10-301/601: Introduction to Machine Learning Lecture 33 – Gaussian Processes

Henry Chai

6/17/25

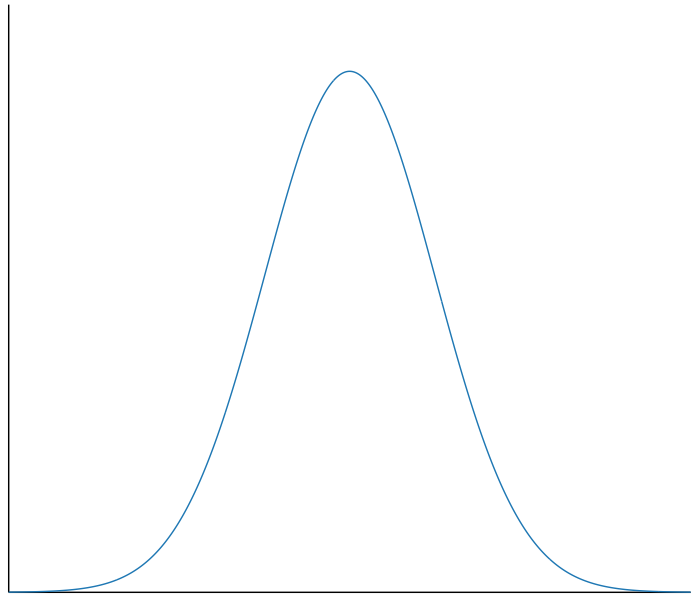
Front Matter

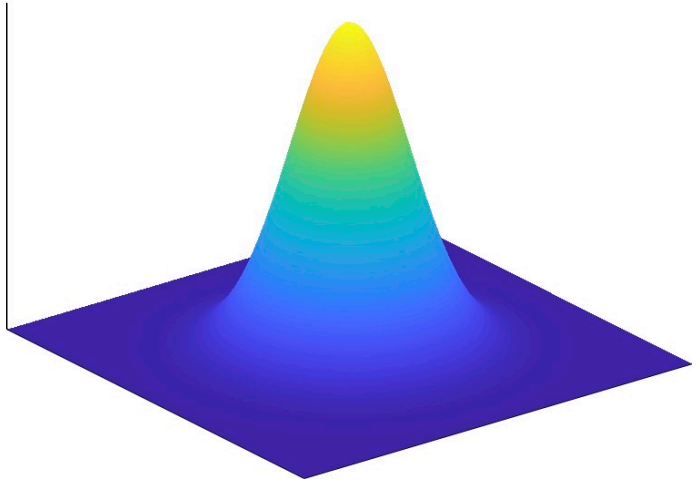
- Announcements
  - HW8 released 6/13, due 6/17 (today!) at 11:59 PM
  - Final on 6/20 (next Friday) at **8:30 AM** in BH A36 (here!)
    - **We will not use the full 3-hour window**
    - All topics from Lectures 17 to 30 are in-scope
    - **The final is *not* cumulative:** pre-midterm content may be referenced but will not be the *primary* focus of any question
    - You are allowed to bring one letter-size sheet of notes; you may put *whatever* you want on *both sides*

Gaussians

(Univariate) Gaussians:

$$x \sim \mathcal{N}(x; \mu = 0, \sigma^2 = 1)$$

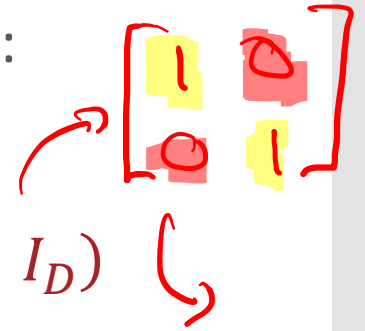




- Multivariate Gaussians:

$$\boldsymbol{x} = [x_1, \ldots, x_D]^T$$

$$\sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu} = \mathbf{0}_D, \Sigma = I_D)$$



covariance between x_1 & x_2

Some fun facts about Gaussians

- Closure under linear transformations:

if $\vec{x} \sim N(\vec{x}; \vec{\mu}, \Sigma)$

then $A\vec{x} + \vec{b} \sim N(A\vec{\mu} + \vec{b}, A\Sigma A^T)$

- Closure under addition:

if $\vec{x} \sim N(\vec{x}; \vec{\mu}, \Sigma)$ and $\vec{y} \sim N(\vec{y}; \vec{m}, S)$

then $\vec{x} + \vec{y} \sim N(\vec{\mu} + \vec{m}, \Sigma + S)$

- Closure under conditioning:

if $\vec{x} = \begin{bmatrix} \vec{x}_1 \\ \vec{x}_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \vec{x}_1 \\ \vec{x}_2 \end{bmatrix}; \begin{bmatrix} \vec{\mu}_1 \\ \vec{\mu}_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$

then $\vec{x}_1 \mid \vec{x}_2 = \vec{c} \sim N\left(\vec{x}_1; \vec{\mu}_1 + \Sigma_{12}\Sigma_{22}^{-1}(\vec{c} - \vec{\mu}_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}\right)$

Outline

Gaussian process =

Bayesian linear regression + Kernels

Outline

Gaussian process =

Bayesian linear regression + Kernels

Recall: Maximum a Posteriori (MAP) Estimation

- Insight: sometimes we have *prior* information we want to incorporate into parameter estimation
- Idea: use Bayes rule to reason about the *posterior* distribution over the parameters
  - MLE finds $\hat{\theta} = \underset{\theta}{\text{argmax}}\ p(\mathcal{D}|\theta)$
  - MAP finds $\hat{\theta} = \underset{\theta}{\text{argmax}}\ p(\theta|\mathcal{D})$

$$= \underset{\theta}{\text{argmax}}\ p(\mathcal{D}|\theta)p(\theta)/p(\mathcal{D})$$

$$= \underset{\theta}{\text{argmax}}\ p(\mathcal{D}|\theta)p(\theta)$$

likelihood prior

$$= \underset{\theta}{\text{argmax}}\ \underbrace{\log p(\mathcal{D}|\theta) + \log p(\theta)}_{\text{log-posterior}}$$

MAP for Linear Regression

- If we assume a probabilistic linear model with additive Gaussian noise:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \rightarrow \boldsymbol{y} \sim N(X\boldsymbol{w}, \sigma^2 I_N)$$

and independent, identical Gaussian priors on the weights…

$$\boldsymbol{w} \sim N\left(\vec{\mathbf{0}}_{D+1}, \frac{\sigma^2}{\lambda} I_{D+1}\right) \rightarrow p(\boldsymbol{w}) \propto \exp\left(-\frac{1}{2\sigma^2}(\lambda \boldsymbol{w}^T \boldsymbol{w})\right)$$

- … then, the MAP of $\boldsymbol{w}$ is the ridge regression (L2-regularized) solution!

$$\boldsymbol{w}_{MAP} = \underset{\boldsymbol{w}}{\text{argmin}}\ (X\boldsymbol{w} - \boldsymbol{y})^T (X\boldsymbol{w} - \boldsymbol{y}) + \lambda \boldsymbol{w}^T \boldsymbol{w}$$
$$= (X^T X + \lambda I_{D+1})^{-1} X^T \boldsymbol{y}$$

Bayesian Linear Regression

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{w} \sim N(\mathbf{0}_{D+1}, \Sigma)$$

then,

$$\boldsymbol{y} \sim N(X\mathbf{0}_{D+1} + \mathbf{0}_N = \mathbf{0}_N, X\Sigma X^T + \sigma^2 I_N)$$

Bayesian Linear Regression

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{w} \sim N(\mathbf{0}_{D+1}, \Sigma)$$

then,

$$\begin{bmatrix} \boldsymbol{w} \\ \boldsymbol{y} \end{bmatrix} \sim N\left(\begin{bmatrix} \mathbf{0}_{D+1} \\ \mathbf{0}_N \end{bmatrix}, \begin{bmatrix} \Sigma & ??? \\ ??? & X\Sigma X^T + \sigma^2 I_N \end{bmatrix}\right)$$

$$\begin{aligned} \text{cov}(\vec{w}, \vec{y}) &= \text{cov}(\vec{w}, X\vec{w} + \vec{\epsilon}) \\ &= \text{Cov}(\vec{w}, X\vec{w}) \\ &= X\text{Cov}(\vec{w}, \vec{w}) = X\Sigma \end{aligned}$$

Bayesian Linear Regression

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{w} \sim N(\mathbf{0}_{D+1}, \Sigma)$$

then,

$$\begin{bmatrix} \boldsymbol{w} \\ \boldsymbol{y} \end{bmatrix} \sim N\left(\begin{bmatrix} \mathbf{0}_{D+1} \\ \mathbf{0}_N \end{bmatrix}, \begin{bmatrix} \Sigma & X\Sigma \\ X\Sigma & X\Sigma X^T + \sigma^2 I_N \end{bmatrix}\right)$$

Bayesian Linear Regression

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{w} \sim N(\mathbf{0}_{D+1}, \Sigma)$$

then,

$$\boldsymbol{w} \mid \boldsymbol{y} \sim N(\boldsymbol{\mu}_{POST}, \Sigma_{POST})$$

where

$$\boldsymbol{\mu}_{POST} = \Sigma X^T (X\Sigma X^T + \sigma^2 I_N)^{-1} \boldsymbol{y},$$
$$\Sigma_{POST} = \Sigma - \Sigma X^T (X\Sigma X^T + \sigma^2 I_N)^{-1} X\Sigma$$

