10-315: Introduction to Machine Learning Lecture 23 – Gaussian Processes

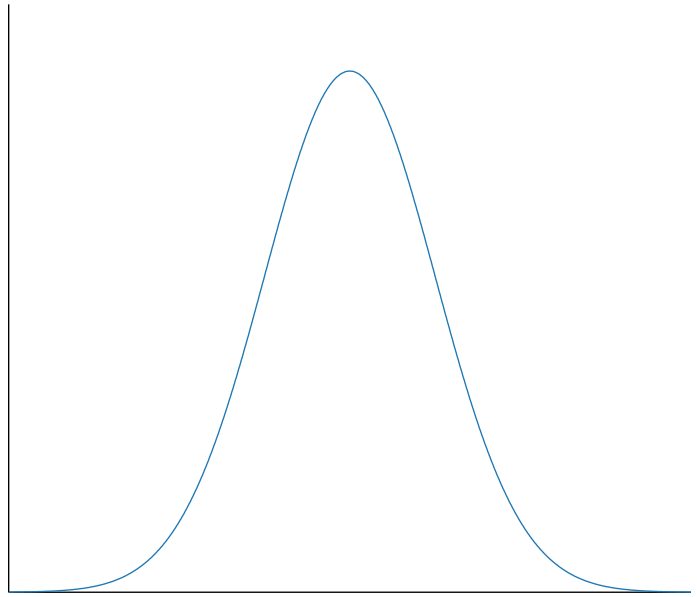
Henry Chai

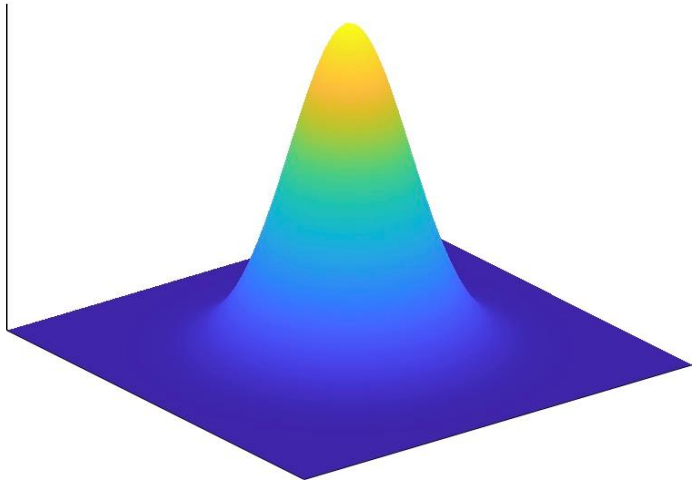
4/18/22

Gaussians

- (Univariate) Gaussians:

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





- Multivariate Gaussians:

$$\vec{x} = \left[x_1, \ldots, x_p\right]^T$$

$$\sim \mathcal{N}\left(\vec{x}; \vec{\mu} = \vec{0}_p, \Sigma = I_p\right)$$

Some fun facts about Gaussians

- Closure under linear transformations:

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

- Closure under addition:

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

- Closure under conditioning:

$$\text{if } \vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

$$\text{then } x_1 \mid x_2 = c \sim N(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(c - \mu_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$$

Some old friends

Gaussian process =

Bayesian linear regression + Kernels

Some old friends

Gaussian process =

Bayesian linear regression + Kernels

Recall: MAP for Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

- If we assume a linear model with additive Gaussian noise

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \rightarrow \vec{y} \sim N\left(A\vec{\beta}, \sigma^2 I_n\right)$$

and a Gaussian prior on the weights…

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

- … then, the MAP of $\vec{\beta}$ is the ridge regression solution!

$$\vec{\beta}_{MAP} = \underset{\vec{\beta}}{\operatorname{argmin}} \left(A\vec{\beta} - \vec{y}\right)^T \left(A\vec{\beta} - \vec{y}\right) + \lambda \vec{\beta}^T \vec{\beta}$$

$$= \left(A^T A + \lambda I_{p+1}\right)^{-1} A^T \vec{y}$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then,

$$\vec{y} \sim N\left(A\vec{0} + \vec{0},\ A\Sigma A^T + \sigma^2 I_n\right)$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then,

$$\vec{y} \sim N\left(\vec{0}_n, A\Sigma A^T + \sigma^2 I_n\right)$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then,

$$\begin{bmatrix} \vec{\beta} \\ \vec{y} \end{bmatrix} \sim N\left(\begin{bmatrix} \vec{0}_{p+1} \\ \vec{0}_n \end{bmatrix}, \begin{bmatrix} \Sigma & ??? \\ ??? & A\Sigma A^T + \sigma^2 I_n \end{bmatrix}\right)$$

- Covariance between $\vec{y}$ and $\vec{\beta}$:

$$Cov(\vec{y}, \vec{\beta}) = Cov(A\vec{\beta} + \vec{\epsilon}, \vec{\beta}) = Cov(A\vec{\beta}, \vec{\beta})$$
$$= A\,Cov(\vec{\beta}, \vec{\beta}) = A\Sigma$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then,

$$\begin{bmatrix} \vec{\beta} \\ \vec{y} \end{bmatrix} \sim N\left(\begin{bmatrix} \vec{0}_{p+1} \\ \vec{0}_n \end{bmatrix}, \begin{bmatrix} \Sigma & \Sigma A^T \\ A\Sigma & A\Sigma A^T + \sigma^2 I_n \end{bmatrix}\right)$$

- Covariance between $\vec{y}$ and $\vec{\beta}$:

$$\text{Cov}\left(\vec{y}, \vec{\beta}\right) = \text{Cov}\left(A\vec{\beta} + \vec{\epsilon}, \vec{\beta}\right) = A\text{Cov}\left(\vec{\beta}, \vec{\beta}\right) = A\Sigma$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then,

$$\vec{\beta}|\vec{y} \sim N(\vec{\mu}_{POST}, \Sigma_{POST})$$

where

$$\vec{\mu}_{POST} = \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} \vec{y},$$

$$\Sigma_{POST} = \Sigma - \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} A\Sigma$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^* | \vec{y} = \vec{x}^{*T} \vec{\beta} \, | \vec{y} \sim N\left(\vec{x}^{*T} \vec{\mu}_{POST}, \vec{x}^{*T} \Sigma_{POST} \vec{x}^*\right)$$

where

$$\vec{\mu}_{POST} = \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} \vec{y},$$

$$\Sigma_{POST} = \Sigma - \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} A\Sigma$$

Bayesian Linear Regression

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^* | \vec{y} = \vec{x}^{*T} \vec{\beta} \, | \vec{y} \sim N(\vec{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\vec{\mu}_{PRED} = \vec{x}^{*T} \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} \vec{y},$$

$$\Sigma_{PRED} = \vec{x}^{*T} \Sigma \vec{x}^* - \vec{x}^{*T} \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} A\Sigma \vec{x}^*$$

Some old friends

Gaussian process =

Bayesian linear regression + Kernels

Some old friends

Gaussian process =

Bayesian linear regression + Kernels

Bayesian Linear Regression…

$$A = \begin{bmatrix} 1 & \vec{x}_1 \\ 1 & \vec{x}_2 \\ \vdots & \vdots \\ 1 & \vec{x}_n \end{bmatrix}$$

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

$$\vec{y} = A\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^* | \vec{y} = \vec{x}^{*T} \vec{\beta} \, | \vec{y} \sim N(\vec{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\vec{\mu}_{PRED} = \vec{x}^{*T} \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} \vec{y},$$

$$\Sigma_{POST} = \vec{x}^{*T} \Sigma \vec{x}^* - \vec{x}^{*T} \Sigma A^T \left(A\Sigma A^T + \sigma^2 I_n\right)^{-1} A\Sigma \vec{x}^*$$

Bayesian Linear Regression can be kernelized!

$$\Phi = \begin{bmatrix} 1 & \phi(\vec{x}_1) \\ 1 & \phi(\vec{x}_2) \\ \vdots & \vdots \\ 1 & \phi(\vec{x}_n) \end{bmatrix}$$

