WHEN TRUST MATTERS

Introduction to **Gaussian Processes** for surrogate modelling

HIPERWIND PhD Summer School

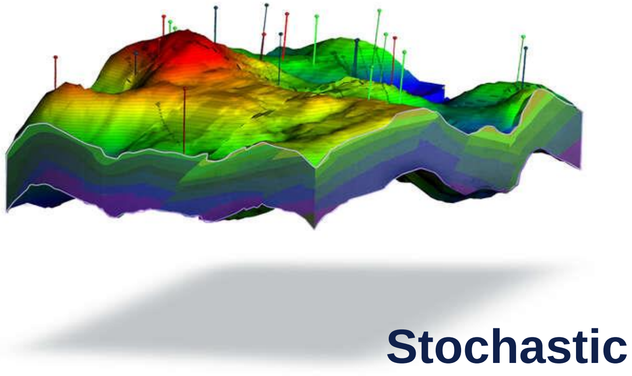
Christian Agrell

29 August 2023

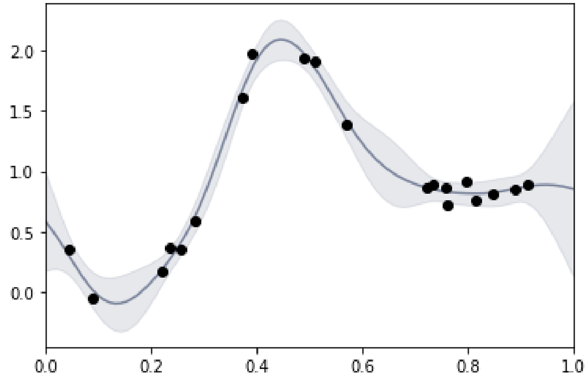
Gaussian processes appear in many different fields

Geostatistics

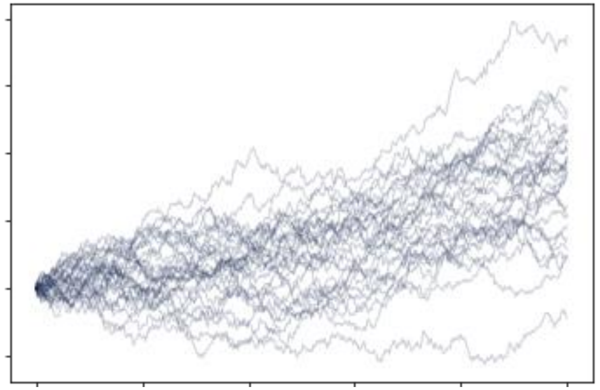
(Kriging)



