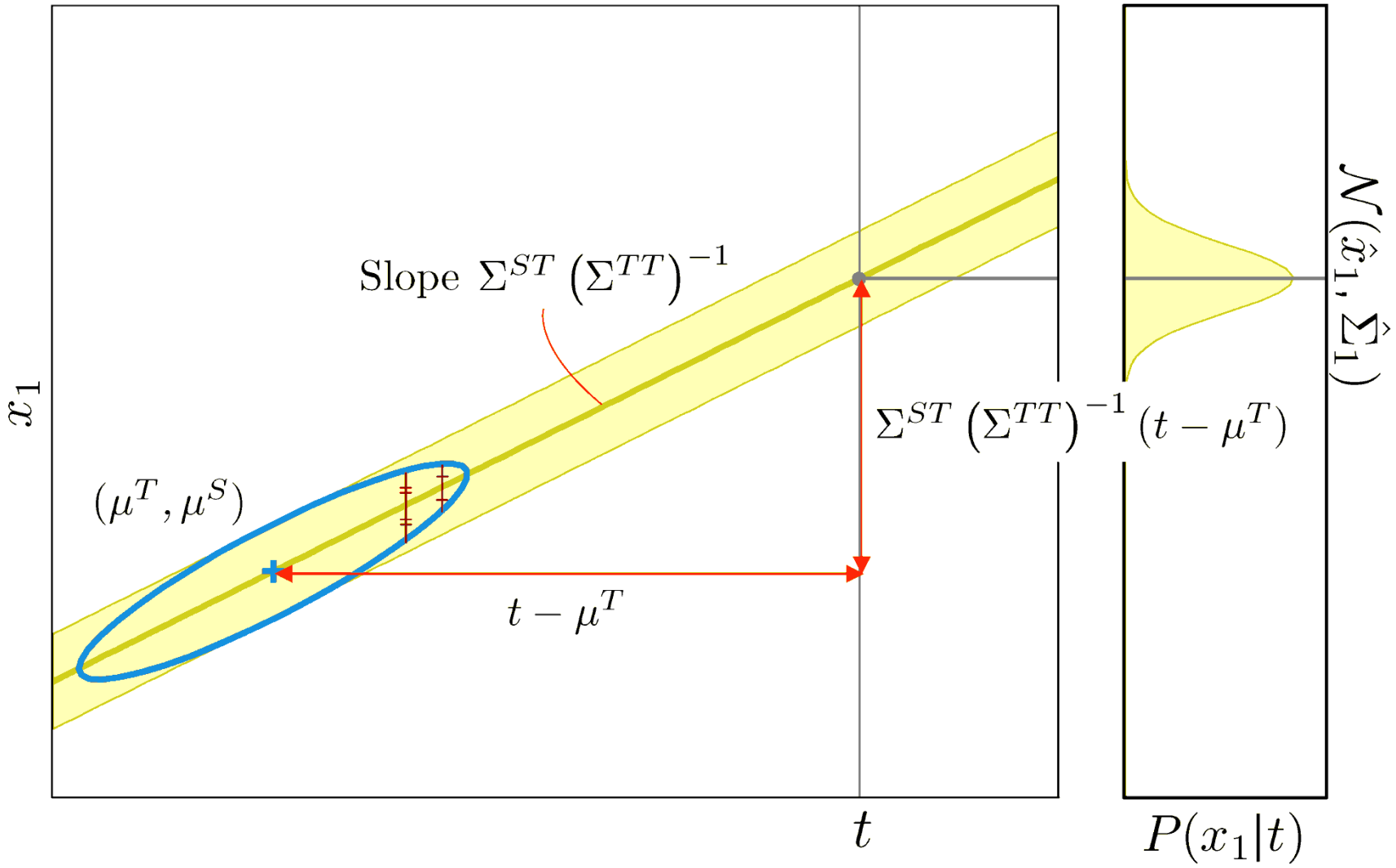
If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we have that $\boldsymbol{x}^{\mathcal{O}} | \boldsymbol{x}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\right)$,
with parameters

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

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 probability - Geometric interpretation



Conditional probability - Proof

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

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

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

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 probability - Proof

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

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

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

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

The above result can be shown by using a LDU decomposition of $\boldsymbol{\Sigma}$, 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 probability - Proof

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

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

With this partitioning, we can see that

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

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 probability - Proof

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

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

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

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

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

By using the Schur decomposition, we can see that

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

Conditional probability - Proof

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

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

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 many cases, evaluating the conditional distribution with precision matrices is computationally more efficient than with covariance matrices.

Conditional probability - Summary

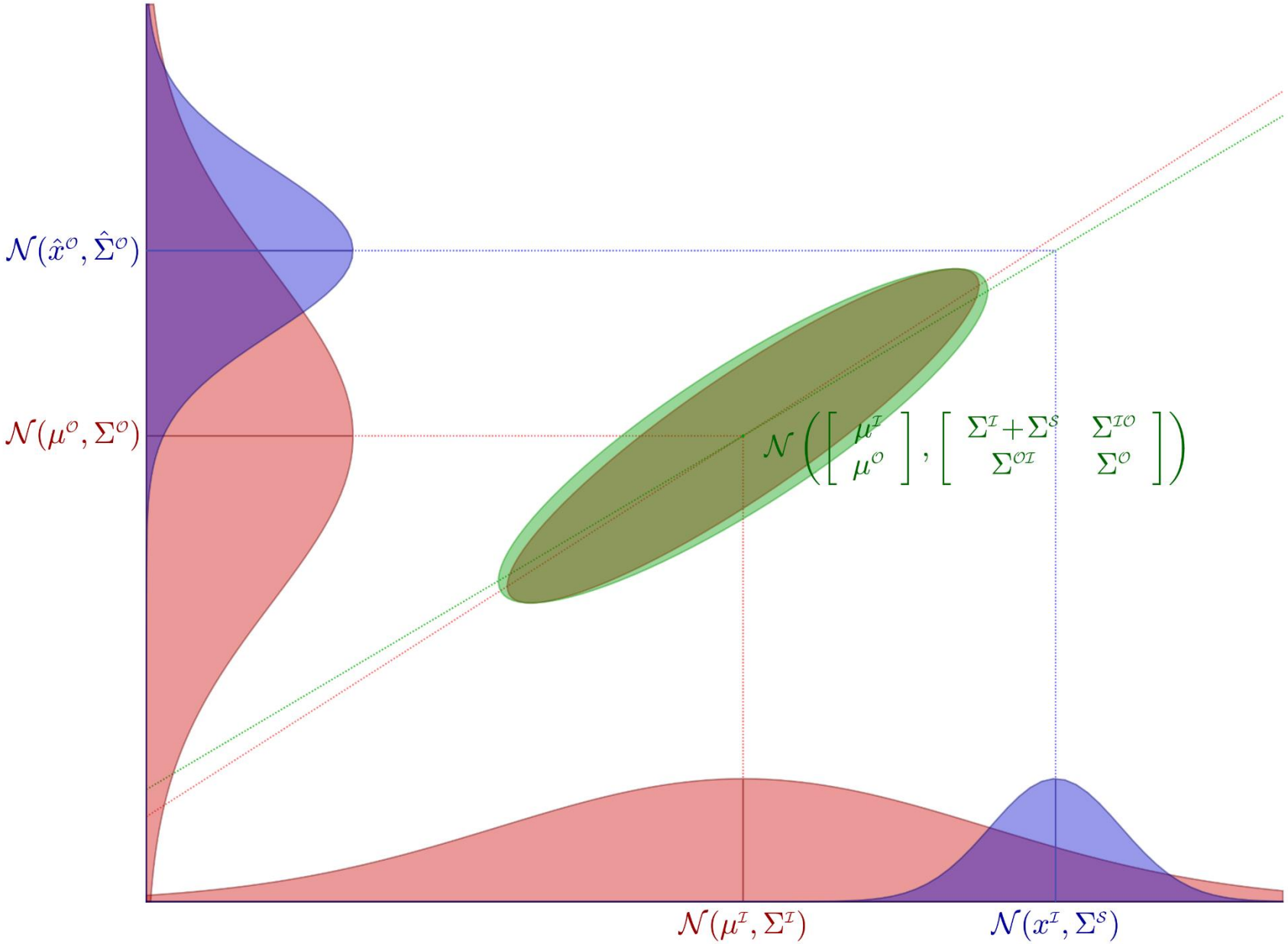
If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we have that $\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\right)$, with parameters