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

$$\vec{y} = \Phi\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p'+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^*|\vec{y} = \phi(\vec{x}^*)^T\vec{\beta}\ |\vec{y} \sim N(\vec{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\vec{\mu}_{PRED} = \phi(\vec{x}^*)^T\Sigma\Phi^T\left(\Phi\Sigma\Phi^T + \sigma^2 I_n\right)^{-1}\vec{y},$$

$$\Sigma_{POST}$$

$$= \phi(\vec{x}^*)^T\Sigma\phi(\vec{x}^*) - \phi(\vec{x}^*)^T\Sigma\Phi^T\left(\Phi\Sigma\Phi^T + \sigma^2 I_n\right)^{-1}\Phi\Sigma\phi(\vec{x}^*)$$

Bayesian Linear Regression can be kernelized!

$$\Phi = \begin{bmatrix} 1 & \phi(\vec{x}_1)^T \\ 1 & \phi(\vec{x}_2)^T \\ \vdots & \vdots \\ 1 & \phi(\vec{x}_n)^T \end{bmatrix}$$

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

$$\vec{y} = \Phi\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p'+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^*|\vec{y} = \phi(\vec{x}^*)^T\vec{\beta}\,|\vec{y} \sim N(\vec{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\vec{\mu}_{PRED} = \phi(\vec{x}^*)^T\Sigma\Phi^T\left(\Phi\Sigma\Phi^T + \sigma^2 I_n\right)^{-1}\vec{y},$$

$$\Sigma_{POST}$$

$$= \phi(\vec{x}^*)^T\Sigma\phi(\vec{x}^*) - \phi(\vec{x}^*)^T\Sigma\Phi^T\left(\Phi\Sigma\Phi^T + \sigma^2 I_n\right)^{-1}\Phi\Sigma\phi(\vec{x}^*)$$

- Define the kernel function to be

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

$\Sigma = UU^T$ is p.s.d.

Bayesian Linear Regression can be kernelized!

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

$$\vec{y} = \Phi\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p'+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^* | \vec{y} \sim N(\vec{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\vec{\mu}_{PRED} = K(\vec{x}^*, A)\left(K(A,A) + \sigma^2 I_n\right)^{-1}\vec{y},$$

$$\Sigma_{POST} = K(\vec{x}^*, \vec{x}^*) - K(\vec{x}^*, A)\left(K(A,A) + \sigma^2 I_n\right)^{-1}K(A, \vec{x}^*)$$

- Define the kernel function to be

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

Wait, what happened to the weights?

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

$$\vec{y} = \Phi\vec{\beta} + \vec{\epsilon} \text{ where } \vec{\epsilon} \sim N\left(\vec{0}_n, \sigma^2 I_n\right) \text{ and } \vec{\beta} \sim N\left(\vec{0}_{p'+1}, \Sigma\right)$$

- then given a new test data point $\vec{x}^*$, the prediction is

$$y^*|\vec{y} \sim N(\vec{\mu}_{PRED}, \Sigma_{PRED})$$

where

$$\vec{\mu}_{PRED} = K(\vec{x}^*, A)\left(K(A,A) + \sigma^2 I_n\right)^{-1}\vec{y},$$

$$\Sigma_{POST} = K(\vec{x}^*, \vec{x}^*) - K(\vec{x}^*, A)\left(K(A,A) + \sigma^2 I_n\right)^{-1}K(A, \vec{x}^*)$$

- Define the kernel function to be

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

Some old friends

Gaussian process =

Bayesian linear regression + Kernels

A new perspective

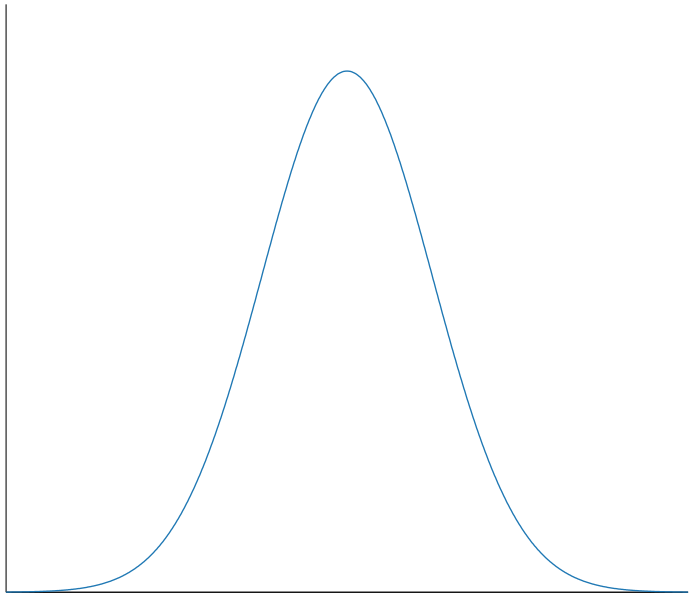
Gaussian process =

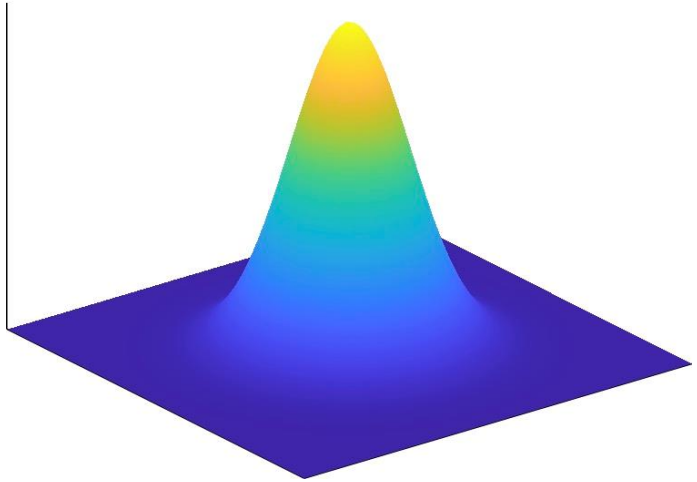
The extension of a Gaussian distribution to functions

Gaussians

- (Univariate) Gaussians:

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





- Multivariate Gaussians:

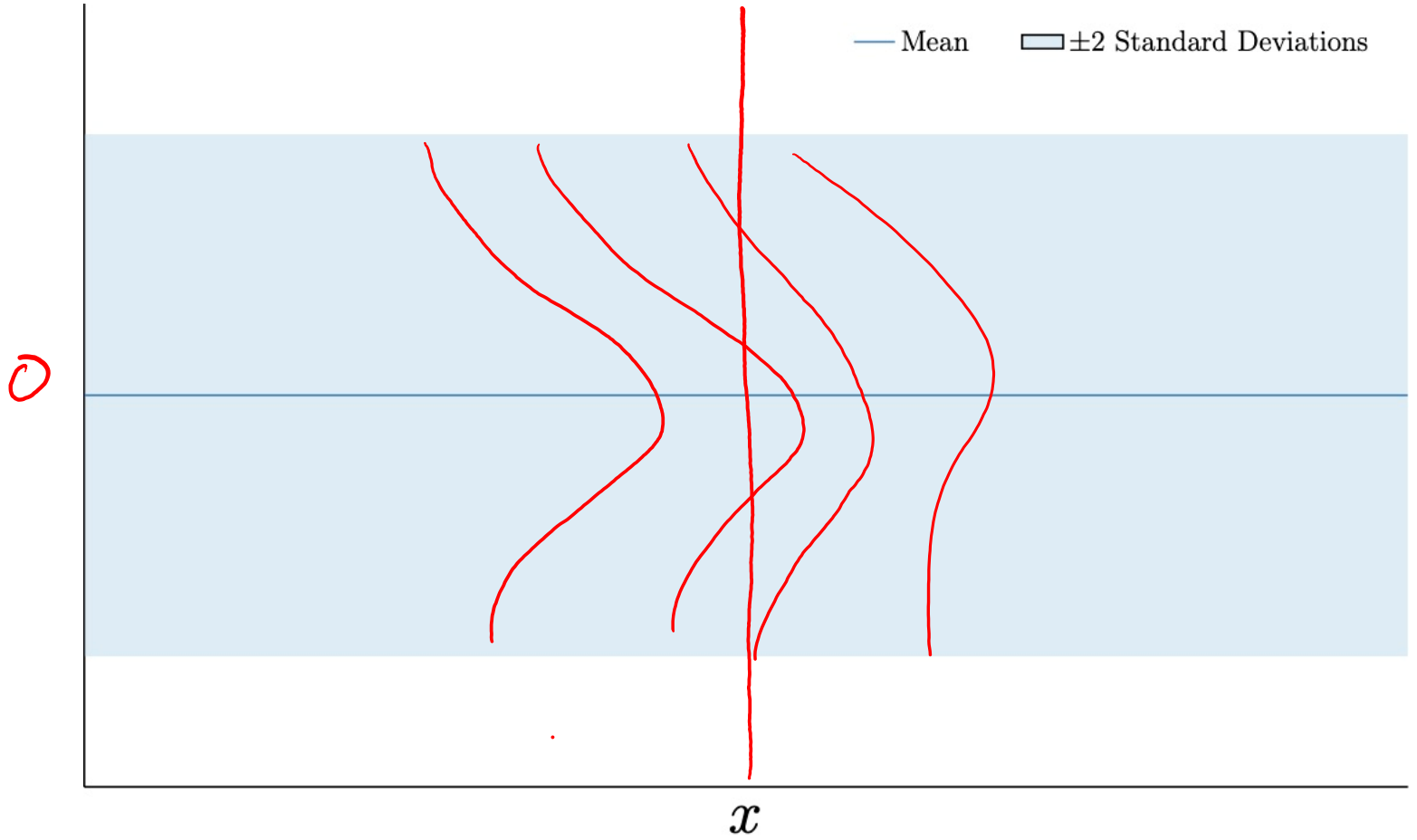
$$\vec{x} = \left[x_1, \ldots, x_p\right]^T$$