Machine Learning

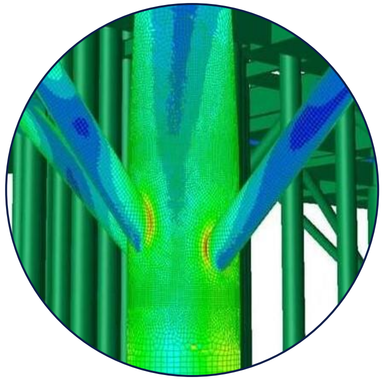


Stochastic Differential Equations



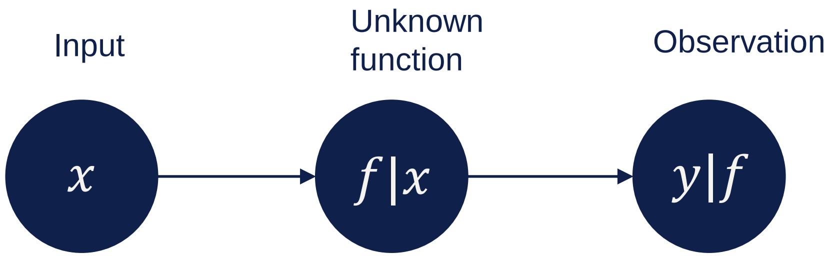
Uncertainty Quantification

Computer experiment



Laboratory experiment

Bayesian nonparametric function estimation

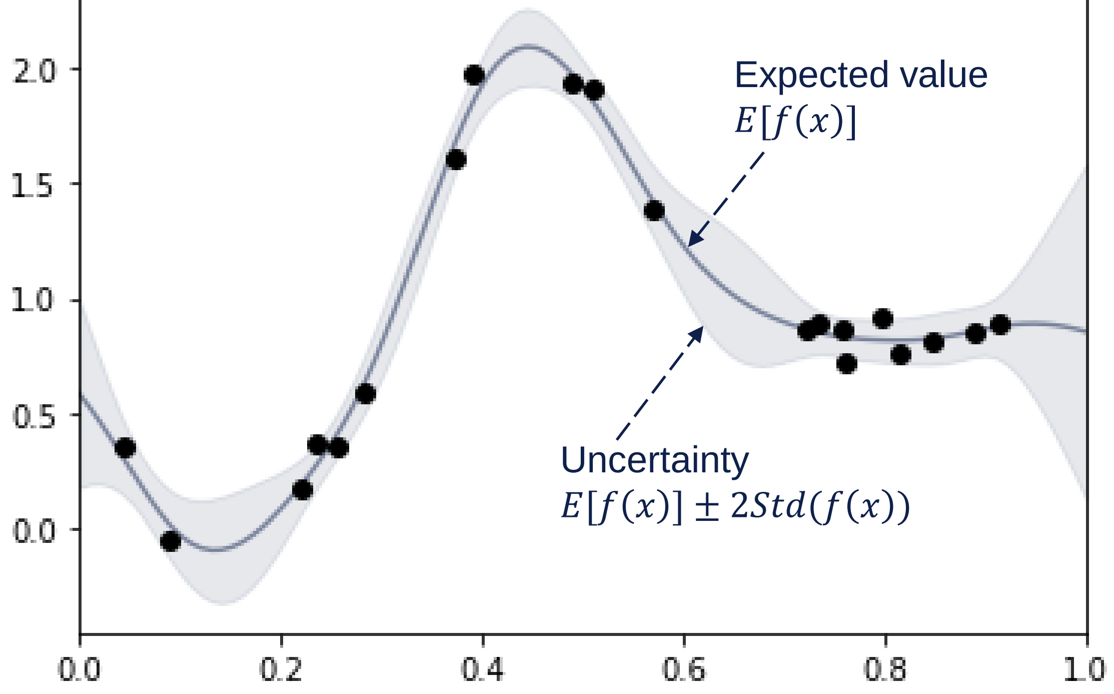


Goal

Infer the function $f(x)$, given a set of observations $\mathrm{D} = \{(x_1, y_1), \dots, (x_n, y_n)\}$

Canonical case

Input and output: $x \in \mathbb{R}^d$, $f(x) \in \mathbb{R}$, $y \in \mathbb{R}$
Observations: $y_i = f(x_i) + \varepsilon_i$, ε_i = noise