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

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 conditioning with uncertain inputs



Gaussian conditioning with uncertain inputs

For inputs corrupted by noise $\boldsymbol{\epsilon}^{\mathcal{I}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{\mathcal{S}})$, we have that $\boldsymbol{x}^{\mathcal{O}} | \boldsymbol{x}^{\mathcal{I}} + \boldsymbol{\epsilon}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\right)$ with parameters

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

This can for example be shown by redefining the joint distribution $\boldsymbol{\Sigma}$ with

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

and applying the same formulas as for standard Gaussian conditioning.

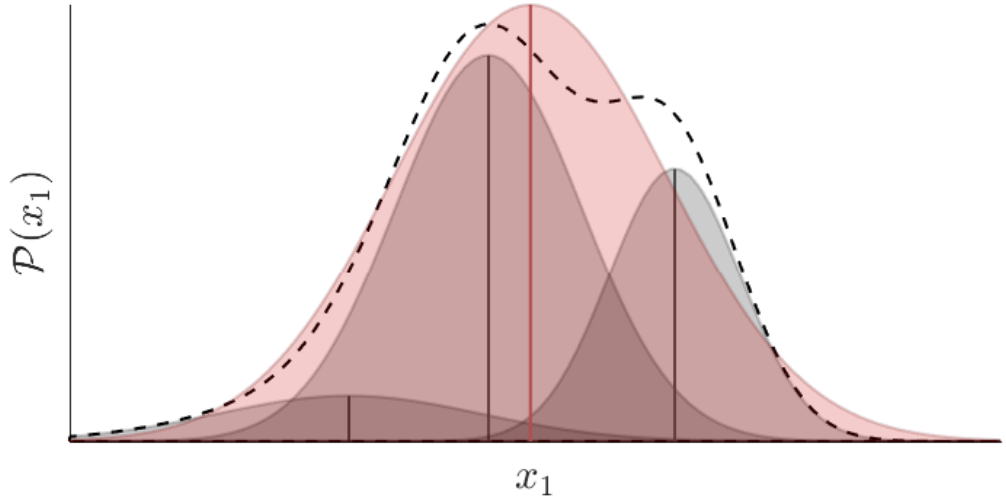
Gaussian estimate of a mixture of Gaussians

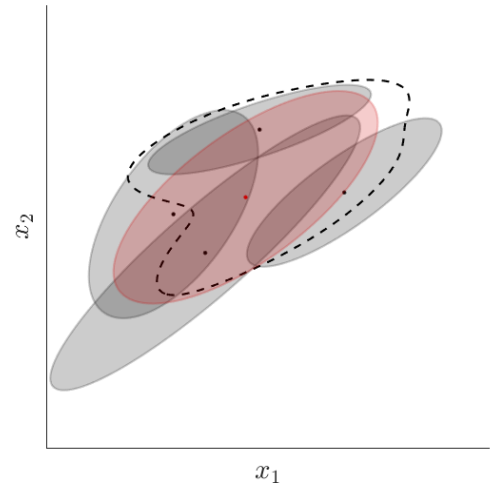
We can approximate a mixture of Gaussians $\sum_{i=1}^{K} h_i \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ with a single Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, by **moment matching of the means (first moments) and covariances (second moments)** with

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

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

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





Gaussian estimate of a mixture - Proof

The result can be demonstrated by developing the expressions

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

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

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

$\mathbb{E}(\boldsymbol{x}) = \int \boldsymbol{x} \, \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \, d\boldsymbol{x}$

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

Gaussian estimate of a mixture of Gaussians

By noting that

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

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

$$\mathbb{E}(\boldsymbol{x}\boldsymbol{x}^\top) = \int \boldsymbol{x}\boldsymbol{x}^\top \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\, d\boldsymbol{x}$$

the covariance is then computed as

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

Locally weighted regression (LWR)

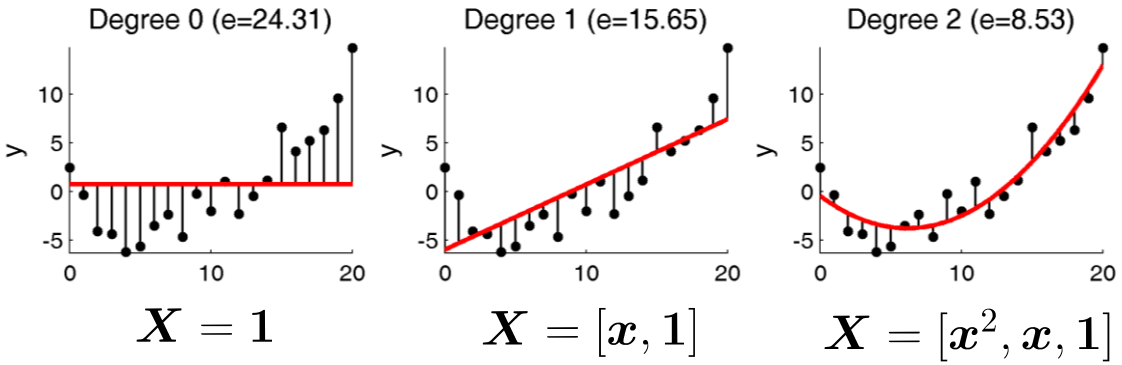
Matlab codes: demo_LWR01.m, demo_DMP01.m

[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]

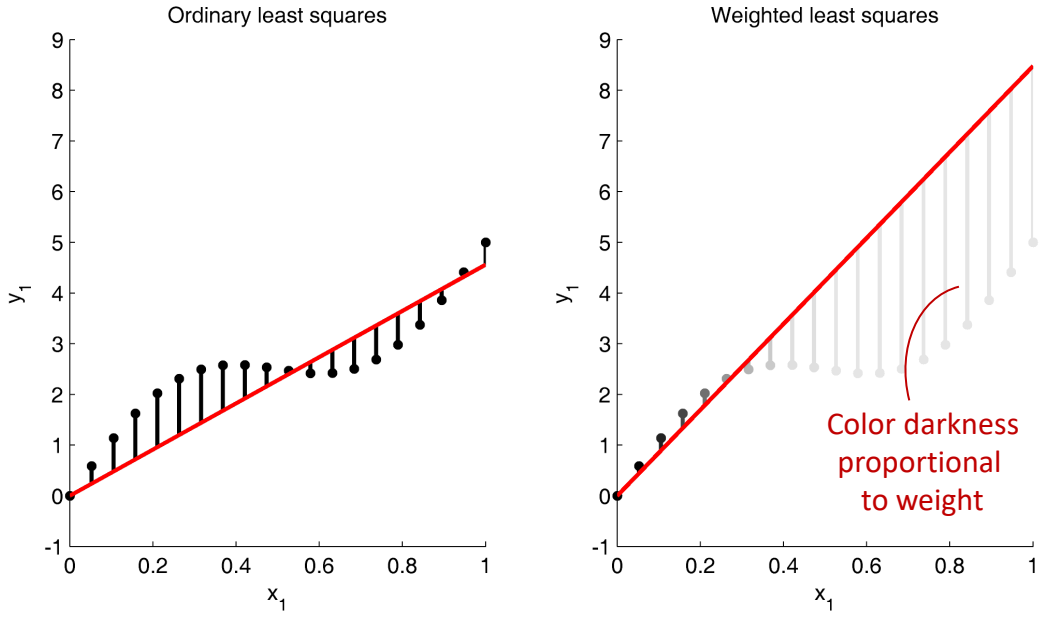
Previous lecture on linear regression

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



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

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



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 $\{\boldsymbol{X}^{\mathcal{I}}, \boldsymbol{X}^{\mathcal{O}}\}$.