$$\sim \mathcal{N}\left(\vec{x}; \vec{\mu} = \vec{0}_p, \Sigma = I_p\right)$$

squared exponential

Gaussian Process (GP)

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

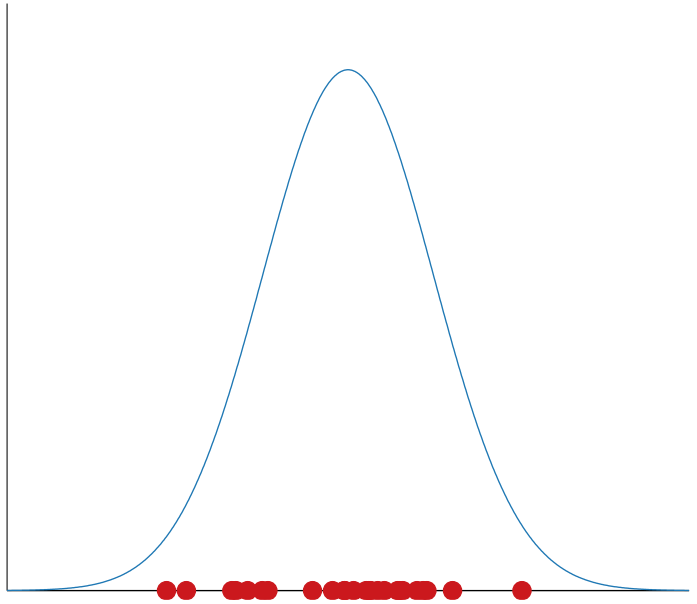


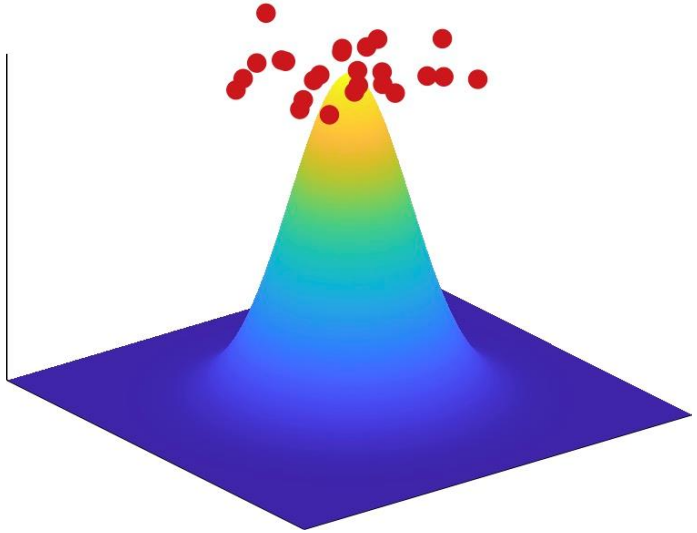
$$f \sim \mathcal{GP}(\mu, \Sigma) \rightarrow f(x) \sim \mathcal{N}\big(\mu(x), \Sigma(x, x)\big)$$

Gaussians

- (Univariate) Gaussians:

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





- Multivariate Gaussians:

$$\vec{x} = \left[x_1, \ldots, x_p\right]^T$$

$$\sim \mathcal{N}\left(\vec{x}; \vec{\mu} = \vec{0}_p, \Sigma = I_p\right)$$

Gaussian Process (GP)

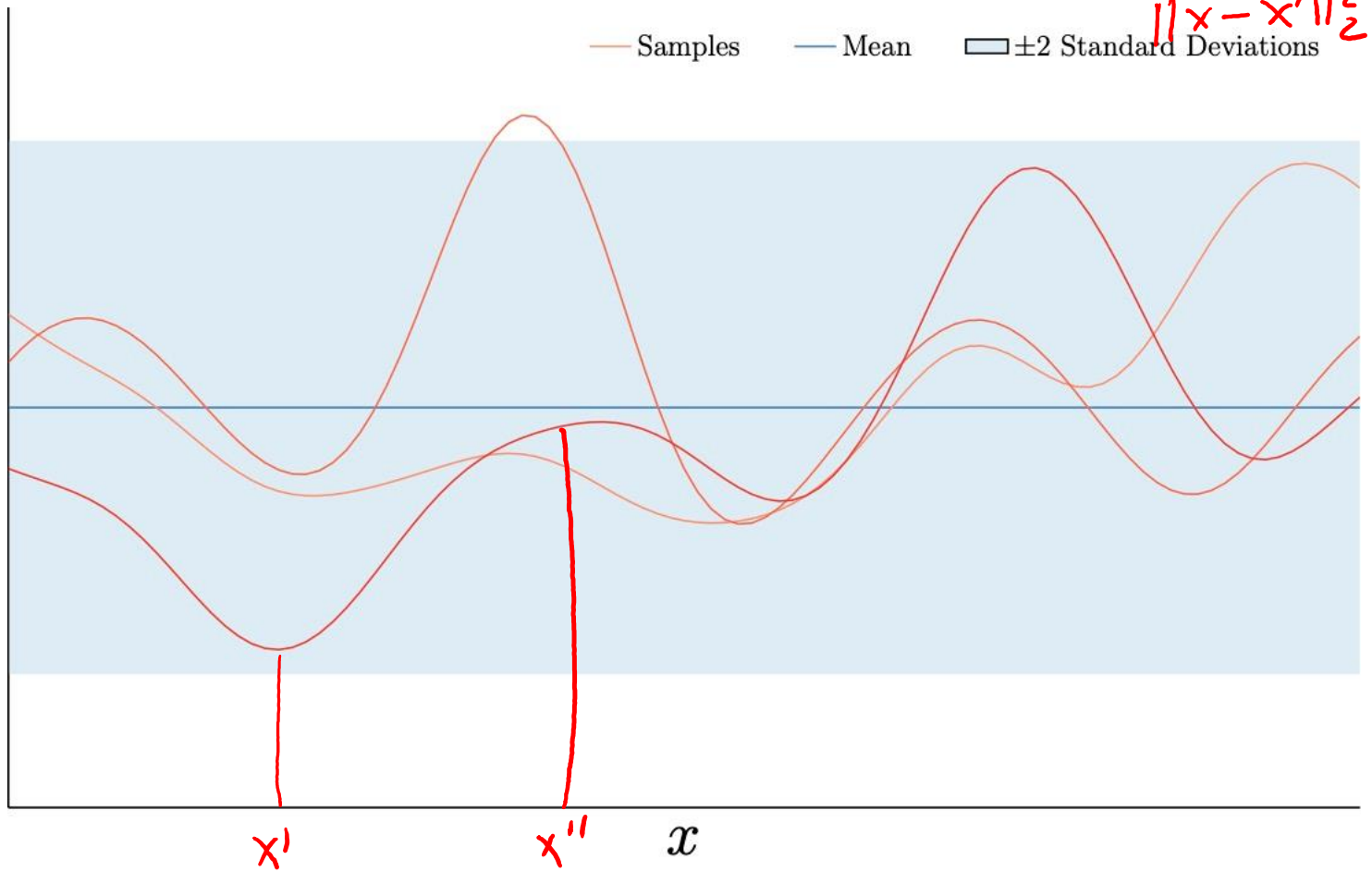
$$f: \mathbb{R}^p \mapsto \mathbb{R} \sim \mathcal{GP}\left(f; \mu(x) = 0, \Sigma(x, x') = \exp\left(-(x - x')^2\right)\right)$$