Preliminary
The Gaussian conditional distribution

Preliminary:
Multivariate Gaussian conditional distribution

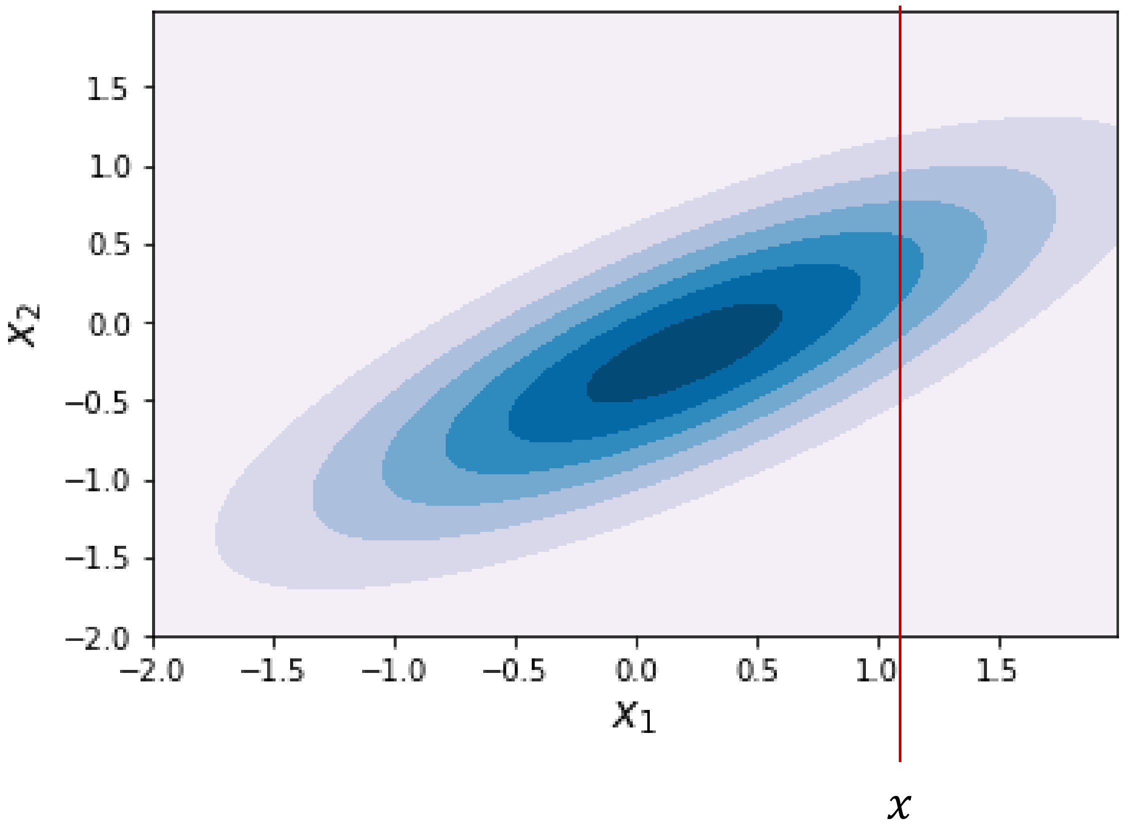
Let X_1 and X_2 be Gaussian random variables with means μ_1 and μ_2, variances σ_1^2 and σ_2^2, and correlation ρ.

- You observe $X_2 = x$.

Then $X_1|X_2 = x$ is Gaussian with mean and variance:

$$E[X_1|X_2 = x] = \mu_1 + \frac{\sigma_1}{\sigma_2}\rho(x - \mu_2)$$

$$Var[X_1|X_2 = x] = (1 - \rho^2)\sigma_1^2$$



Preliminary: **Multivariate Gaussian conditional distribution**

Let $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ be Gaussian <u>vectors</u>, with joint mean and covariance

$$\boldsymbol{\mu} = E\begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}$$

$$\boldsymbol{\Sigma} = COV\begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}$$

$$[\boldsymbol{X}_1, \boldsymbol{X}_2] = [\ \underbrace{X_{11}, \dots, X_{1m},}_{N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11})}\quad \underbrace{X_{21}, \dots, X_{2n}}_{N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22})}\ ]$$

- You observe $\boldsymbol{X}_1 = \boldsymbol{x}$.

Predict

$$\begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{bmatrix}$$

Observe

$\boldsymbol{X}_1 | \boldsymbol{X}_2 = \boldsymbol{x}$ is Gaussian with mean and variance:

$$E[\boldsymbol{X}_1 | \boldsymbol{X}_2 = \boldsymbol{x}] = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_1)$$

$$Var[\boldsymbol{X}_1 | \boldsymbol{X}_2 = x] = \boldsymbol{\Sigma}_{11} \text{-}\ \boldsymbol{\Sigma}_{12}\ \boldsymbol{\Sigma}_{22}^{-1}\ \boldsymbol{\Sigma}_{21}$$

Constructing a GP

Constructing a Gaussian process (GP)

Definition : Gaussian process

$f(\cdot)$ has a Gaussian process (GP) distribution if for any $n \in \mathbb{N}$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, the joint distribution of $f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n)$ is multivariate normal

$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ — A finite collection from some domain $\mathcal{X}$ (typically $\mathbb{R}^N$)

$\boxed{y_i = f(\boldsymbol{x}_i)}$ — A mapping f

$[y_1, \ldots, y_n]$ — An n-dimensional Gaussian vector

Constructing a Gaussian process (GP)

Definition : Gaussian process

$f(\cdot)$ has a Gaussian process (GP) distribution if for any $n \in \mathbb{N}$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, the joint distribution of $f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n)$ is multivariate normal

- A GP over functions $f : \mathcal{X} \to \mathbb{R}$ is completely specified by it's mean function μ and covariance function (kernel) k, where

$$\mu(\boldsymbol{x}) = E[f(\boldsymbol{x})]$$
$$k(\boldsymbol{x}, \boldsymbol{x}') = cov(f(\boldsymbol{x}), f(\boldsymbol{x}'))$$

and we write $f \sim GP(\mu, k)$.