It aims at splitting a nonlinear problem so that it can be solved locally by linear regression.

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.

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.

An associated diagonal matrix

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

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

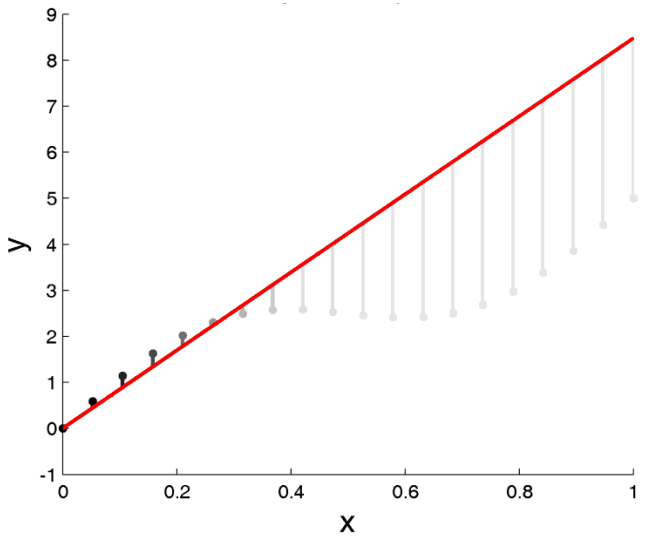
$$\boldsymbol{X}^{\mathcal{O}} = \sum_{k=1}^{K} \boldsymbol{W}_k \boldsymbol{X}^{\mathcal{I}} \hat{\boldsymbol{A}}_k.$$

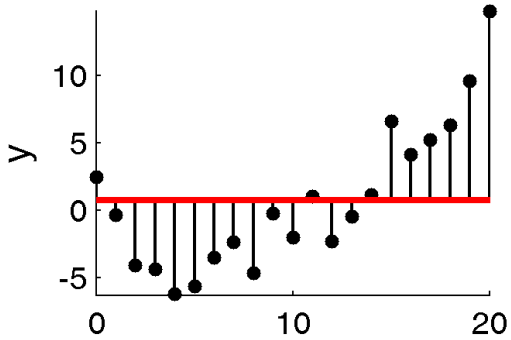
$$\boldsymbol{X}^{\mathcal{I}} = [t_1, t_2, \ldots, t_N]^\top$$
$$\hat{\boldsymbol{A}}_k = (\boldsymbol{X}^{\mathcal{I}\top} \boldsymbol{W}_k \boldsymbol{X}^{\mathcal{I}})^{-1} \boldsymbol{X}^{\mathcal{I}\top} \boldsymbol{W}_k \boldsymbol{X}^{\mathcal{O}}$$

Multiple variants exist, including online estimation within a recursive formulation [Schaal'98], Bayesian treatments of LWR [Ting'08], and also extensions such as **locally weighted projection regression** (LWPR) that exploit partial least squares to handle redundant and/or irrelevant inputs, with an online algorithm to estimate the model parameters [Vijayakumar'05].

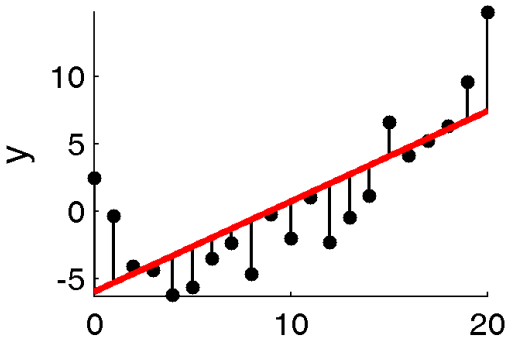
Locally weighted regression (LWR)