$$\frac{\partial f}{\partial x} \sim \mathcal{GP}\left(\frac{\partial \mu}{\partial x}, \frac{\partial^2}{\partial x \partial x'} \Sigma\right)$$

if x and x' are far apart ≈ 0

$\|x - x'\|_2^2$

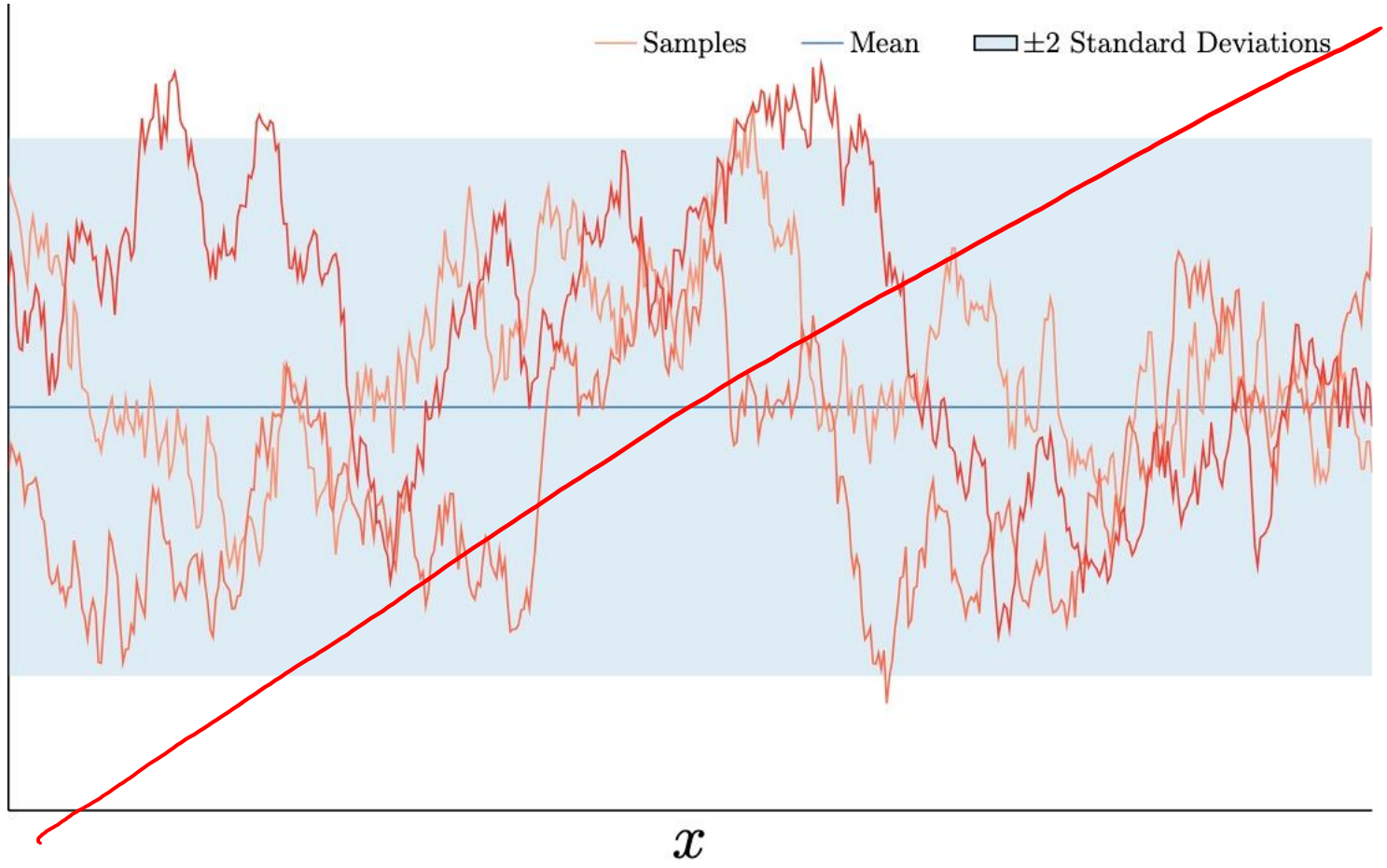
$\Sigma(x', x'')$ = covariance of $f(x')$ and $f(x'')$



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

Gaussian Process (GP)

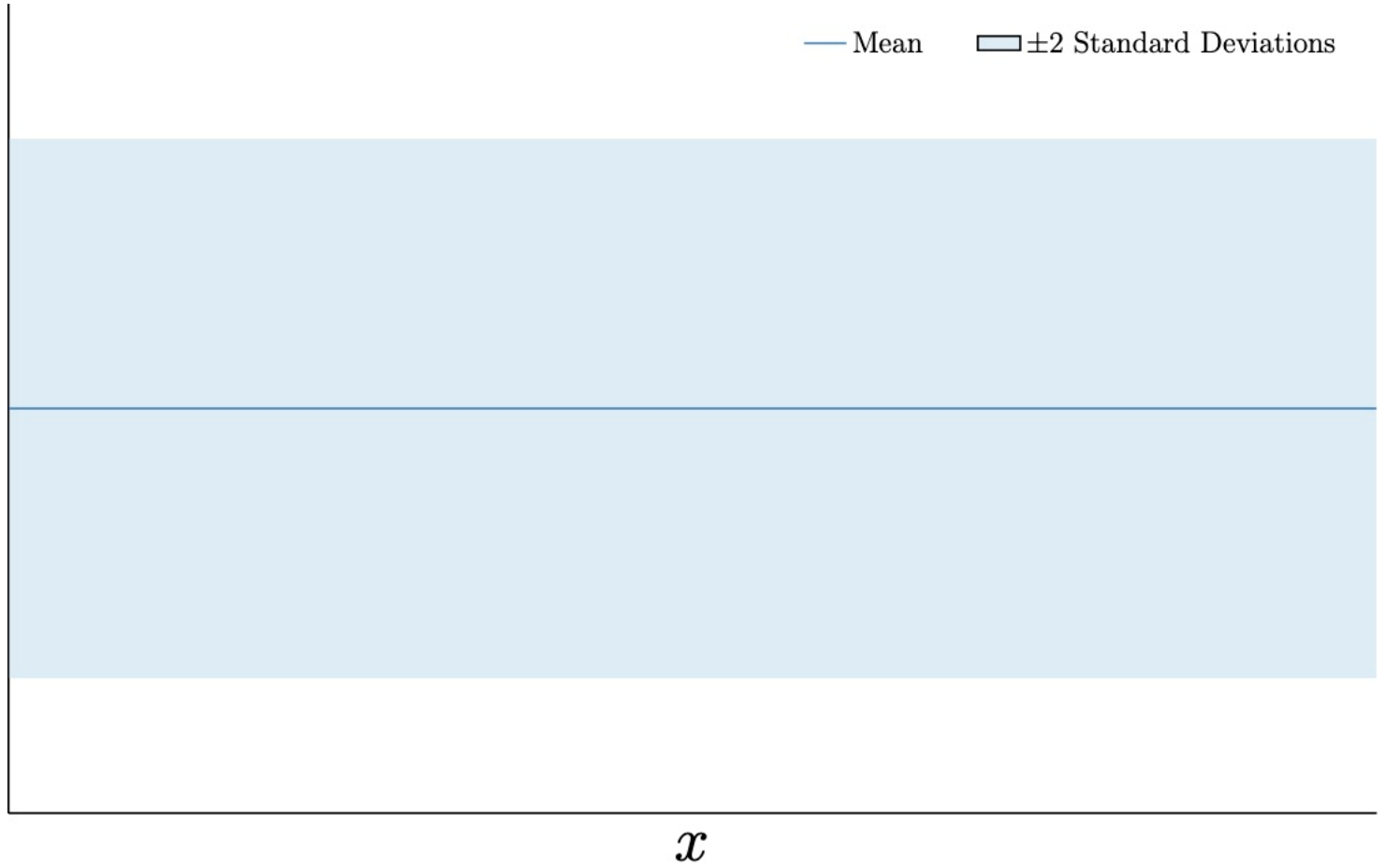
$$f: \mathbb{R}^p \mapsto \mathbb{R} \sim \mathcal{GP}(f; \mu(x) = 0, \Sigma(x, x') = \exp(-|x - x'|))$$