Definition : Covariance function

A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that is

- Symmetric
- Positive semi-definite

Definition : Symmetric

$$k(x_i, x_j) = k(x_j, x_i)$$

Definition : Positive semi-definite

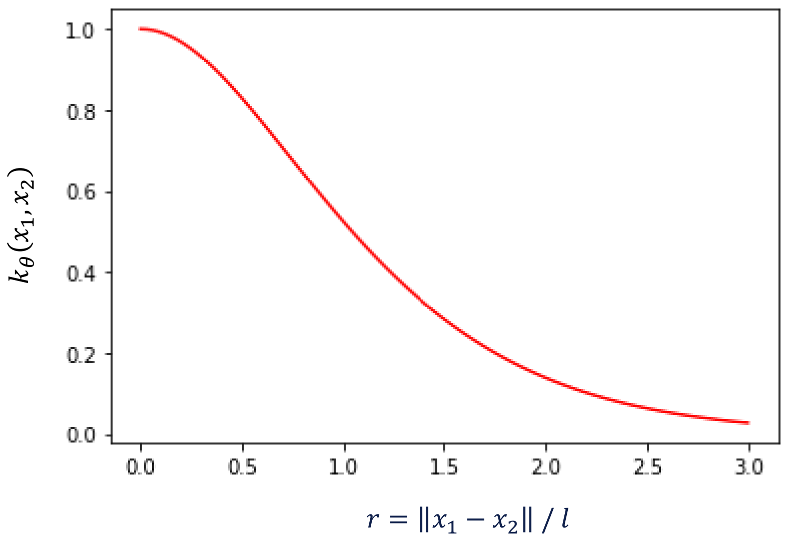
$$\sum_{i,j=1}^{n} \lambda_i \lambda_j \, k(x_i, x_j) \geq 0$$

For all $x_1, \ldots, x_n \in \mathcal{X}$ and $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$

The covariance function (kernel)

Recall: $f \sim GP(\mu, k)$

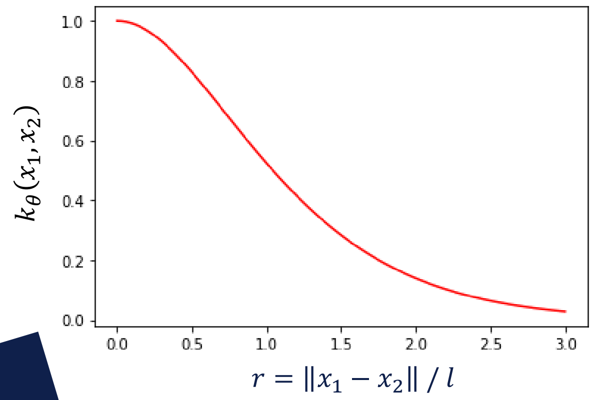
- Assume $\mu = 0$
  (Or that we work with $f - \mu$)
- Assume k is stationary: $k(x_1, x_2)$ can be written as $k(x_1 - x_2)$
  (This is often used in practice)



The covariance function (kernel)

Different covariance functions give different «types» of functions that the GP can represent

- Differentiable
- Periodicity
- Stationary / non-stationary
- Linear trend



Matérn 5/2

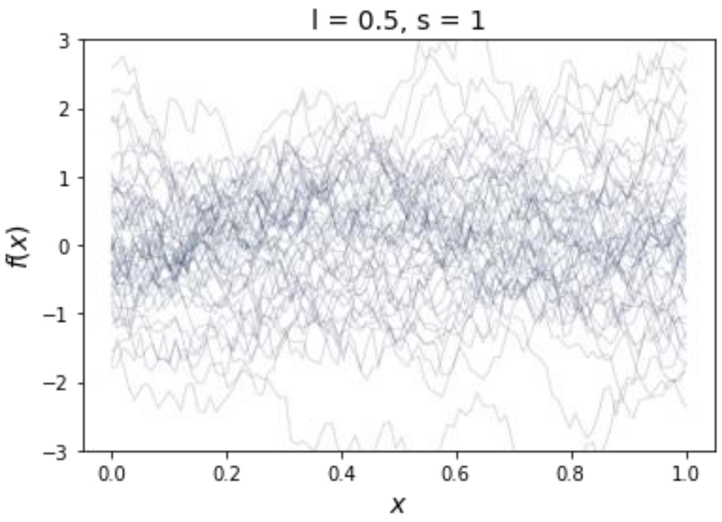
$$k_\theta(x_1, x_2) = s^2\left(1 + \sqrt{5}r + \frac{5}{3}r^2\right)e^{-\sqrt{5}r}$$