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



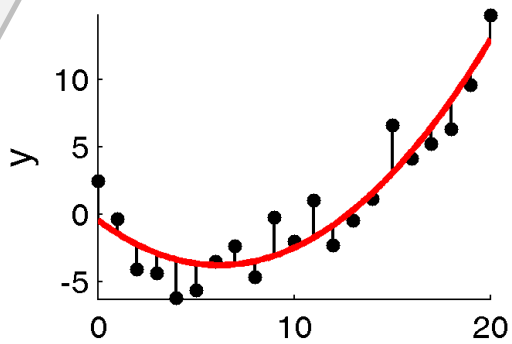


$X = 1$



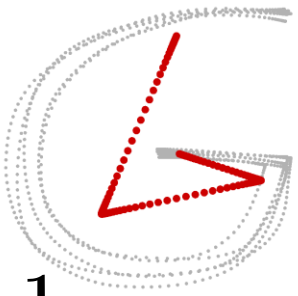
$X = [x, 1]$

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

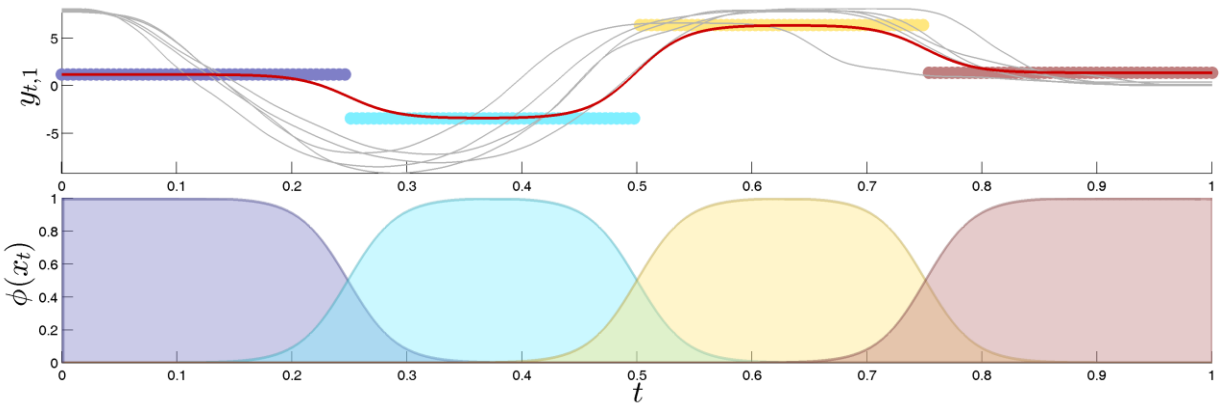


$X = [x^2, x, 1]$

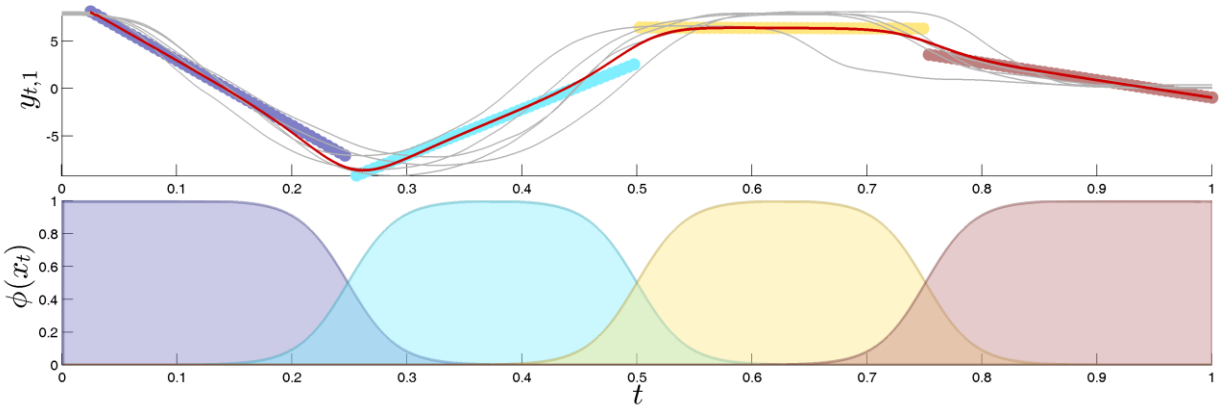
Locally weighted regression (LWR)



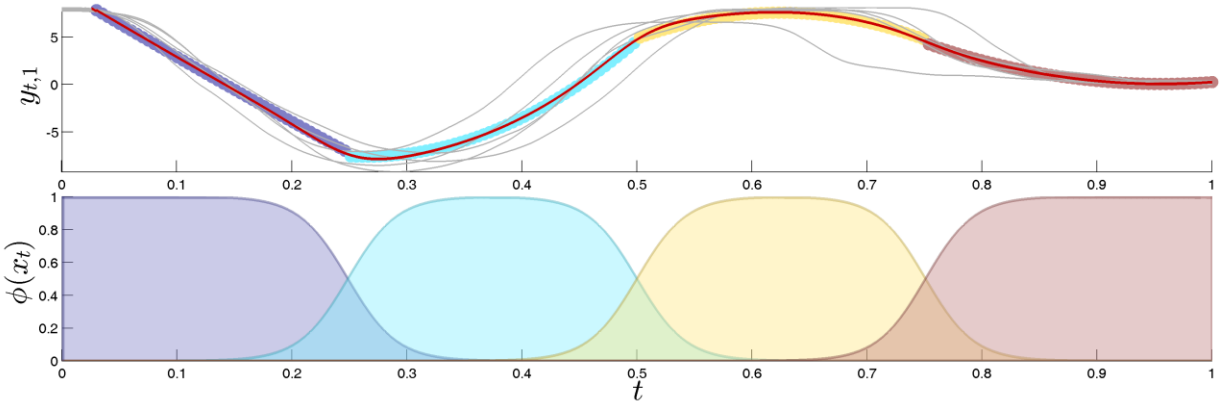
$$X = 1$$



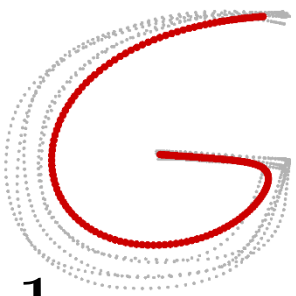
$$X = [x, 1]$$



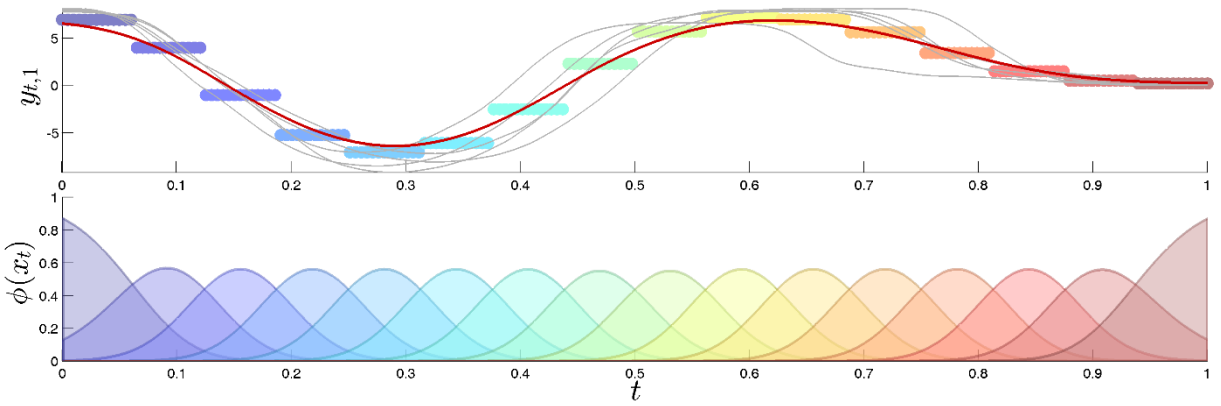
$$X = [x^2, x, 1]$$



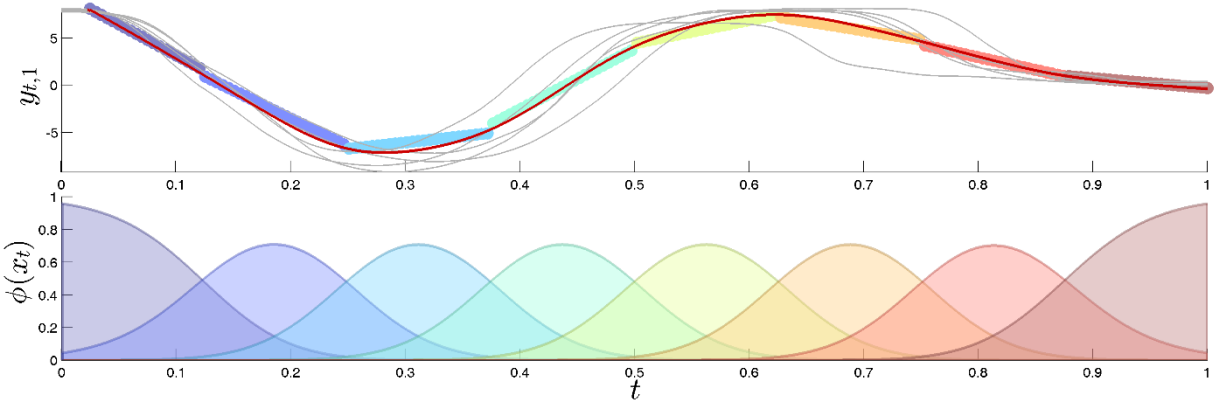
Locally weighted regression (LWR)