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

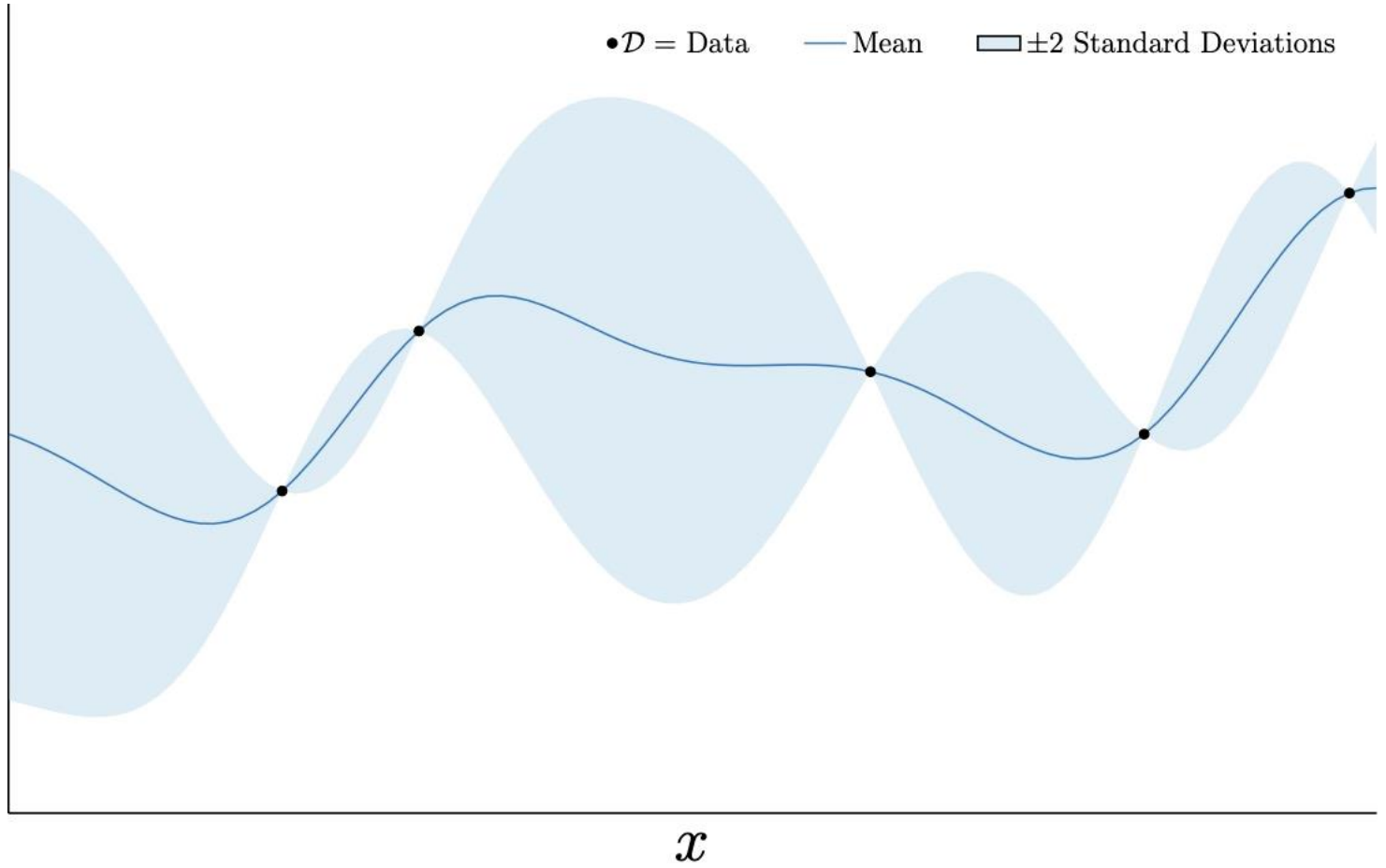
GP Prior

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



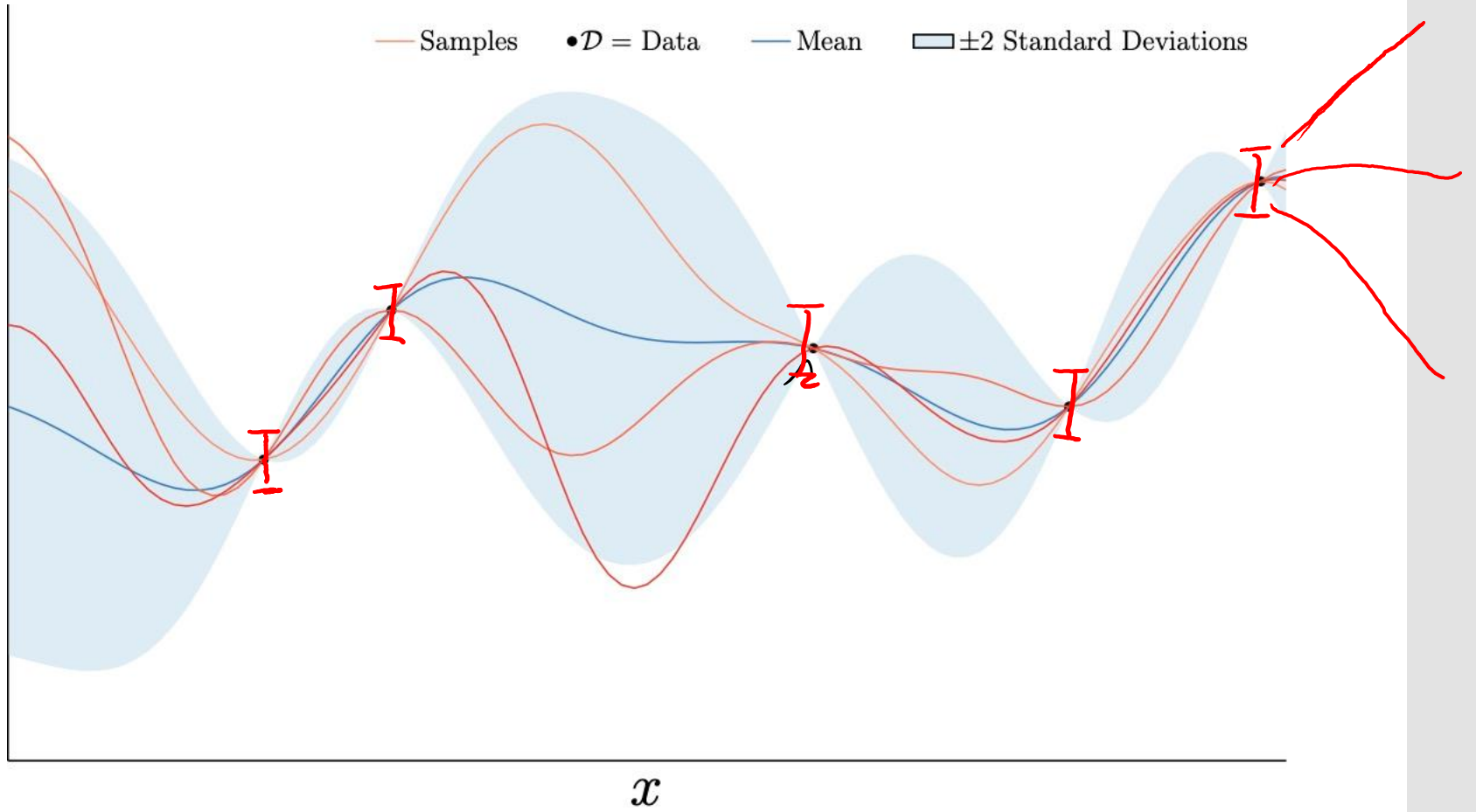
GP Posterior