Bayesian Linear Regression

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{w} \sim N(\mathbf{0}_{D+1}, \Sigma)$$

then given a new test data point $\boldsymbol{x'}$, the prediction is

$$y' \mid \boldsymbol{y} = \boldsymbol{x'}^T \boldsymbol{w} \mid \boldsymbol{y} \sim N\left(\boldsymbol{x'}^T \boldsymbol{\mu}_{POST}, \boldsymbol{x'}^T \Sigma_{POST} \boldsymbol{x'}\right)$$

where

$$\boldsymbol{\mu}_{POST} = \Sigma X^T (X\Sigma X^T + \sigma^2 I_N)^{-1} \boldsymbol{y},$$
$$\Sigma_{POST} = \Sigma - \Sigma X^T (X\Sigma X^T + \sigma^2 I_N)^{-1} X\Sigma$$

Bayesian Linear Regression

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{w} \sim N(\mathbf{0}_{D+1}, \Sigma)$$

then given a new test data point $\boldsymbol{x}'$, the prediction is

$$y' \mid \boldsymbol{y} = \boldsymbol{x}'^T \boldsymbol{w} \mid \boldsymbol{y} \sim N(\boldsymbol{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\boldsymbol{\mu}_{PRED} = \boldsymbol{x}'^T \Sigma X^T (X\Sigma X^T + \sigma^2 I_N)^{-1} \boldsymbol{y},$$
$$\Sigma_{PRED} = \boldsymbol{x}'^T \Sigma \boldsymbol{x}' - \boldsymbol{x}'^{\mathrm{T}} \Sigma X^T (X\Sigma X^T + \sigma^2 I_N)^{-1} X\Sigma \boldsymbol{x}'$$

Bayesian Non-linear Regression…

$$\Phi = \begin{bmatrix} 1 & \phi(\boldsymbol{x}^{(1)})^T \\ 1 & \phi(\boldsymbol{x}^{(2)})^T \\ \vdots & \vdots \\ 1 & \phi(\boldsymbol{x}^{(N)})^T \end{bmatrix}$$

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = \Phi\boldsymbol{\omega} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{\omega} \sim N(\mathbf{0}_{D'+1}, \Sigma)$$

then given a new test data point $\boldsymbol{x}'$, the prediction is

$$y' \mid \boldsymbol{y} = \phi(\boldsymbol{x}')^T\boldsymbol{\omega} \mid \boldsymbol{y} \sim N(\boldsymbol{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\boldsymbol{\mu}_{PRED} = \phi(\boldsymbol{x}')^T\Sigma\Phi^T(\Phi\Sigma\Phi^T + \sigma^2 I_N)^{-1}\boldsymbol{y},$$

$$\Sigma_{PRED} = \phi(\boldsymbol{x}')^T\Sigma\phi(\boldsymbol{x}') - \phi(\boldsymbol{x}')^T\Sigma\Phi^T(\Phi\Sigma\Phi^T + \sigma^2 I_N)^{-1}\Phi\Sigma\phi(\boldsymbol{x}')$$

Bayesian Non-linear Regression can be "kernelized"

$$\Phi = \begin{bmatrix} 1 & \phi(\boldsymbol{x}^{(1)})^T \\ 1 & \phi(\boldsymbol{x}^{(2)})^T \\ \vdots & \vdots \\ 1 & \phi(\boldsymbol{x}^{(N)})^T \end{bmatrix}$$

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = \Phi\boldsymbol{\omega} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{\omega} \sim N(\mathbf{0}_{D'+1}, \Sigma)$$

then given a new test data point $\boldsymbol{x}'$, the prediction is

$$y' \mid \boldsymbol{y} = \phi(\boldsymbol{x}')^T\boldsymbol{\omega} \mid \boldsymbol{y} \sim N(\boldsymbol{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\boldsymbol{\mu}_{PRED} = \phi(\boldsymbol{x}')^T\Sigma\Phi^T(\Phi\Sigma\Phi^T + \sigma^2 I_N)^{-1}\boldsymbol{y},$$

$$\Sigma_{PRED} = \phi(\boldsymbol{x}')^T\Sigma\phi(\boldsymbol{x}') - \phi(\boldsymbol{x}')^T\Sigma\Phi^T(\Phi\Sigma\Phi^T + \sigma^2 I_N)^{-1}\Phi\Sigma\phi(\boldsymbol{x}')$$

- Define a **kernel function** as

$$K(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T\Sigma\phi(\boldsymbol{x}')$$

Bayesian Linear Regression can be kernelized!

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = \Phi\boldsymbol{\omega} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{\omega} \sim N(\mathbf{0}_{D'+1}, \Sigma)$$

then given a new test data point $\boldsymbol{x'}$, the prediction is

$$y' \mid \boldsymbol{y} = \phi(\boldsymbol{x'})^T\boldsymbol{\omega} \mid \boldsymbol{y} \sim N(\boldsymbol{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\boldsymbol{\mu}_{PRED} = K(\boldsymbol{x'}, X)(K(X, X) + \sigma^2 I_N)^{-1}\boldsymbol{y},$$

$$\Sigma_{PRED} = K(\boldsymbol{x'}, \boldsymbol{x'}) - K(\boldsymbol{x'}, X)(K(X, X) + \sigma^2 I_N)^{-1}K(X, \boldsymbol{x})$$

- Define the kernel function to be

$$K(\boldsymbol{x}, \boldsymbol{x'}) = \phi(\boldsymbol{x})^T\Sigma\phi(\boldsymbol{x'})$$

Wait, what happened to the weights?

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$\boldsymbol{y} = \Phi\boldsymbol{\omega} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \boldsymbol{\omega} \sim N(\mathbf{0}_{D'+1}, \Sigma)$$

then given a new test data point $\boldsymbol{x}'$, the prediction is

$$y' \mid \boldsymbol{y} = \phi(\boldsymbol{x}')^T\boldsymbol{\omega} \mid \boldsymbol{y} \sim N(\boldsymbol{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\boldsymbol{\mu}_{PRED} = K(\boldsymbol{x}', X)(K(X, X) + \sigma^2 I_N)^{-1}\boldsymbol{y},$$

$$\Sigma_{PRED} = K(\boldsymbol{x}', \boldsymbol{x}') - K(\boldsymbol{x}', X)(K(X, X) + \sigma^2 I_N)^{-1}K(X, \boldsymbol{x})$$

- Define the kernel function to be

$$K(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T\Sigma\phi(\boldsymbol{x}')$$

Outline

Gaussian process =

Bayesian linear regression + Kernels

A new perspective

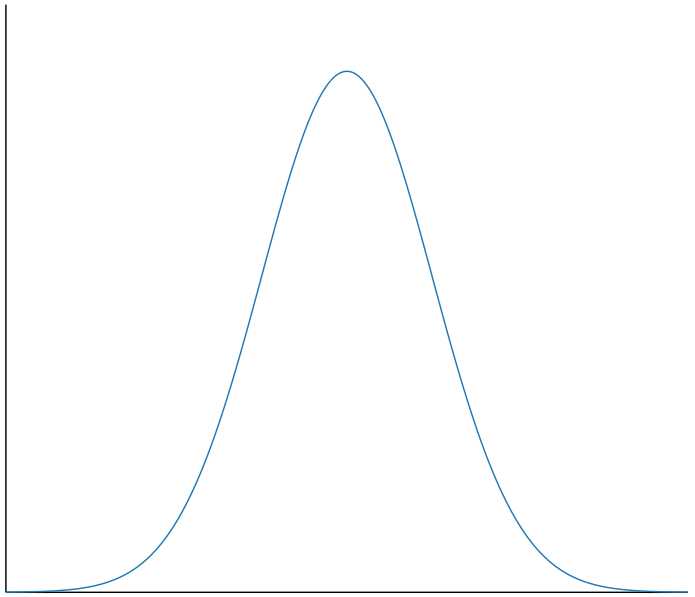
Gaussian process =

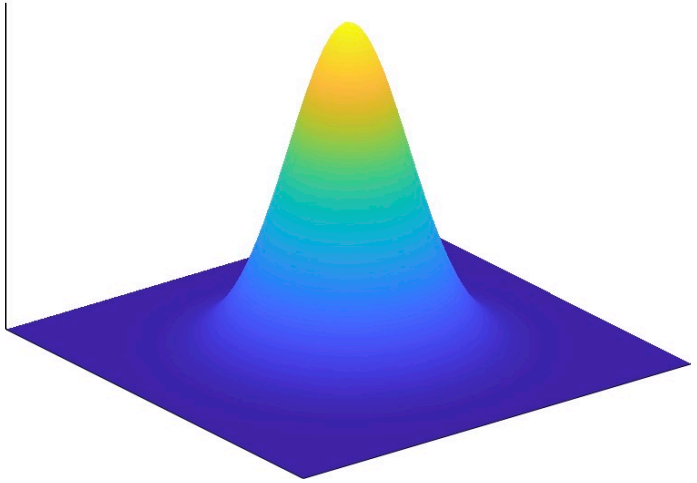
The extension of a Gaussian distribution to functions