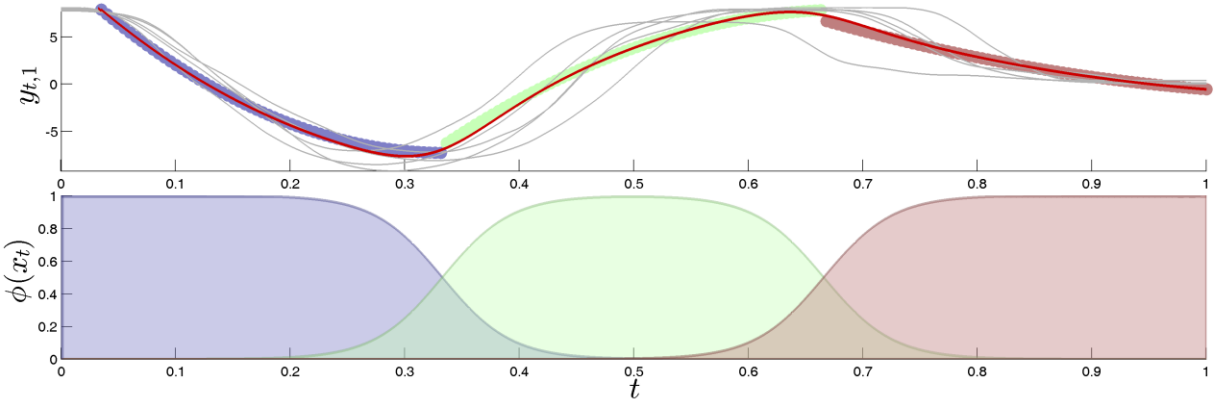
$$X = 1$$



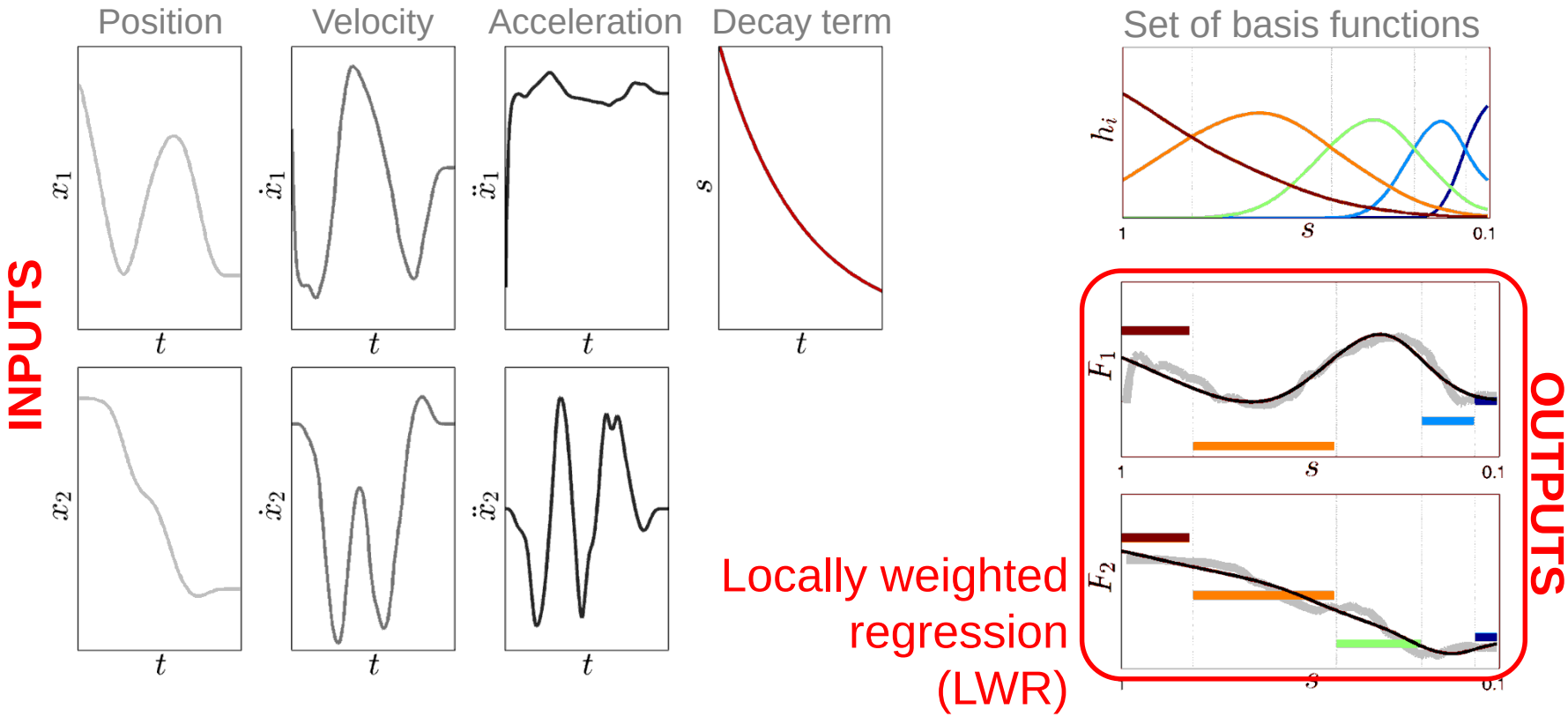
$$X = [x, 1]$$



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



Dynamical movement primitives (DMP)



Locally weighted regression (LWR)

$$\tau\ddot{x} = \kappa^{P}\left[x_T - x\right] - \kappa^{V}\dot{x} + f(s), \quad f(s) = s\sum_{i=1}^{K} h_i(s)\, f_i$$

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

[Ijspeert, Nakanishi and Schaal, NIPS'2003]

[Ijspeert, Nakanishi, Pastor, Hoffmann and Schaal, Neural Computation 25(2), 2013]

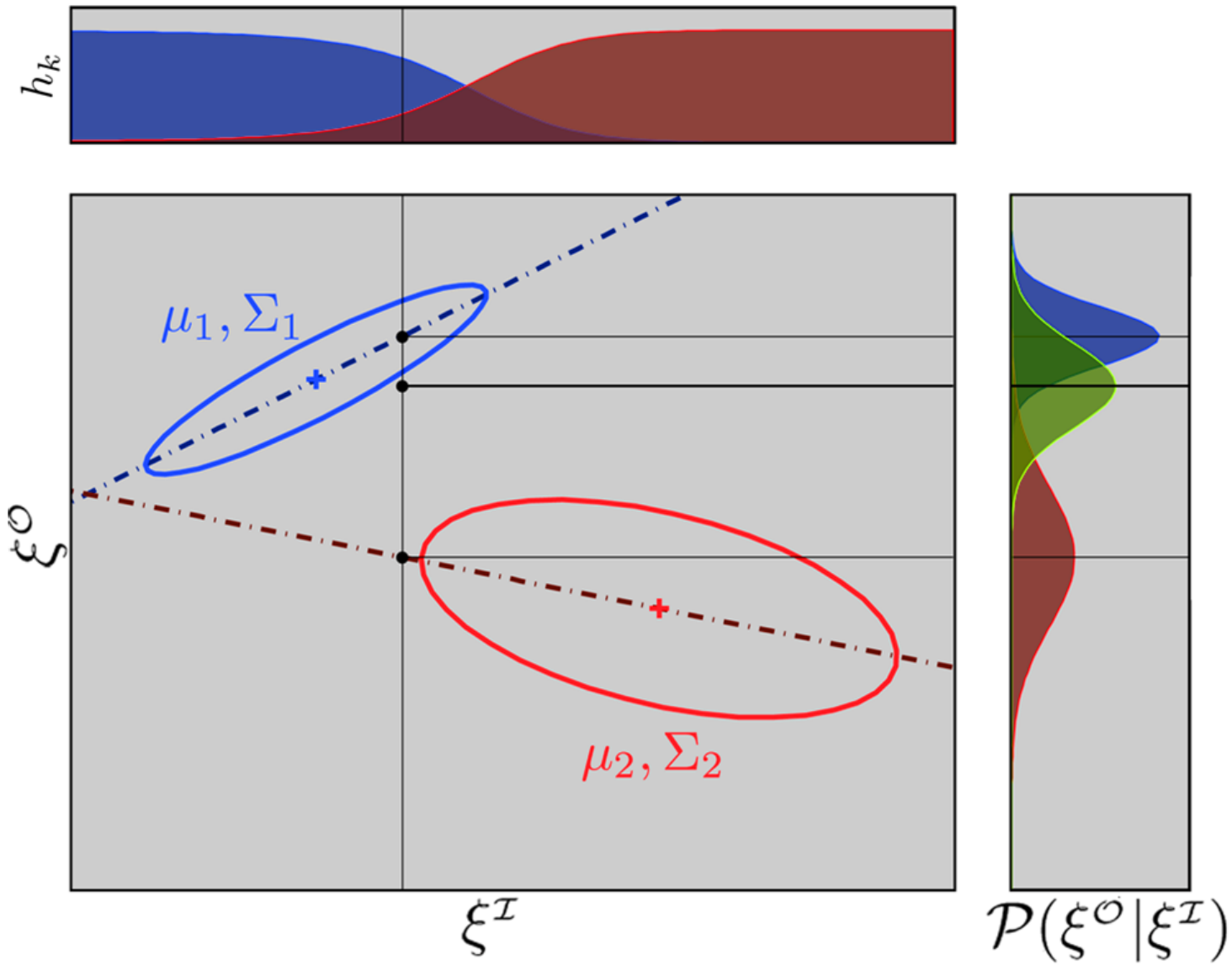
Gaussian mixture regression (GMR)