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



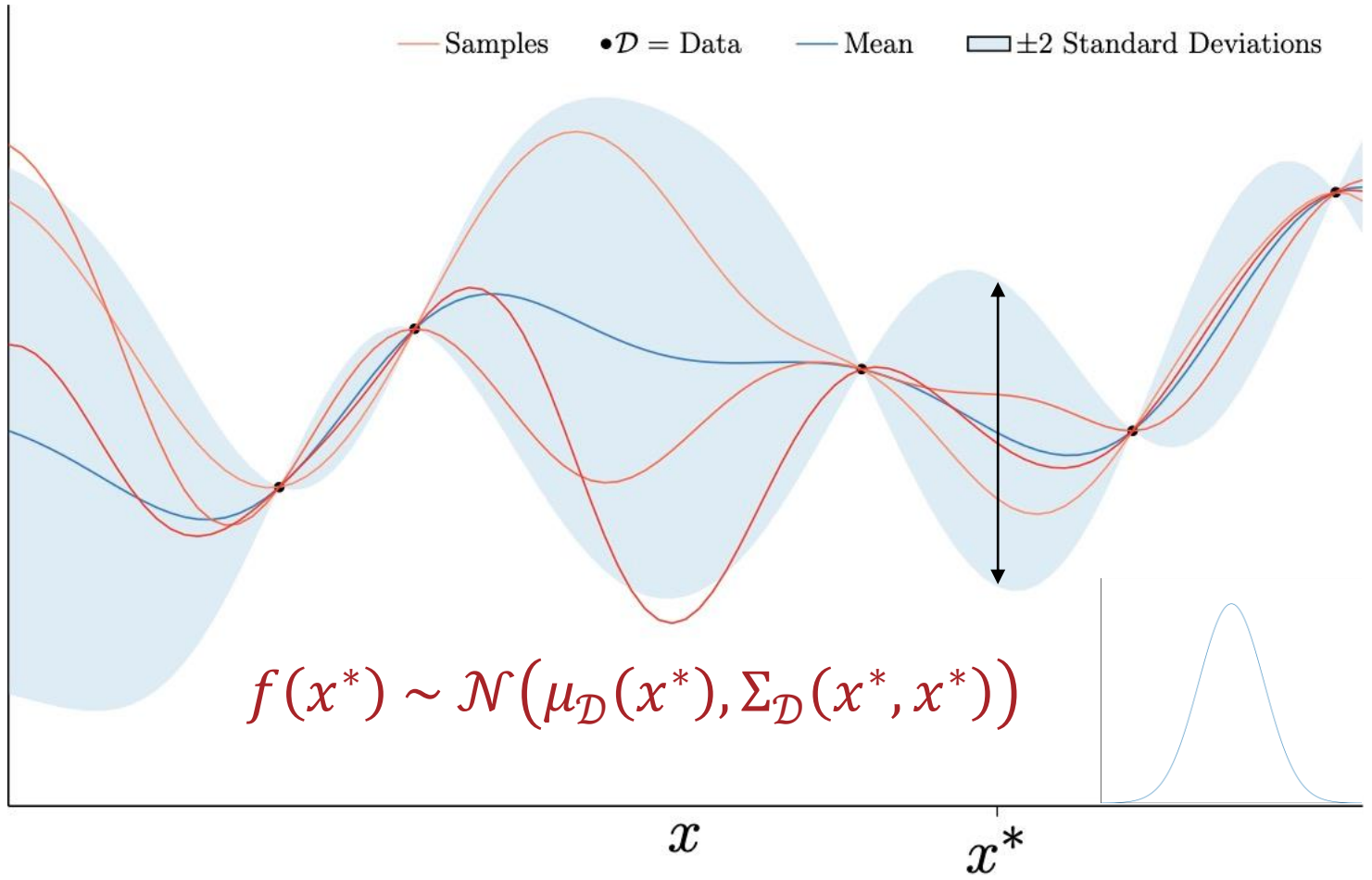
GP Posterior

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

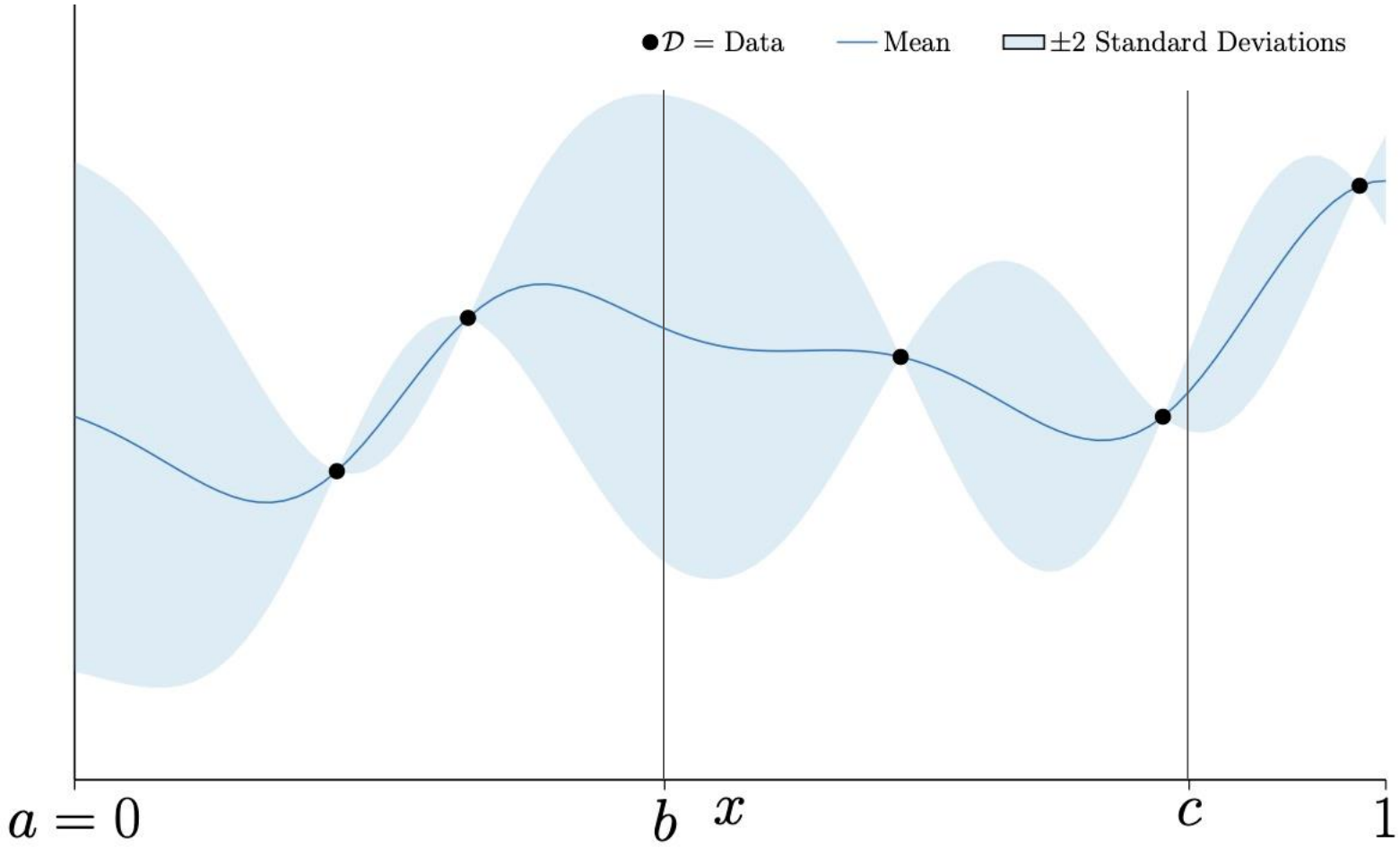


GP Posterior