Gaussians

- (Univariate) Gaussians:

$$x \sim \mathcal{N}(x; \mu = 0, \sigma^2 = 1)$$





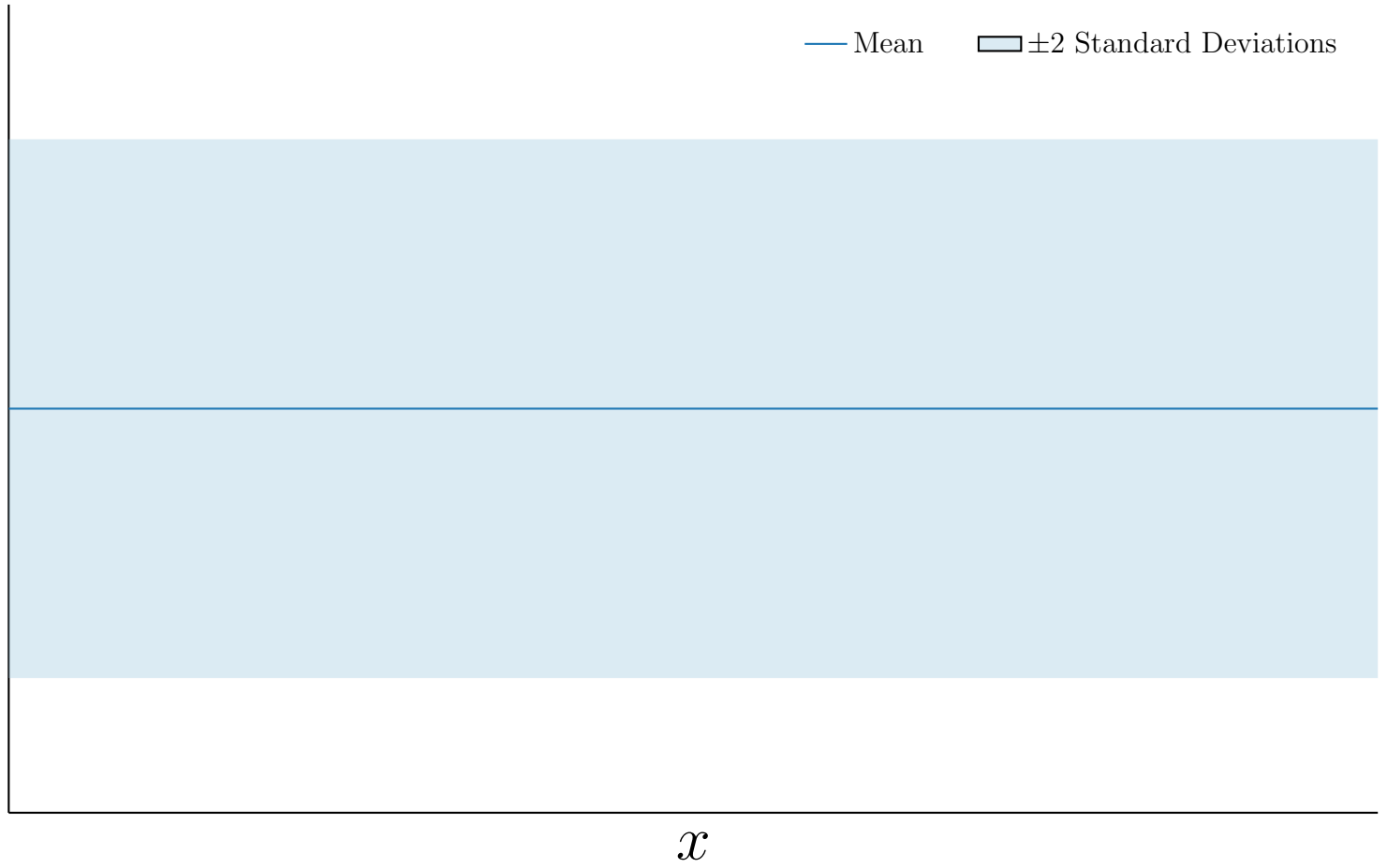
- Multivariate Gaussians:

$$\boldsymbol{x} = [x_1, \ldots, x_D]^T$$

$$\sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu} = \mathbf{0}_D, \Sigma = I_D)$$

Gaussian Process (GP)

$$f: \mathbb{R}^p \mapsto \mathbb{R} \sim \mathcal{GP}(f; \mu(x) \quad , \Sigma(x, x') \quad)$$

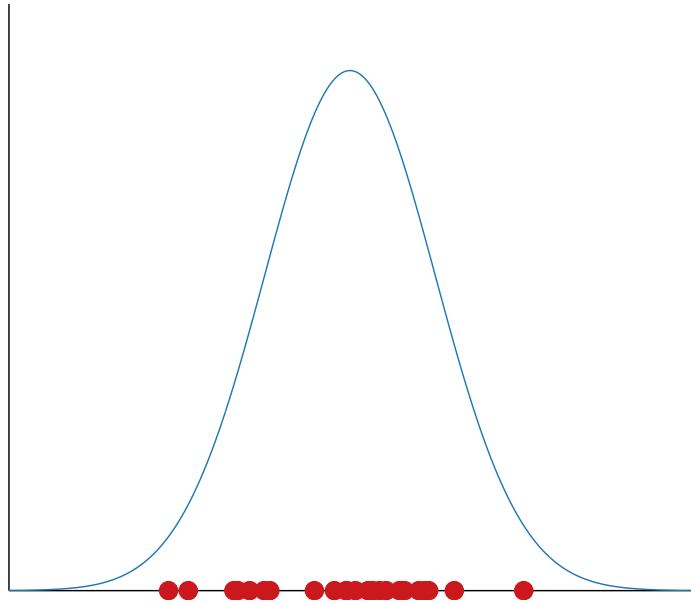


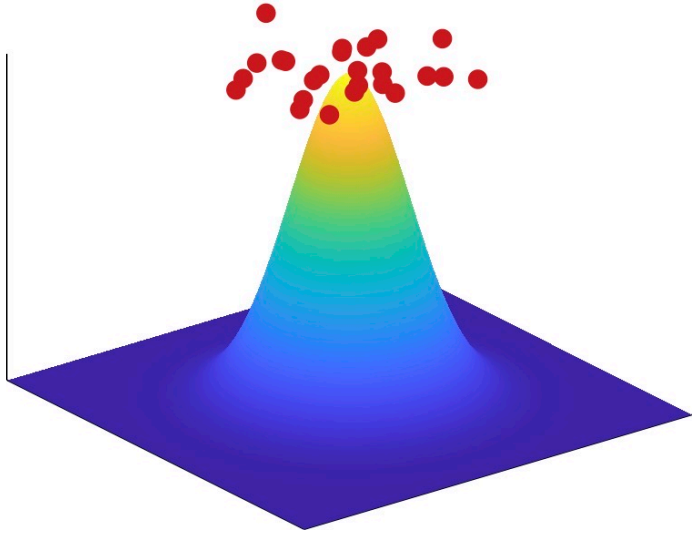
$$f \sim \mathcal{GP}(\mu, \Sigma) \rightarrow f(x) \sim \mathcal{N}\big(\mu(x), \Sigma(x, x)\big)$$

Gaussians

- (Univariate) Gaussians:

$$x \sim \mathcal{N}(x; \mu = 0, \sigma^2 = 1)$$





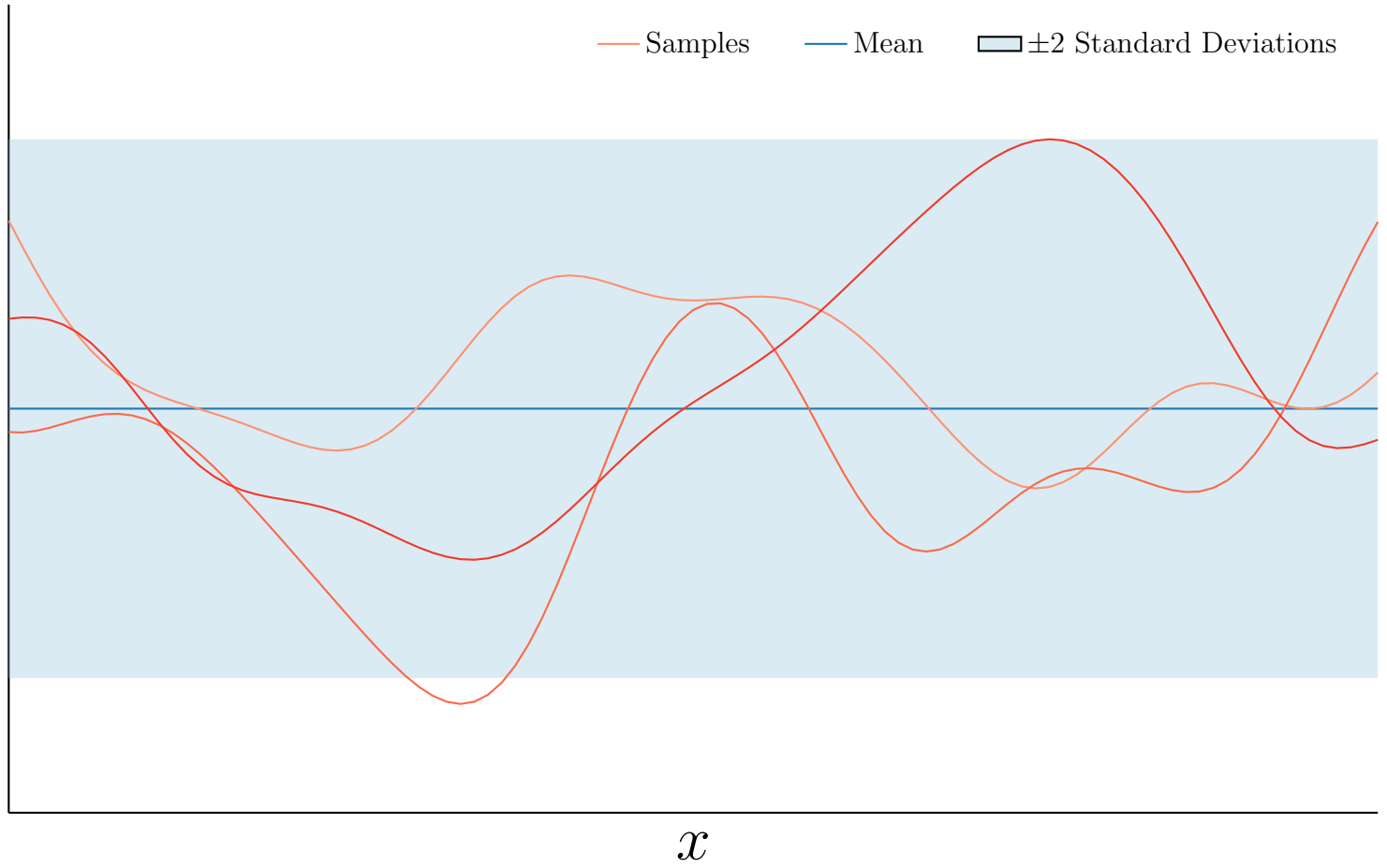
- Multivariate Gaussians:

$$\boldsymbol{x} = [x_1, \ldots, x_D]^T$$

$$\sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu} = \mathbf{0}_D, \Sigma = I_D)$$

Gaussian Process (GP)

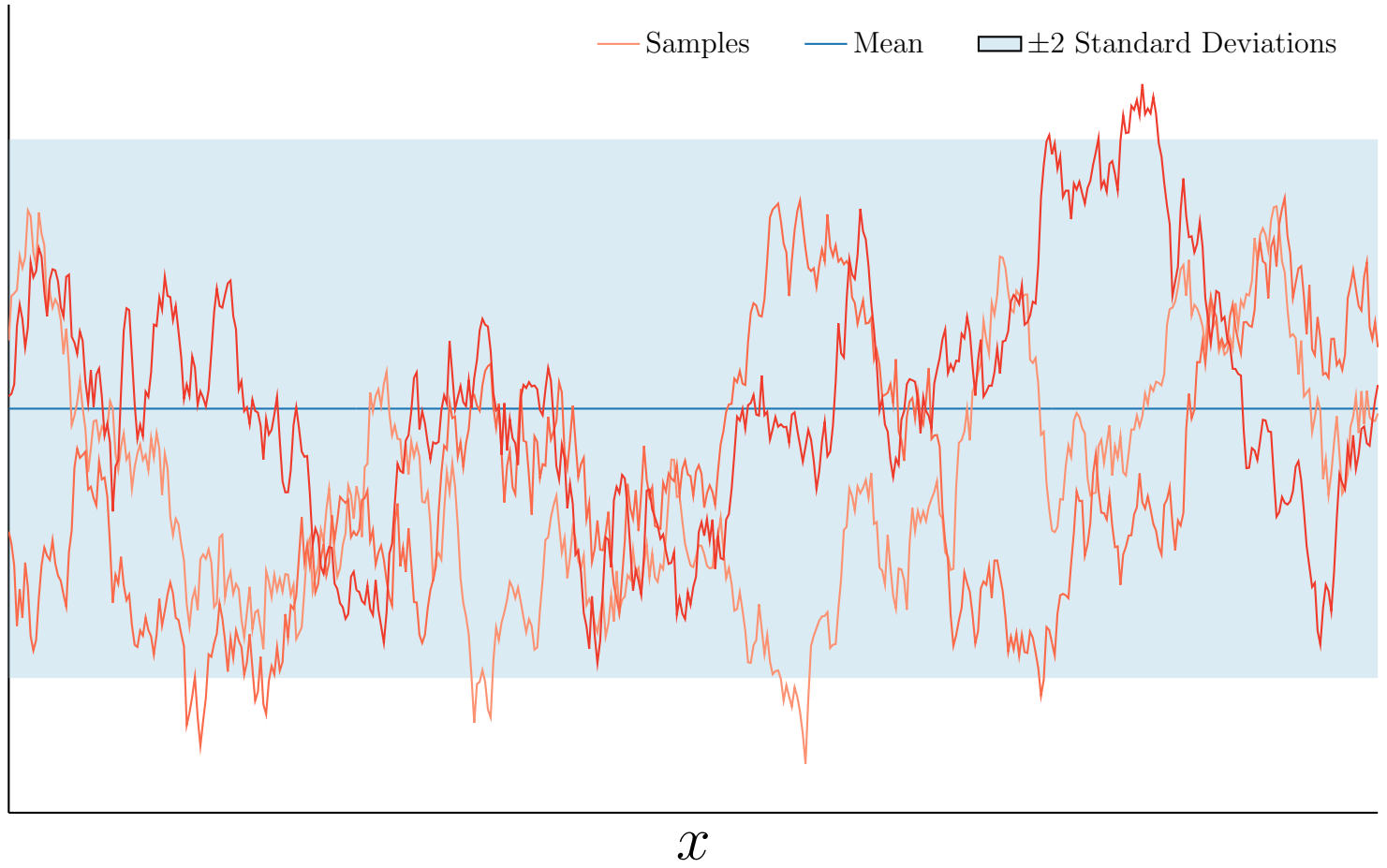
$$f: \mathbb{R}^p \mapsto \mathbb{R} \sim \mathcal{GP}(f; \mu(x) = 0, \Sigma(x, x') = \exp(-(x - x')^2))$$



$$f \sim \mathcal{GP}(\mu, \Sigma) \rightarrow f(x) \sim \mathcal{N}\big(\mu(x), \Sigma(x, x)\big)$$

Gaussian Process (GP)

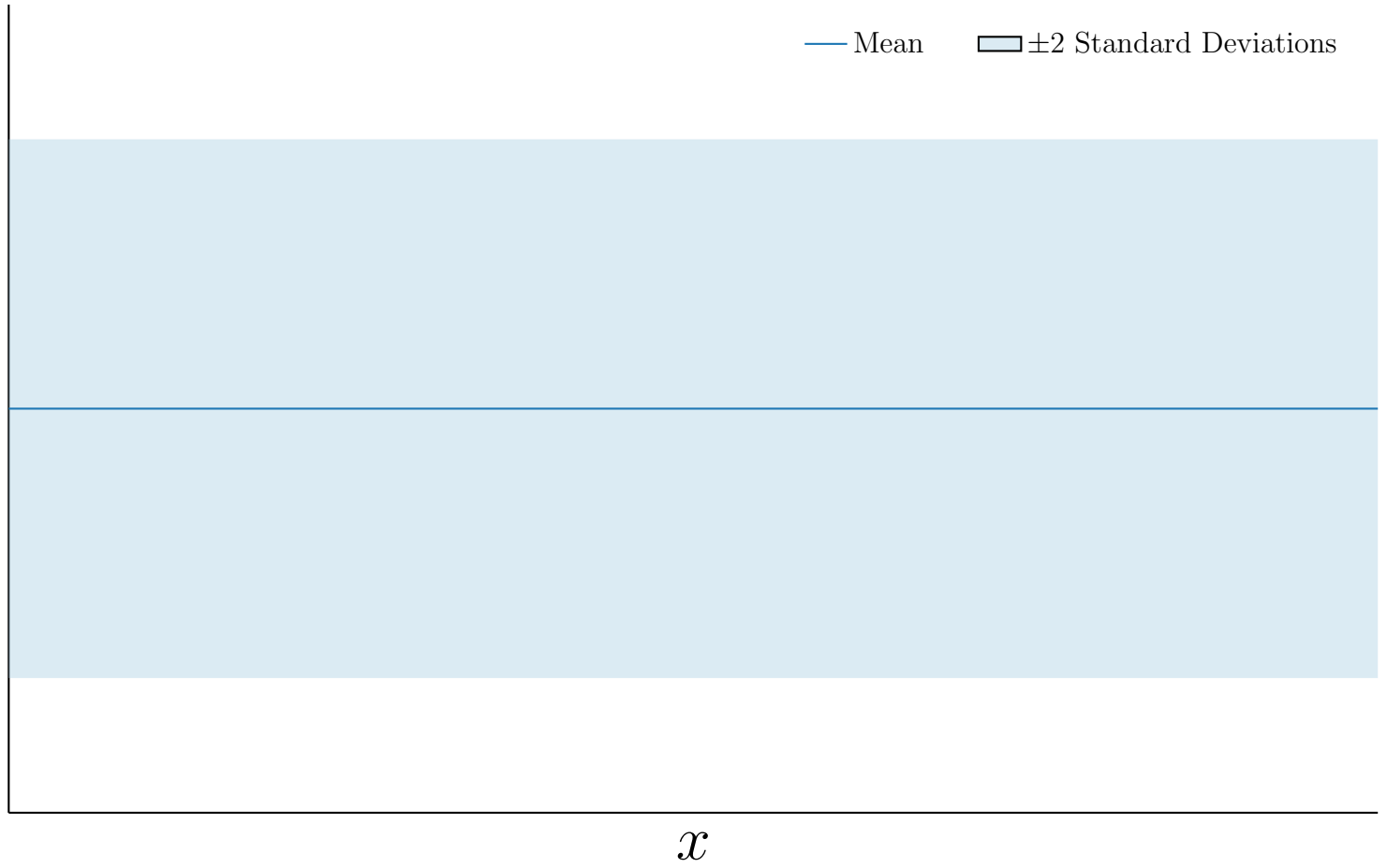
$$f: \mathbb{R}^p \mapsto \mathbb{R} \sim \mathcal{GP}(f; \mu(x) = 0, \Sigma(x, x') = \exp(-|x - x'|))$$