Matlab codes: demo_GMR01.m demo_GMR_polyFit01.m, demo_DMP_GMR01.m

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

[S. Calinon S. A tutorial on task-parameterized movement learning and retrieval. Intelligent Service Robotics 9(1):1–29, 2016]

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)

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

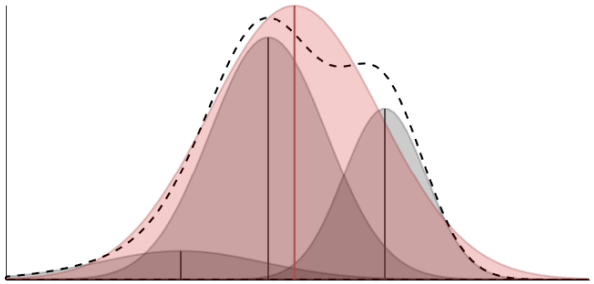
$\mathcal{P}(\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}})$ can be computed as the multimodal conditional distribution

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

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)



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

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

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

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

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

Gaussian mixture regression (GMR) - Proof

The above can be demonstrated by computing

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

The conditional mean can be computed as

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

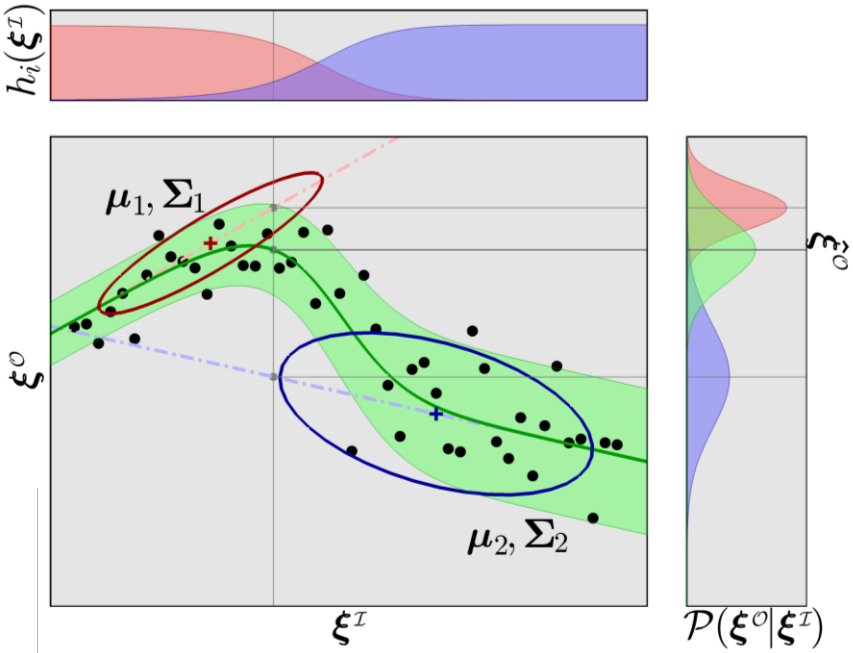
In order to evaluate the covariance, we first note that

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

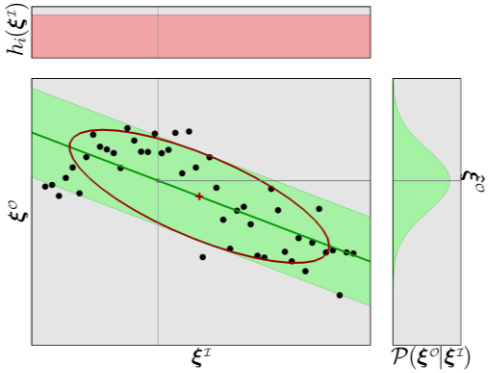
We then have

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

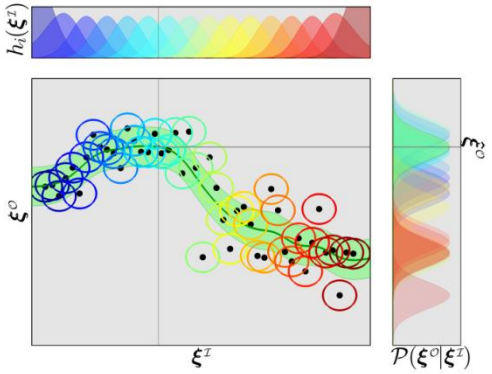
Gaussian mixture regression (GMR)



Least squares linear regression



Nadaraya-Watson kernel regression



GMR can cover a large spectrum of regression mechanisms

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

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

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

GMR with uncertain inputs

For inputs with noise $\boldsymbol{\epsilon}^{\mathcal{I}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{\mathcal{S}})$, we have that $\boldsymbol{x}^{\mathcal{O}}|\boldsymbol{x}^{\mathcal{I}}+\boldsymbol{\epsilon}^{\mathcal{I}} \sim \mathcal{N}\left(\hat{\boldsymbol{\mu}}^{\mathcal{O}}, \hat{\boldsymbol{\Sigma}}^{\mathcal{O}}\right)$, with parameters

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

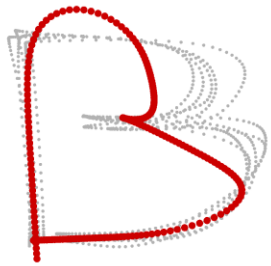
$$\hat{\boldsymbol{\Sigma}}^{\mathcal{O}} = \sum_{i=1}^{K} h_i \left(\hat{\boldsymbol{\Sigma}}_i^{\mathcal{O}} + \hat{\boldsymbol{\mu}}_i^{\mathcal{O}} \; \hat{\boldsymbol{\mu}}_i^{\mathcal{O}\top}\right) - \hat{\boldsymbol{\mu}}^{\mathcal{O}} \hat{\boldsymbol{\mu}}^{\mathcal{O}\top},$$

computed with

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

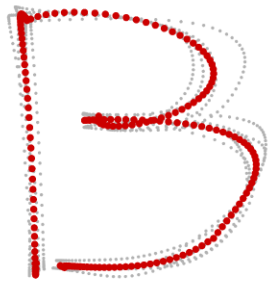
$$\text{and} \quad h_i = \frac{\pi_i \, \mathcal{N}(\boldsymbol{x}^{\mathcal{I}}|\; \boldsymbol{\mu}_i^{\mathcal{I}}, \boldsymbol{\Sigma}_i^{\mathcal{I}} + \boldsymbol{\Sigma}^{\mathcal{S}})}{\sum_k^K \pi_k \, \mathcal{N}(\boldsymbol{x}^{\mathcal{I}}|\; \boldsymbol{\mu}_k^{\mathcal{I}}, \boldsymbol{\Sigma}_k^{\mathcal{I}}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