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

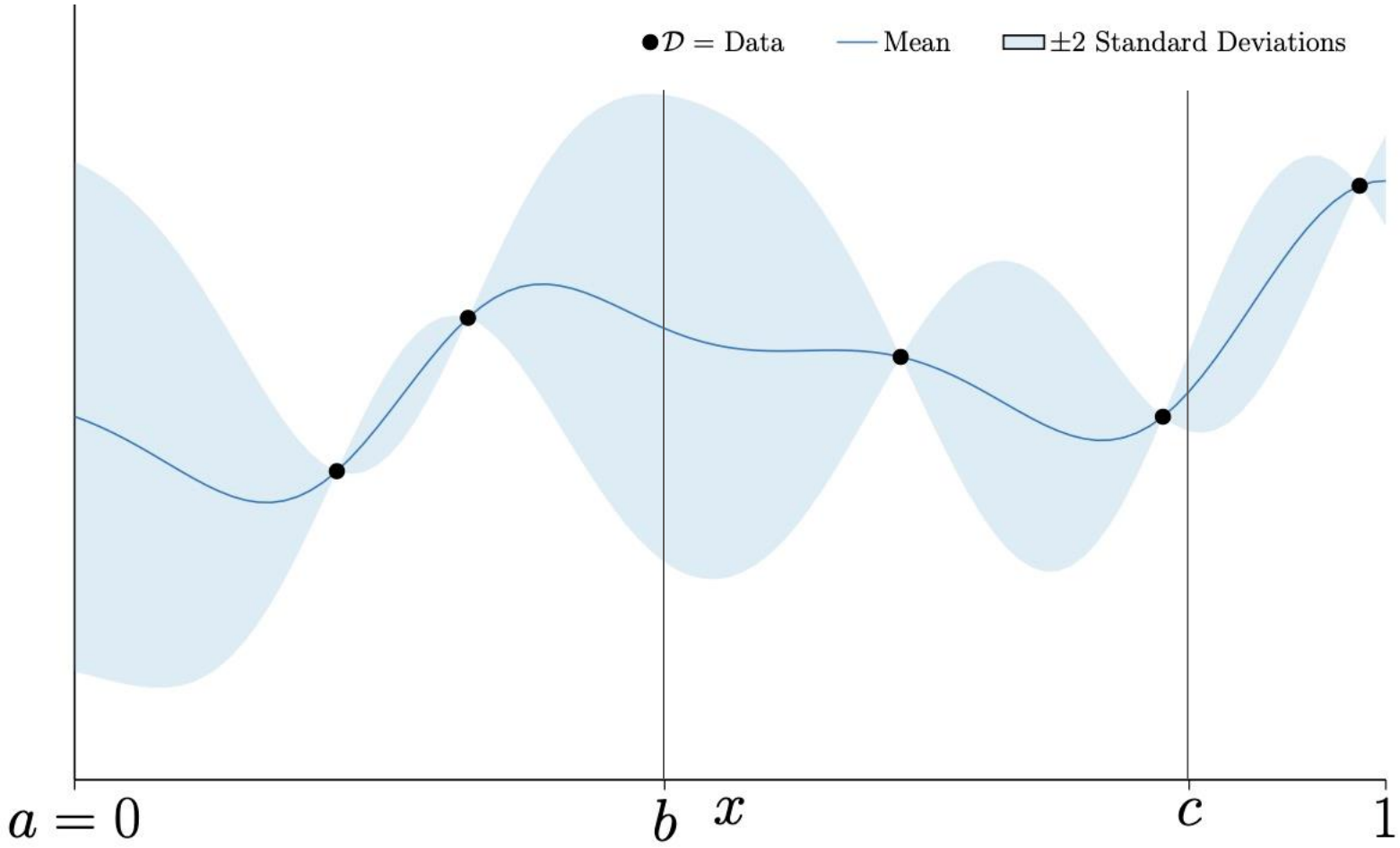


Active Learning



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

Which point would you add and why?

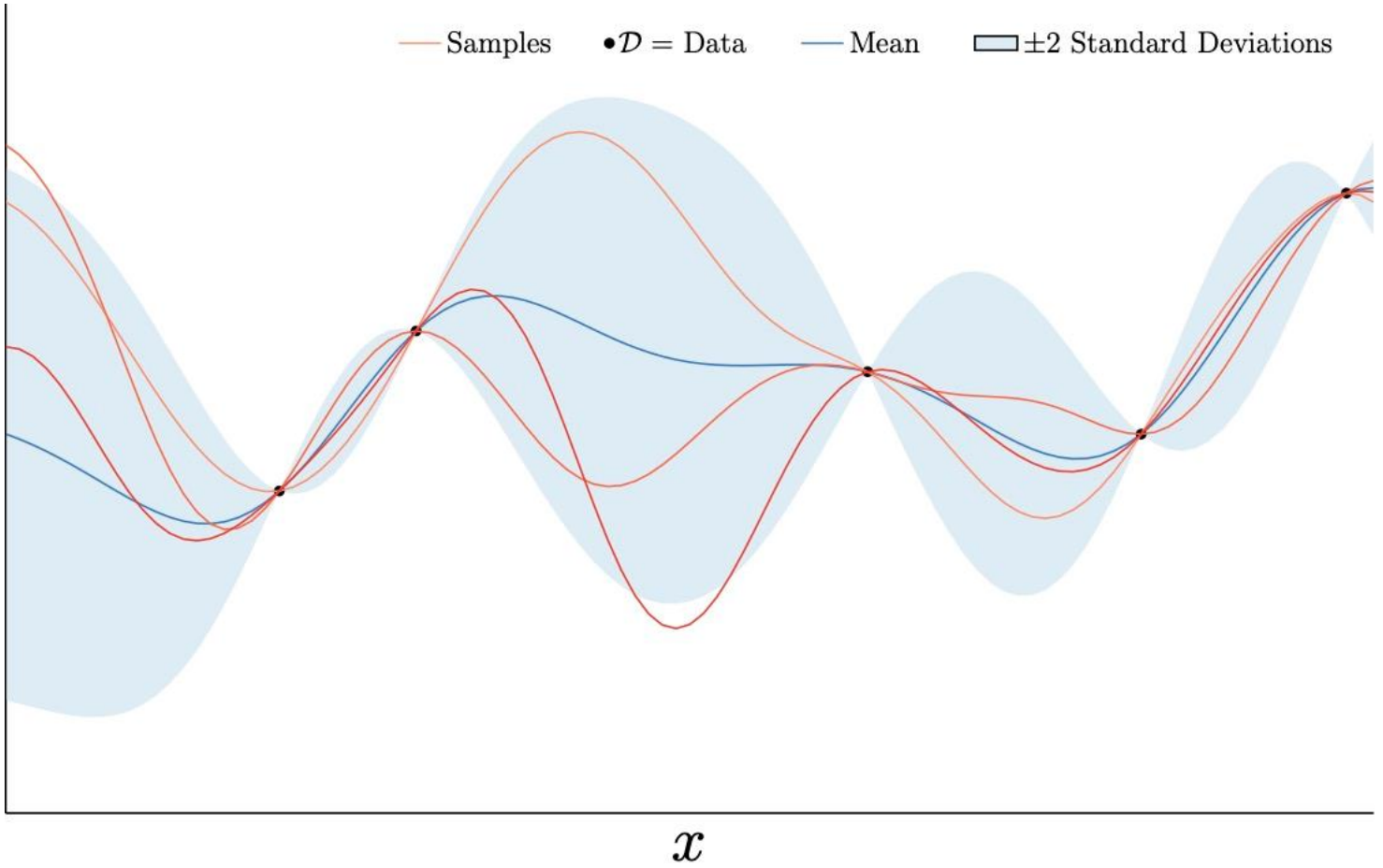


Are GPs:

1. parametric or nonparametric

2. generative or discriminative

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

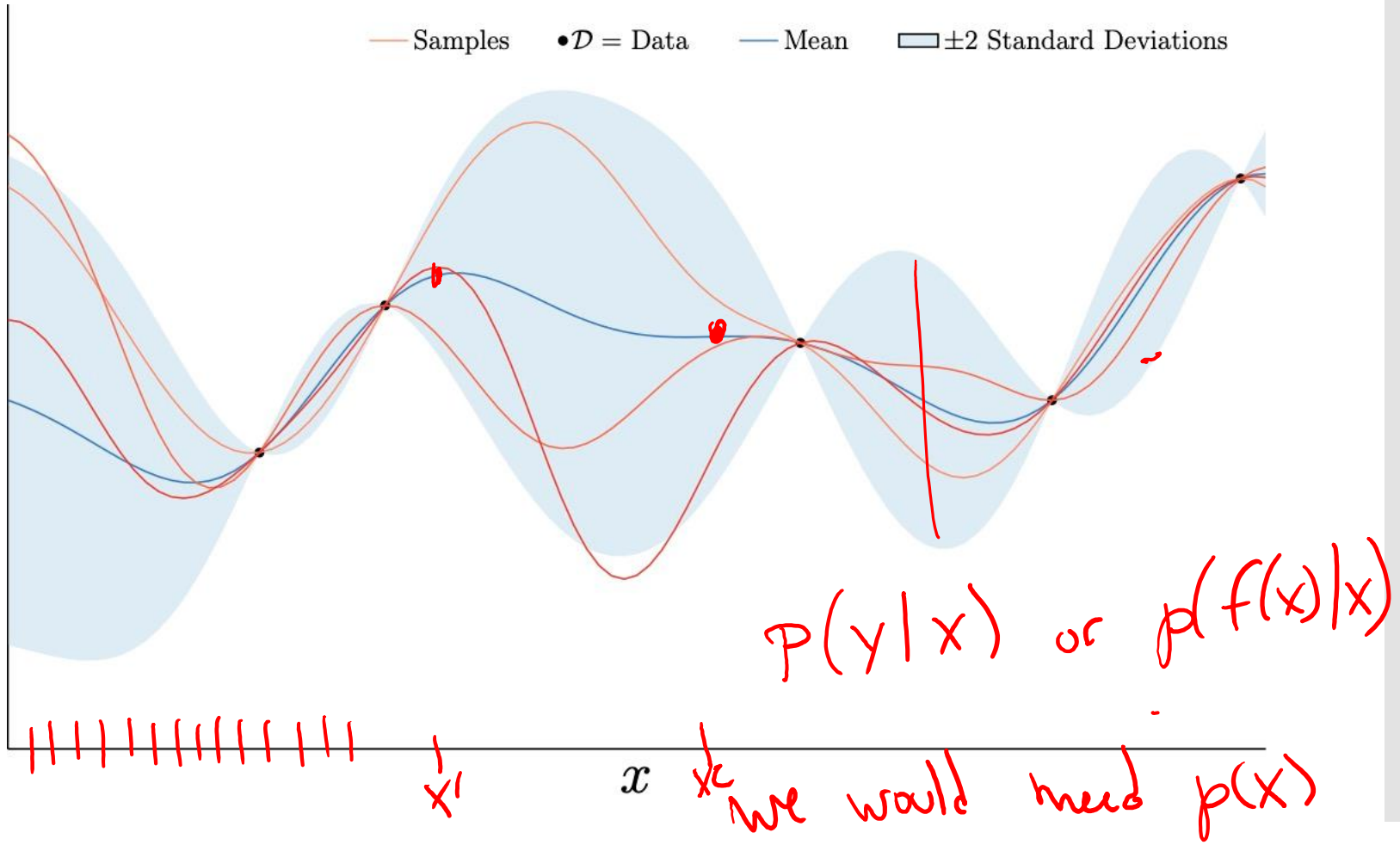


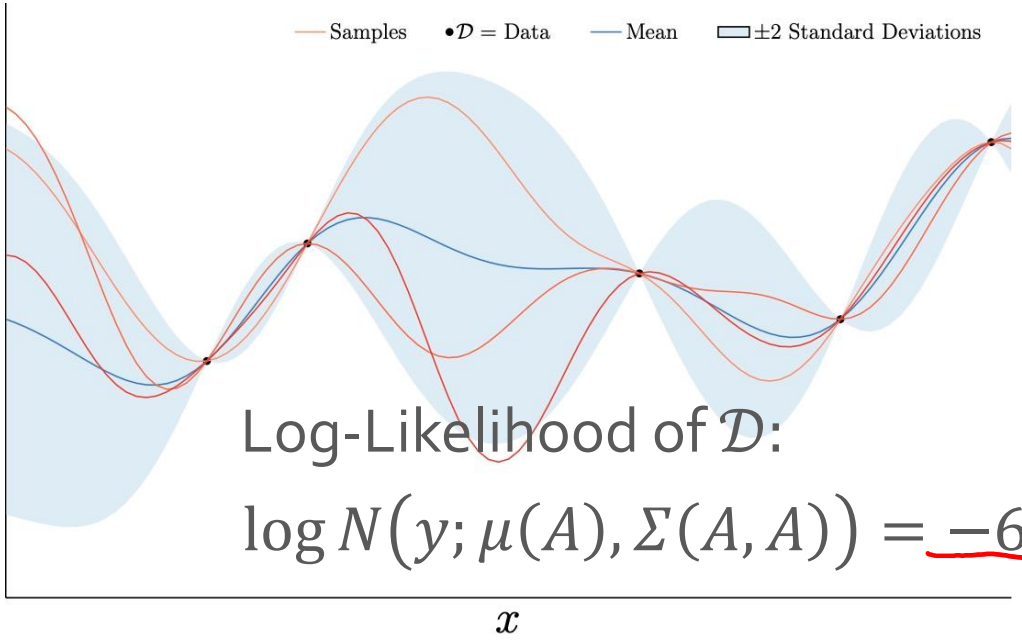
GPs are

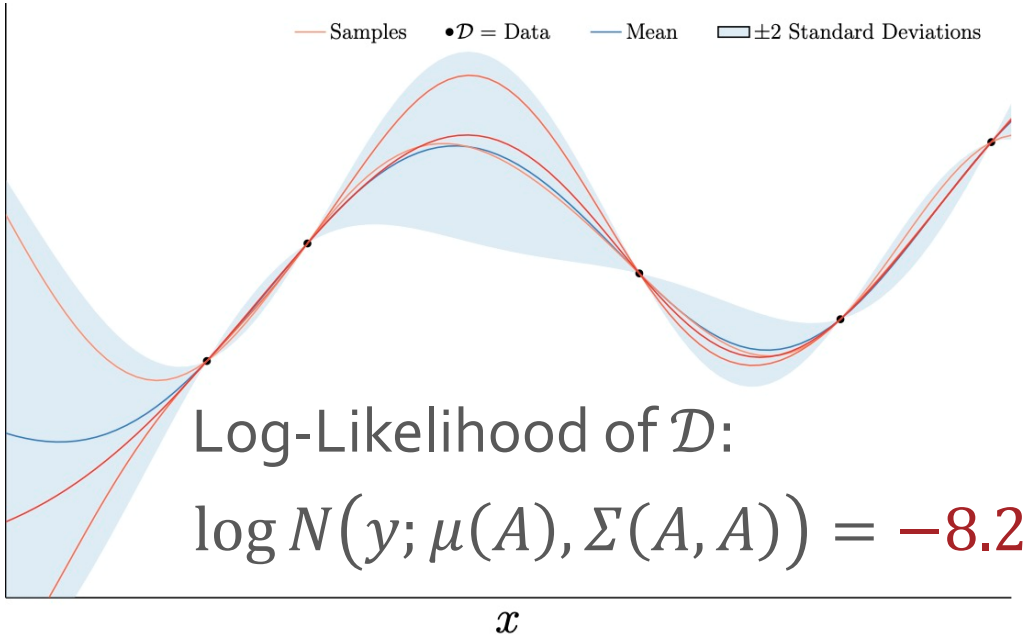
1. parametric or nonparametric

2. generative or discriminative

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







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

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

Kernel Hyperparameters