$$f \sim \mathcal{GP}(\mu, \Sigma) \rightarrow f(x) \sim \mathcal{N}\big(\mu(x), \Sigma(x, x)\big)$$

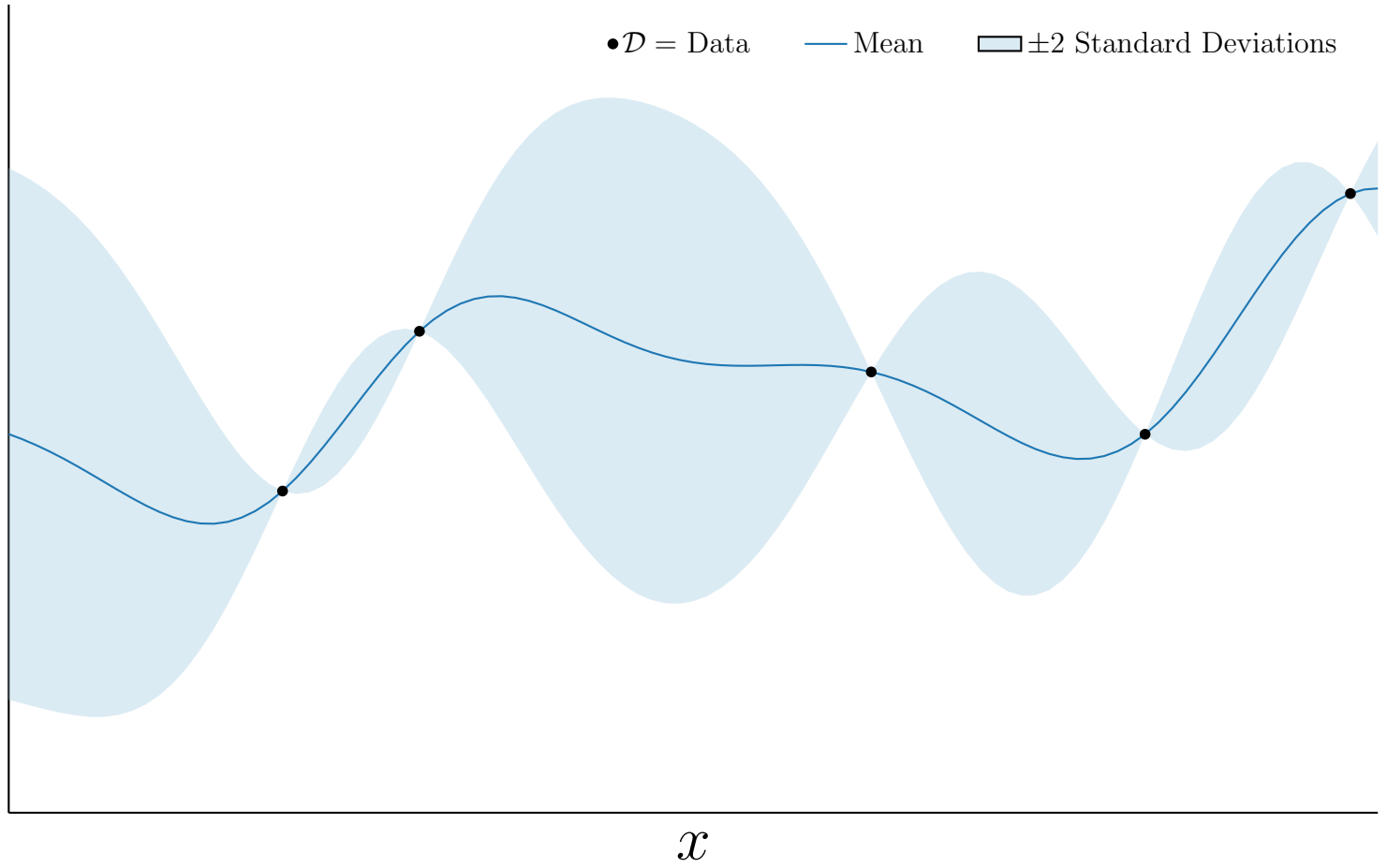
GP Prior

$$f: \mathbb{R}^p \mapsto \mathbb{R} \sim \mathcal{GP}(f; \mu(x) = 0, \Sigma(x, x') = \exp(-(x - x')^2))$$



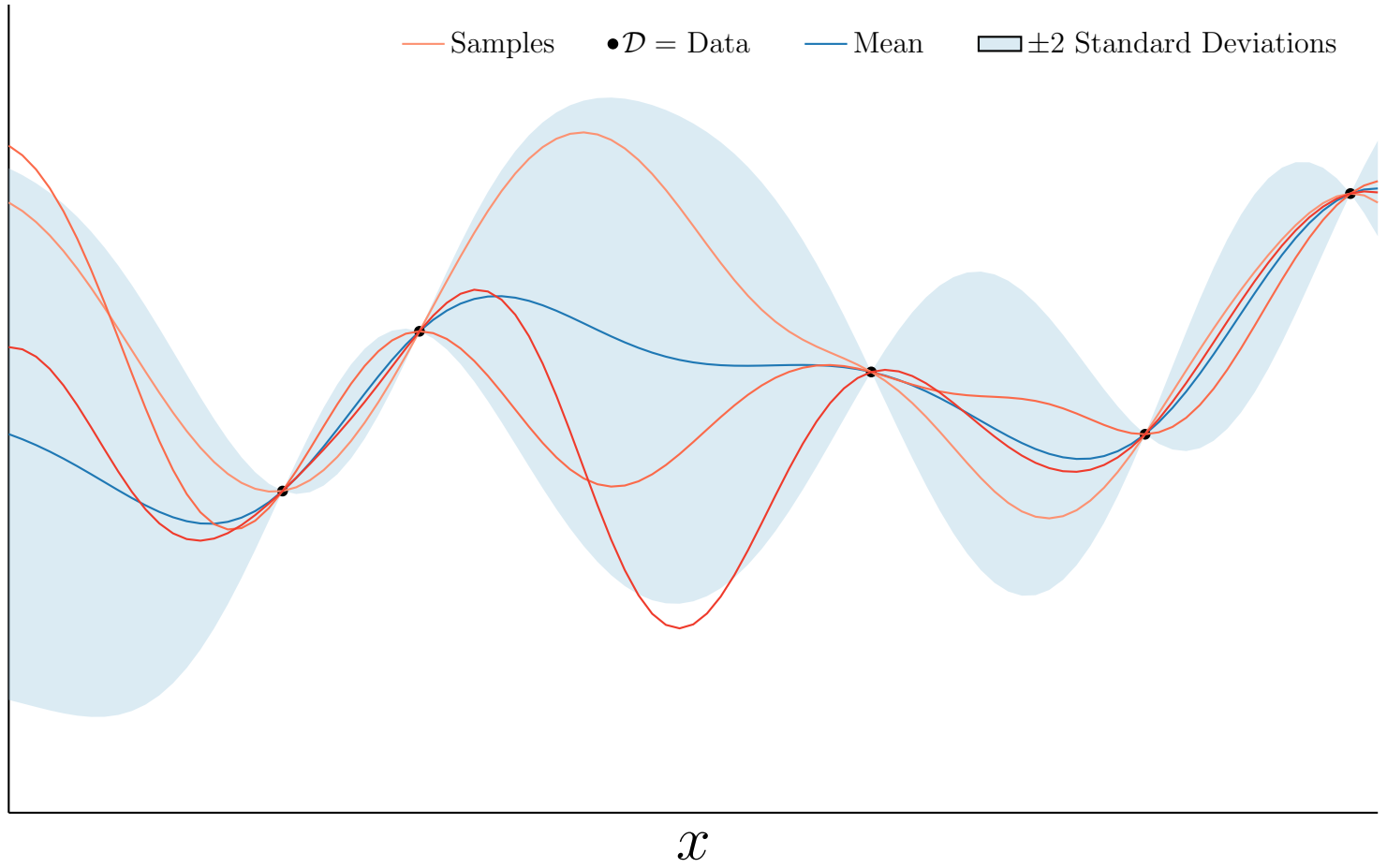
GP Posterior

$$f \mid \mathcal{D} \sim \mathcal{GP}(f; \mu_{\mathcal{D}}, \Sigma_{\mathcal{D}})$$