GMR for smooth piecewise polynomial fitting

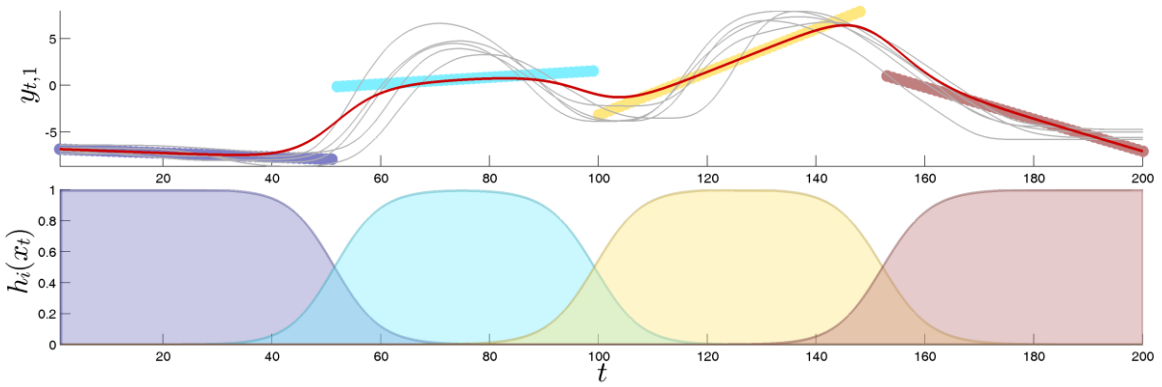


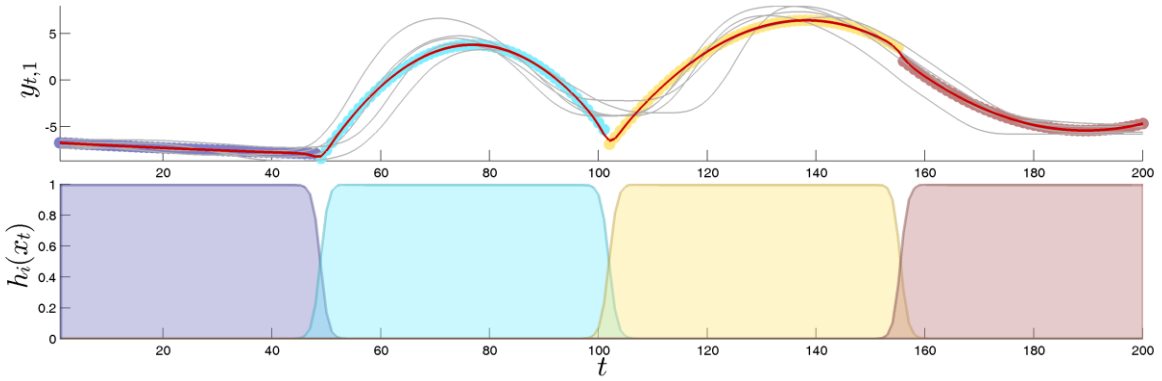
$$\boldsymbol{X} = \boldsymbol{x}$$

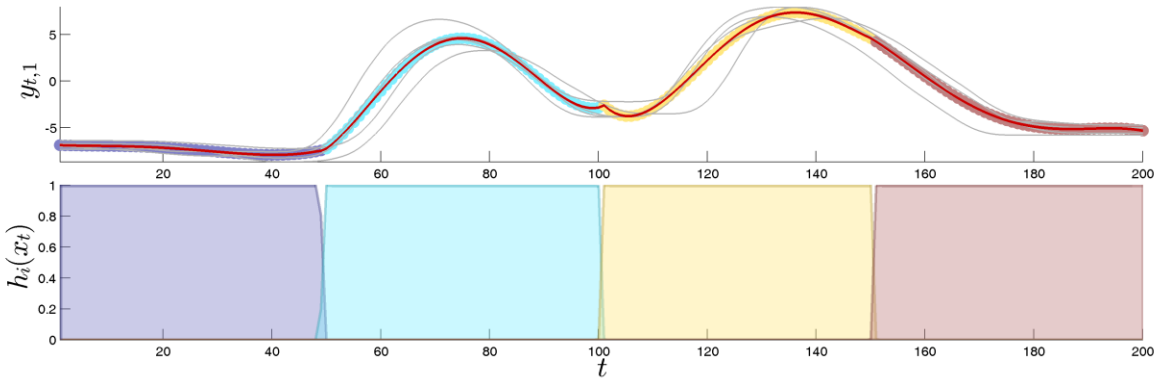
$$\boldsymbol{X} = [\boldsymbol{x}^2, \boldsymbol{x}]$$



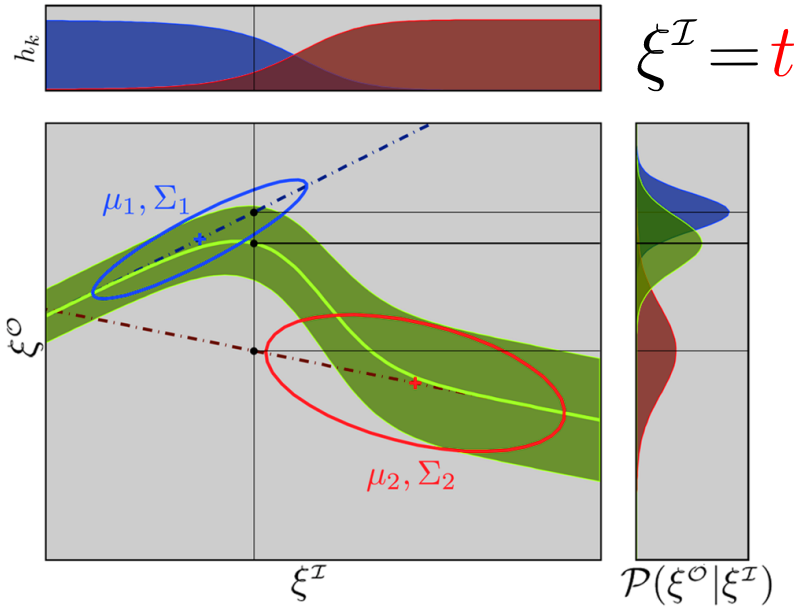
$$\boldsymbol{X} = [\boldsymbol{x}^3, \boldsymbol{x}^2, \boldsymbol{x}]$$