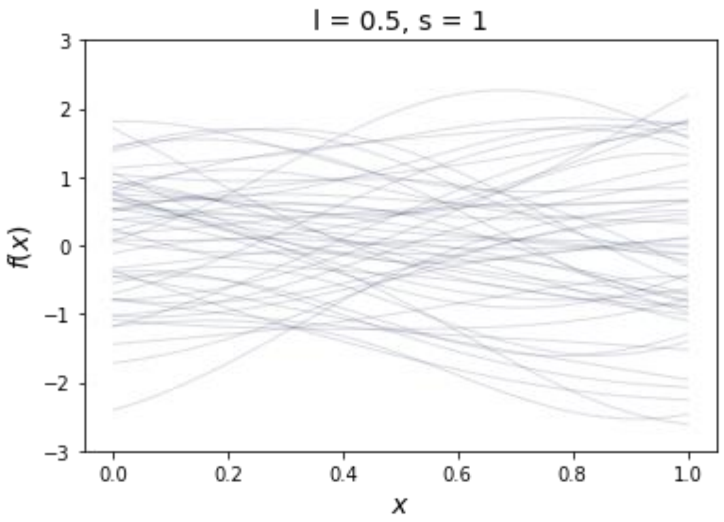
Exponential

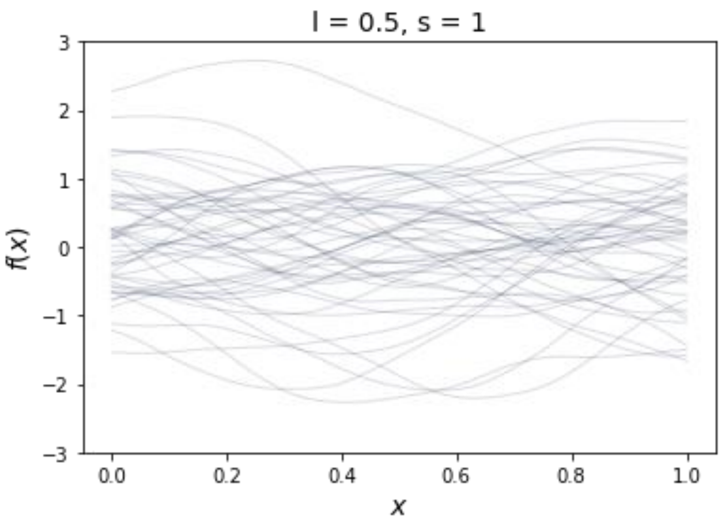
$$k_\theta(x_1, x_2) = s^2 e^{-r}$$

Gaussian

$$k_\theta(x_1, x_2) = s^2 e^{-\frac{1}{2}r^2}$$



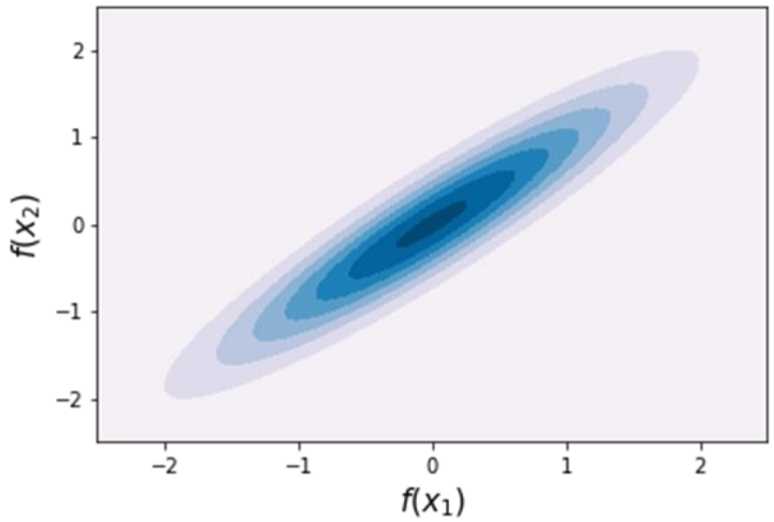




3 ways to think about GPs

3 Ways to think about GPs
1) A large vector

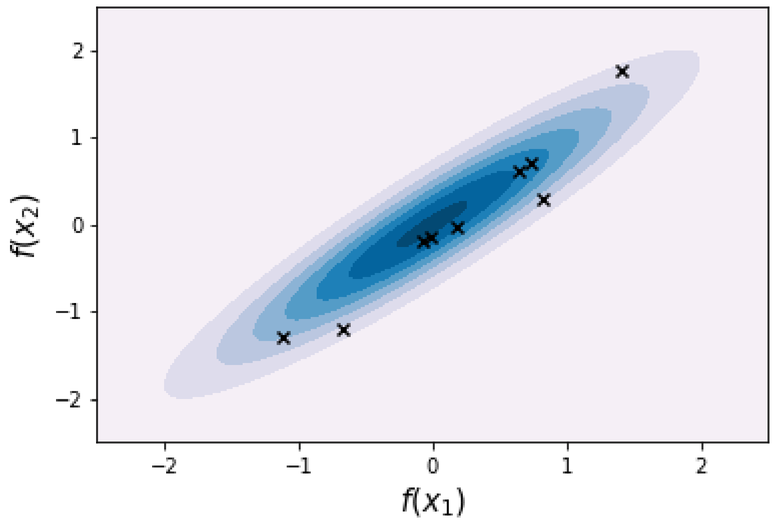
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_N)]$