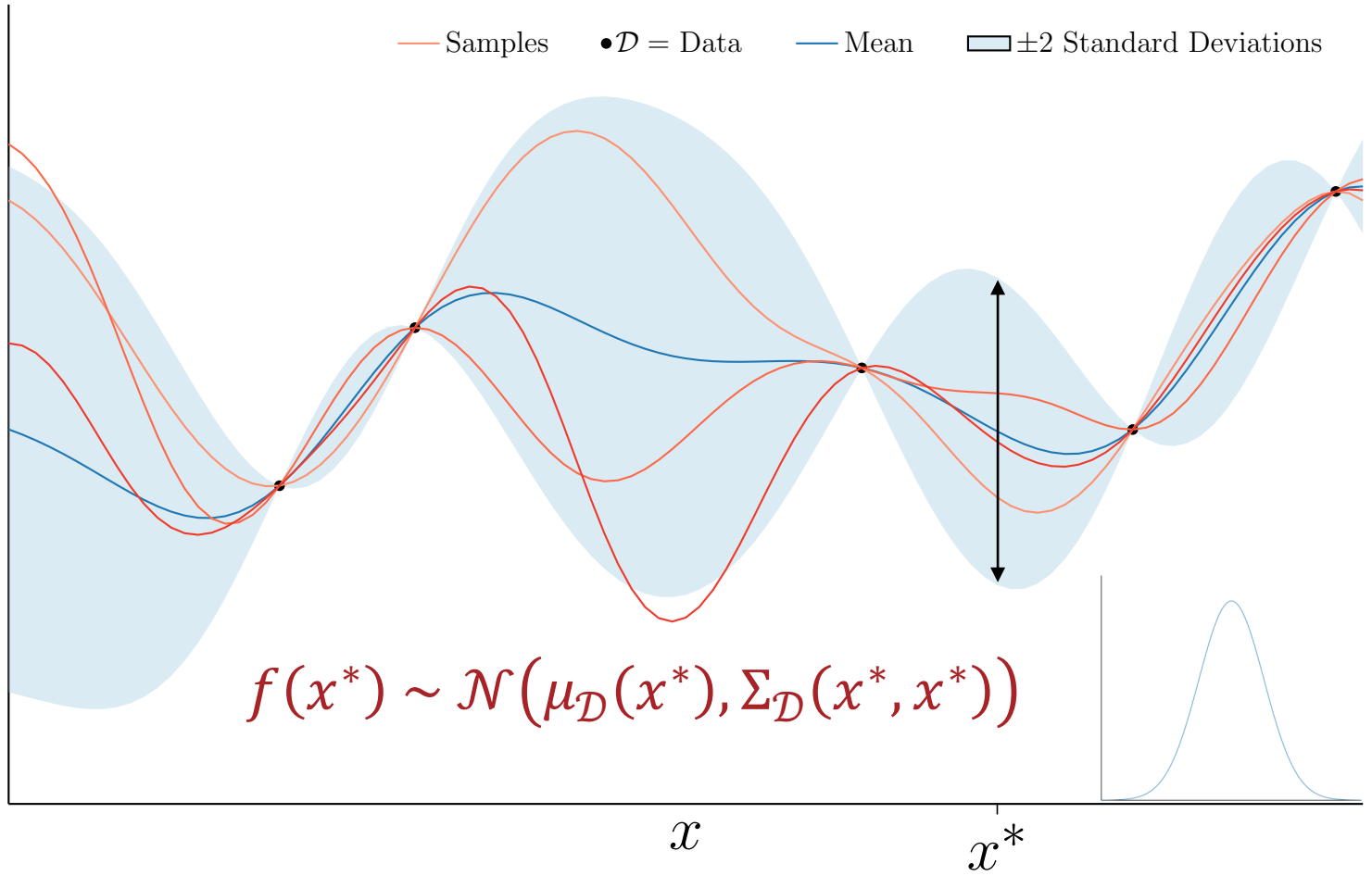
GP Posterior

$$f \mid \mathcal{D} \sim \mathcal{GP}(f; \mu_{\mathcal{D}}, \Sigma_{\mathcal{D}})$$

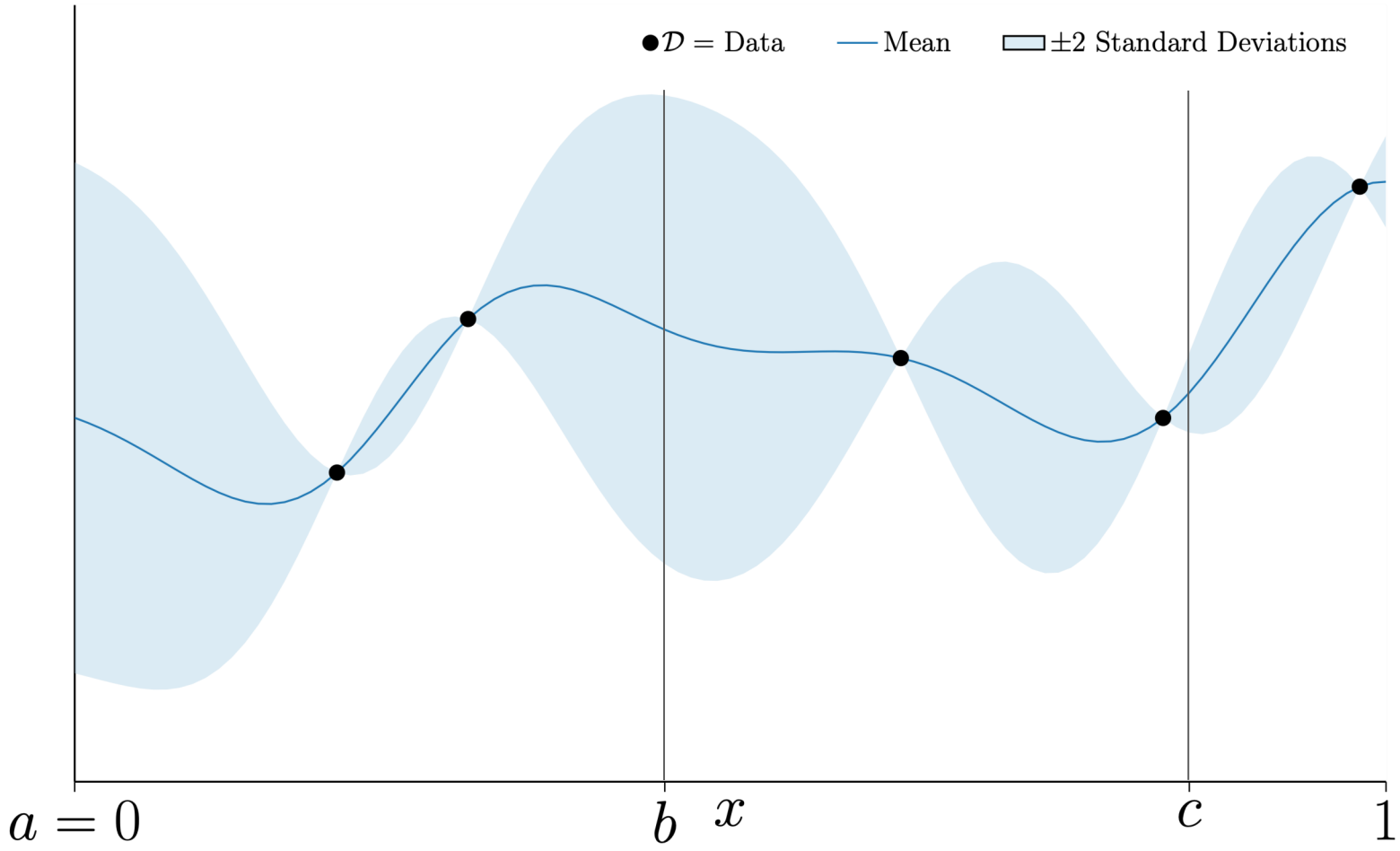


GP Posterior

$$f \mid \mathcal{D} \sim \mathcal{GP}(f; \mu_{\mathcal{D}}, \Sigma_{\mathcal{D}})$$