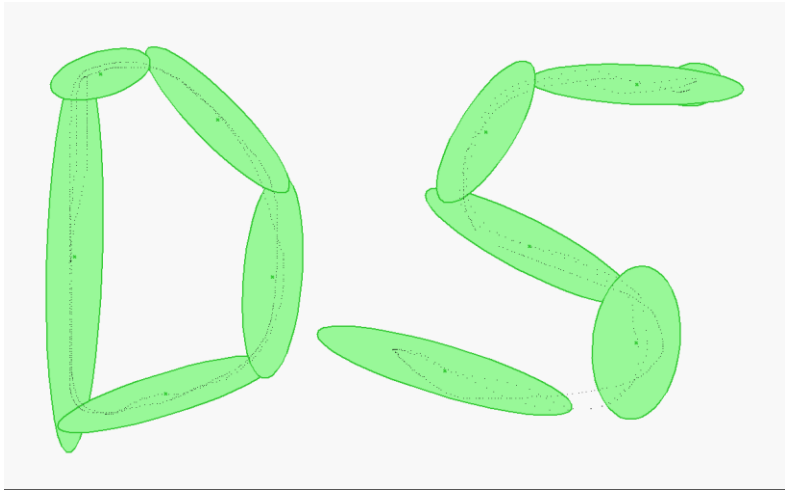


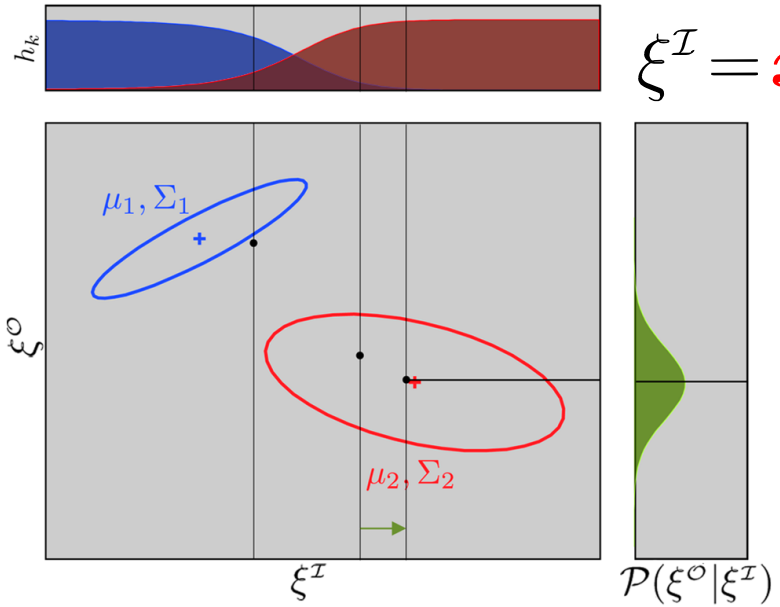
Gaussian mixture regression - Examples



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

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





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

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

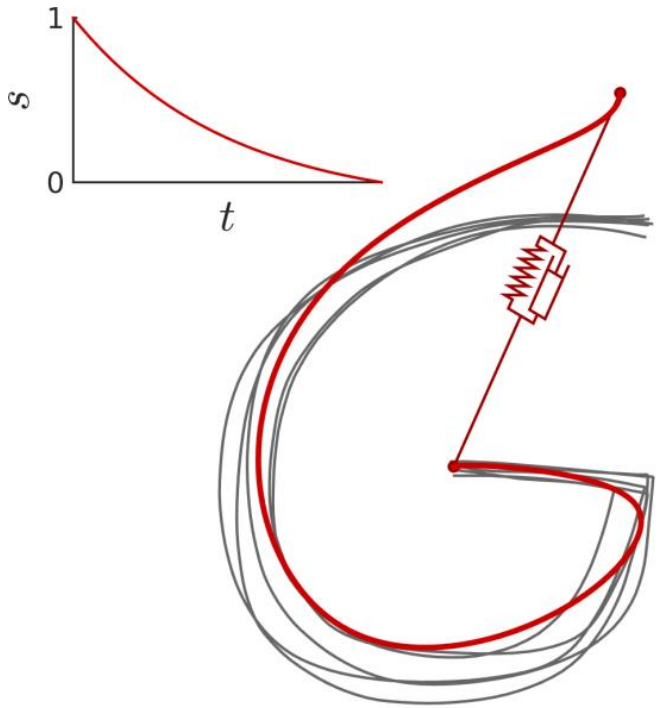
[Hersch, Guenter, Calinon and Billard, IEEE Trans. on Robotics 24(6), 2008]

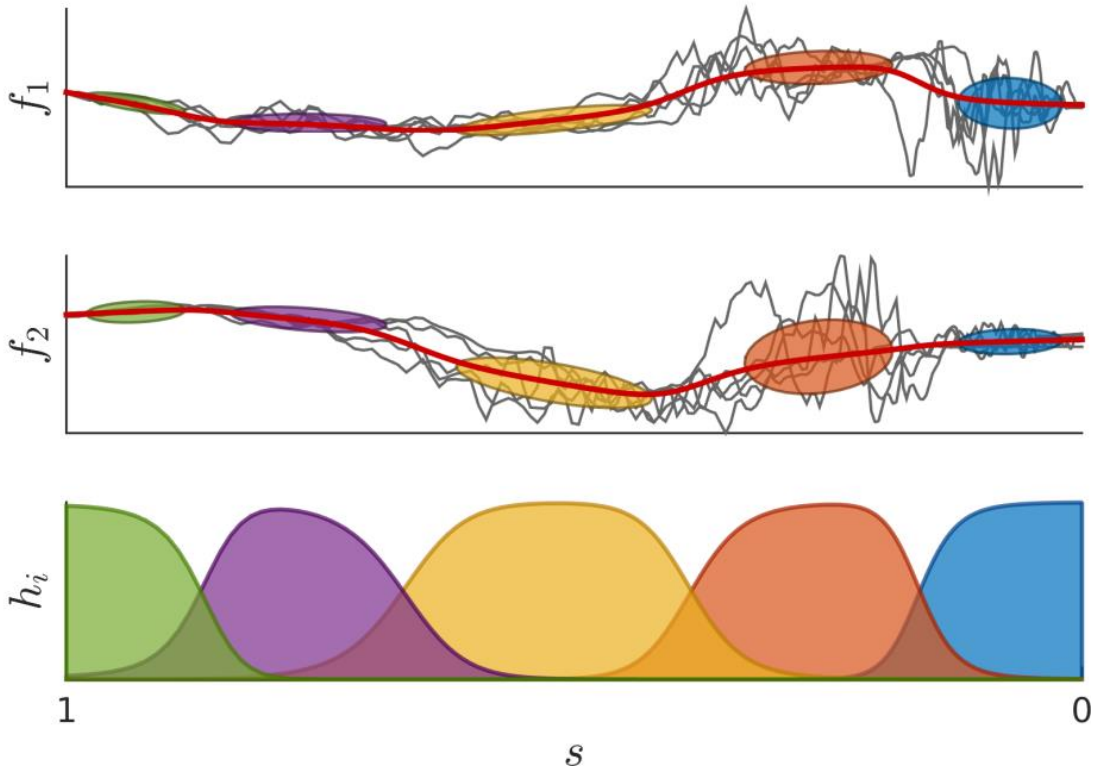
With quadratic programming solver:
(maximizing log-likelihood s.t. stability constraints)

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

Dynamical movement primitives with GMR

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





Main references

Regression

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

LWR

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

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

Recursive formulation of LWR

S. Schaal and C.G. Atkeson. Constructive incremental learning from only local information. Neural Computation 10(8):2047–2084, 1998.

Bayesian formulation of LWR

J. Ting, M. Kalakrishnan, S. Vijayakumar and S. Schaal. Bayesian kernel shaping for learning control. In: Advances in Neural Information Processing Systems (NIPS), pp 1673–1680, 2008.

Main references

DMP

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

LWPR

S. Vijayakumar, A. D'souza and S. Schaal. Incremental online learning in high dimensions. Neural Computation 17(12):2602–2634, 2005

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 S. A tutorial on task-parameterized movement learning and retrieval. Intelligent Service Robotics 9(1):1–29, 2016