3 Ways to think about GPs

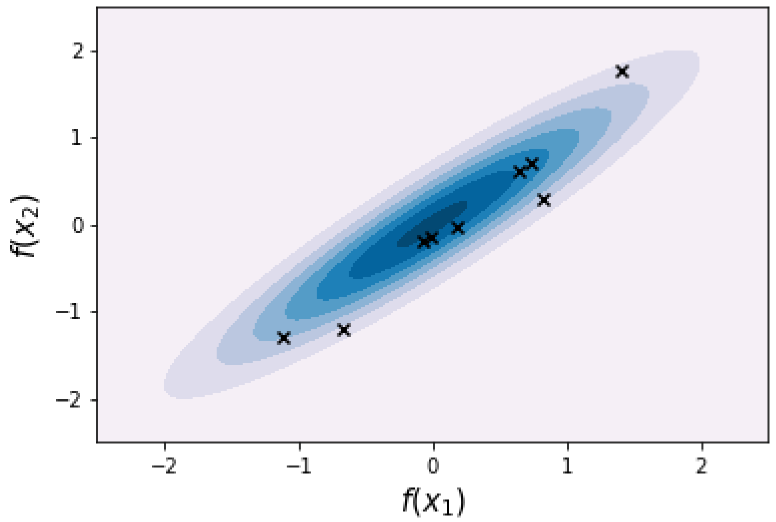
1) A large vector

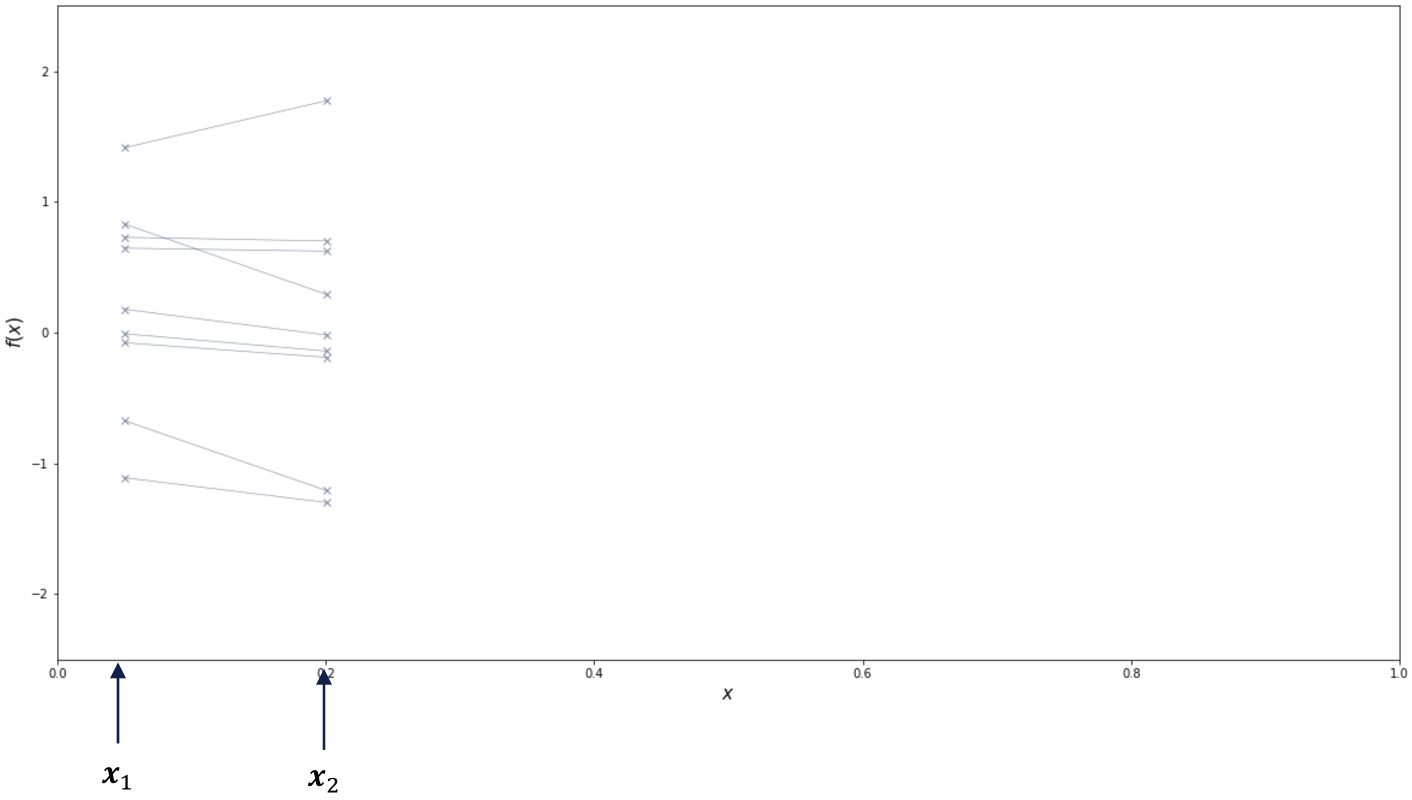
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_N)]$