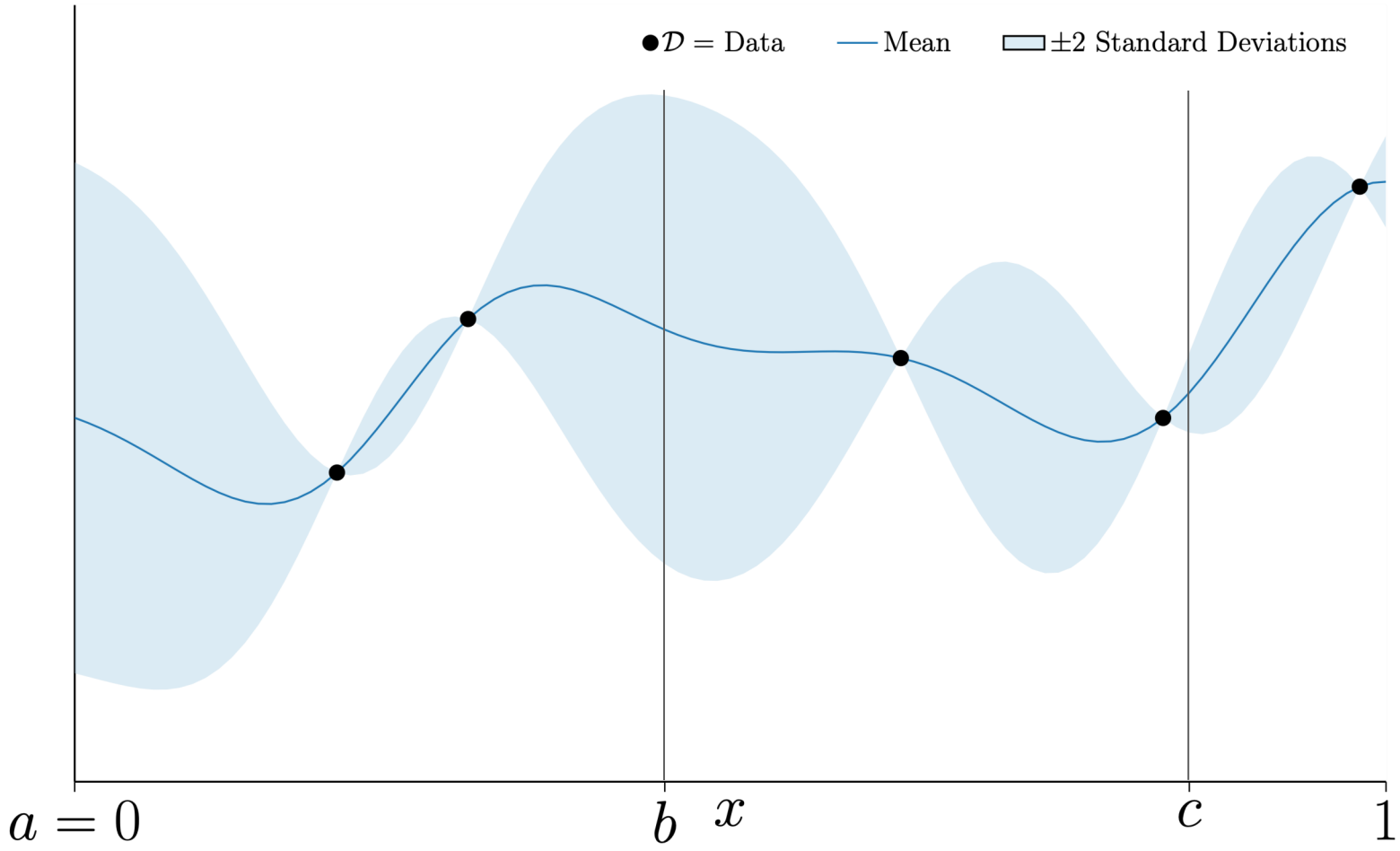


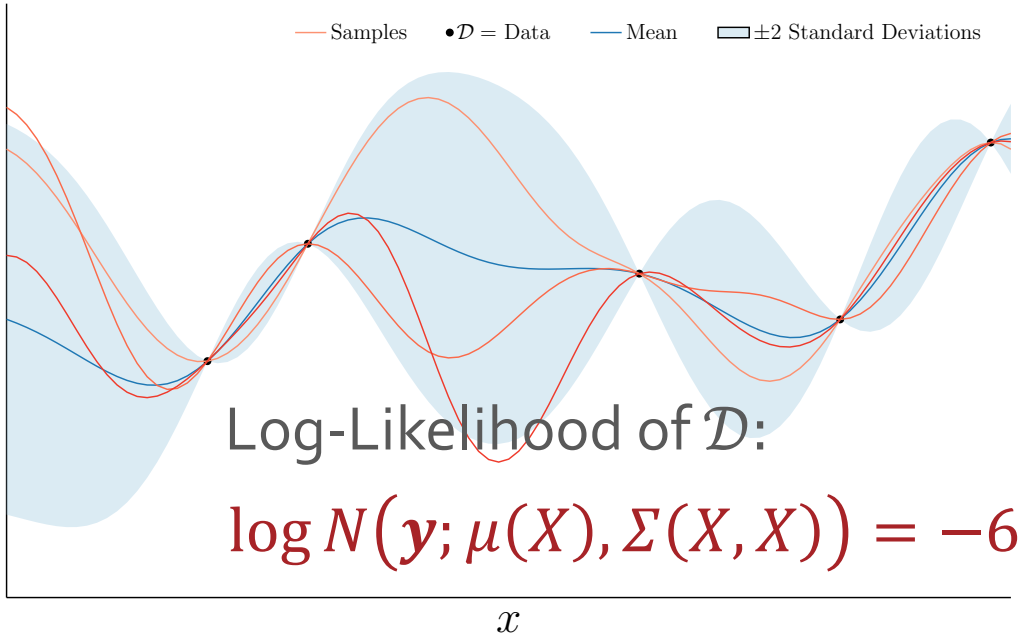
Active Learning

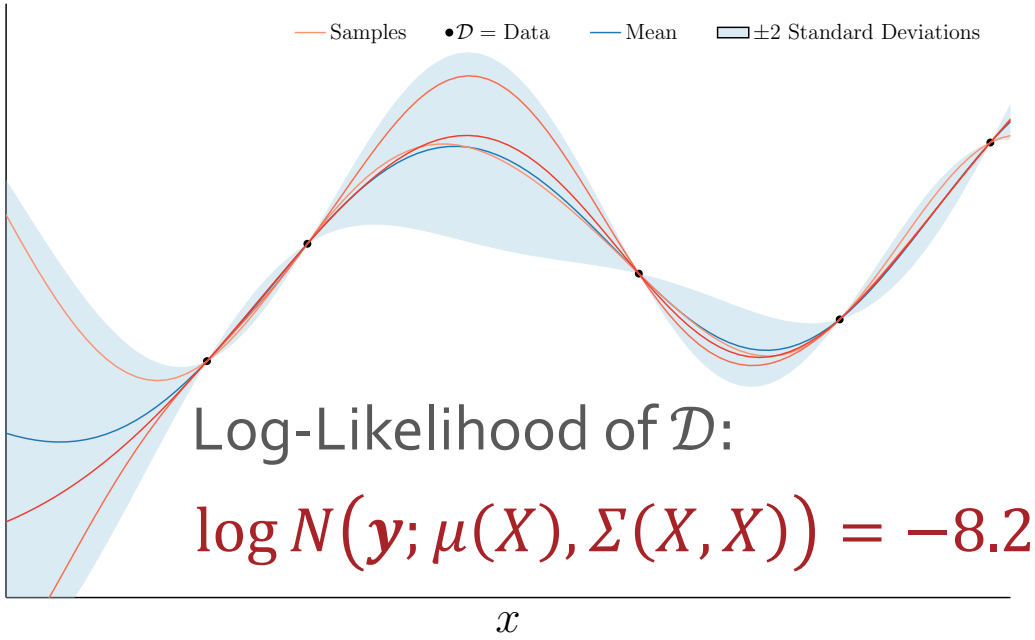


Suppose you can add one data point to your training dataset.

Which value of x would you add and why?





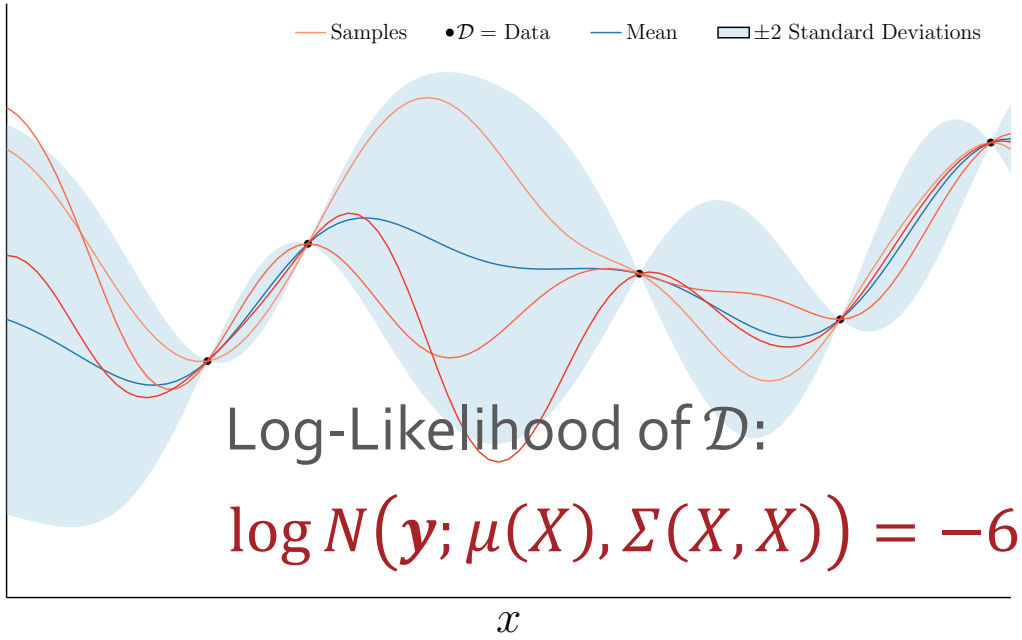


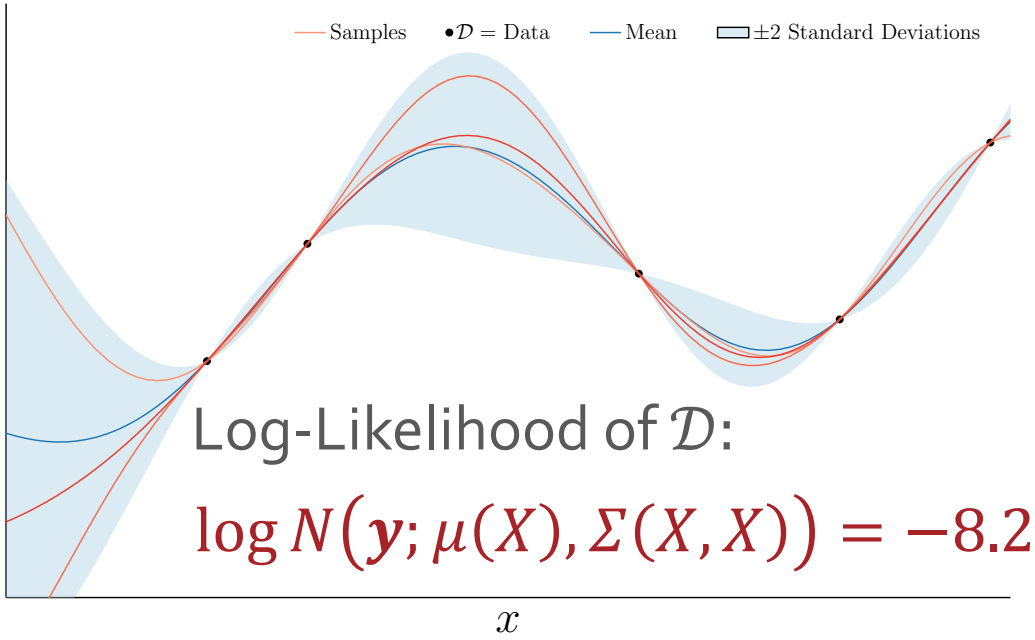
$$f \sim \mathcal{GP}\left(f;\ 0, (1^2)\exp\left(-\frac{(x-x')^2}{1^2}\right)\right)$$

$$f \sim \mathcal{GP}\left(f;\ 0, (2^2)\exp\left(-\frac{(x-x')^2}{2^2}\right)\right)$$

Kernel Hyperparameters

- Can be set via MLE
- As long as μ and Σ are differentiable, the log-likelihood is differentiable with respect to the kernel hyperparameters





$$f \sim \mathcal{GP}\left(f;\ 0, (1^2)\exp\left(-\frac{(x-x')^2}{1^2}\right)\right)$$

$$f \sim \mathcal{GP}\left(f;\ 0, (2^2)\exp\left(-\frac{(x-x')^2}{2^2}\right)\right)$$

Wait doesn't this always get zero training error???

Noise

- Assume a linear model with additive Gaussian noise and a zero-mean Gaussian prior on the weights:

$$y = \Phi\omega + \epsilon \text{ where } \epsilon \sim N(\mathbf{0}_N, \sigma^2 I_N) \text{ and } \omega \sim N(\mathbf{0}_{D'+1}, \Sigma)$$

then given a new test data point x', the prediction is

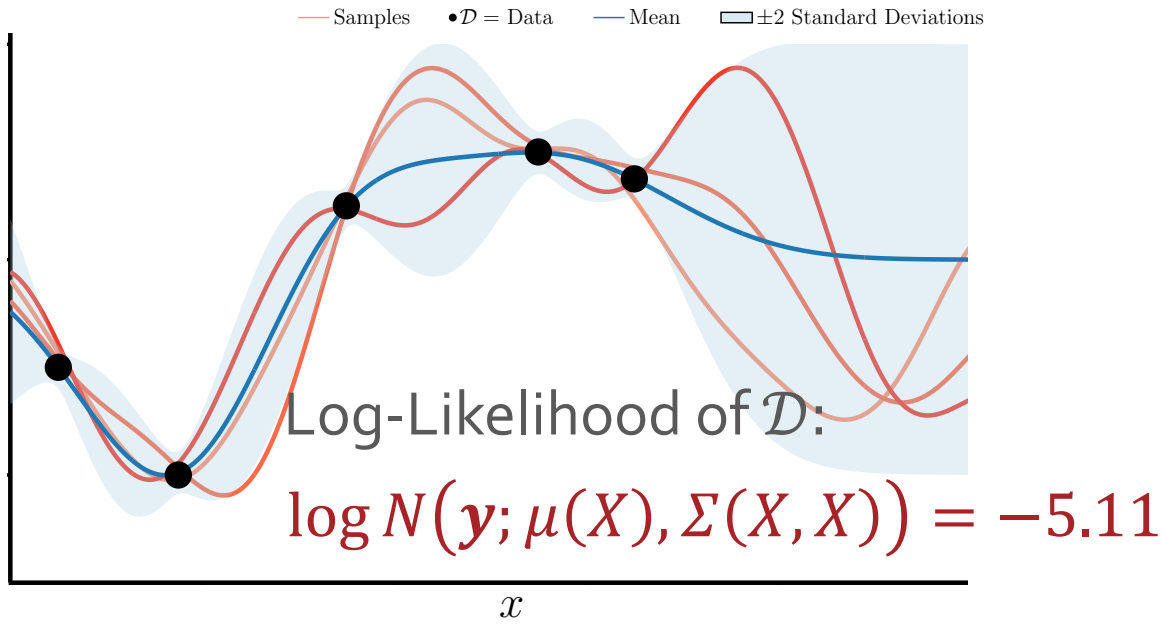
$$y' \mid y = \phi(x')^T \omega \mid y \sim N(\mu_{PRED}, \Sigma_{PRED})$$

where

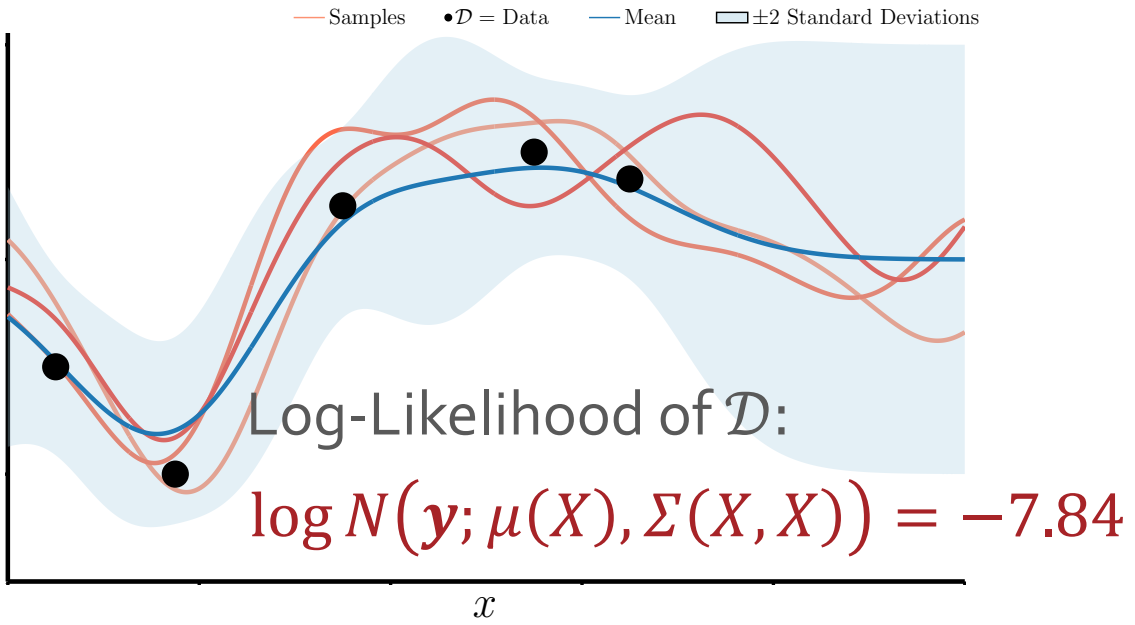
$$\mu_{PRED} = K(x', X)(K(X, X) + \sigma^2 I_N)^{-1} y,$$

$$\Sigma_{PRED} = K(x', x') - K(x', X)(K(X, X) + \sigma^2 I_N)^{-1} K(X, x)$$

- σ^2 is another hyperparameter we can tune
  - $\sigma^2 = 0$ is a noiseless fit: the mean will always pass through the observations exactly; $\sigma^2 > 0$ allows for deviations



$\sigma^2 = 0.1$



$\sigma^2 = 0.5$

Noise