3 Ways to think about GPs
1) A large vector

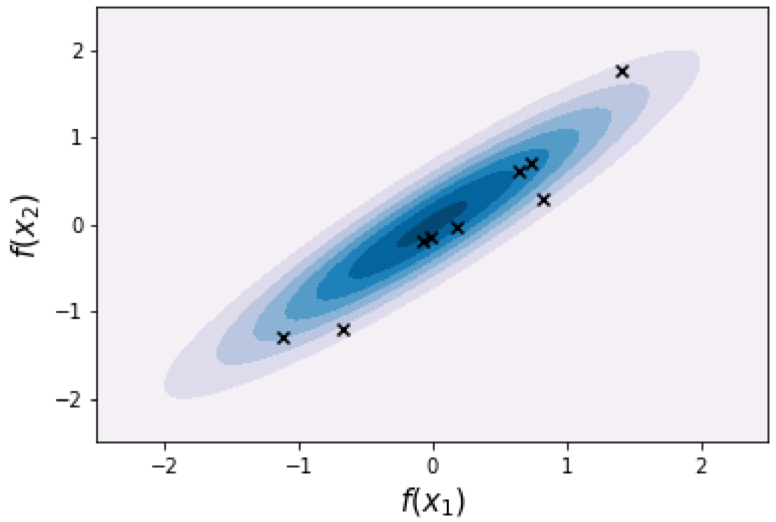
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_N)]$

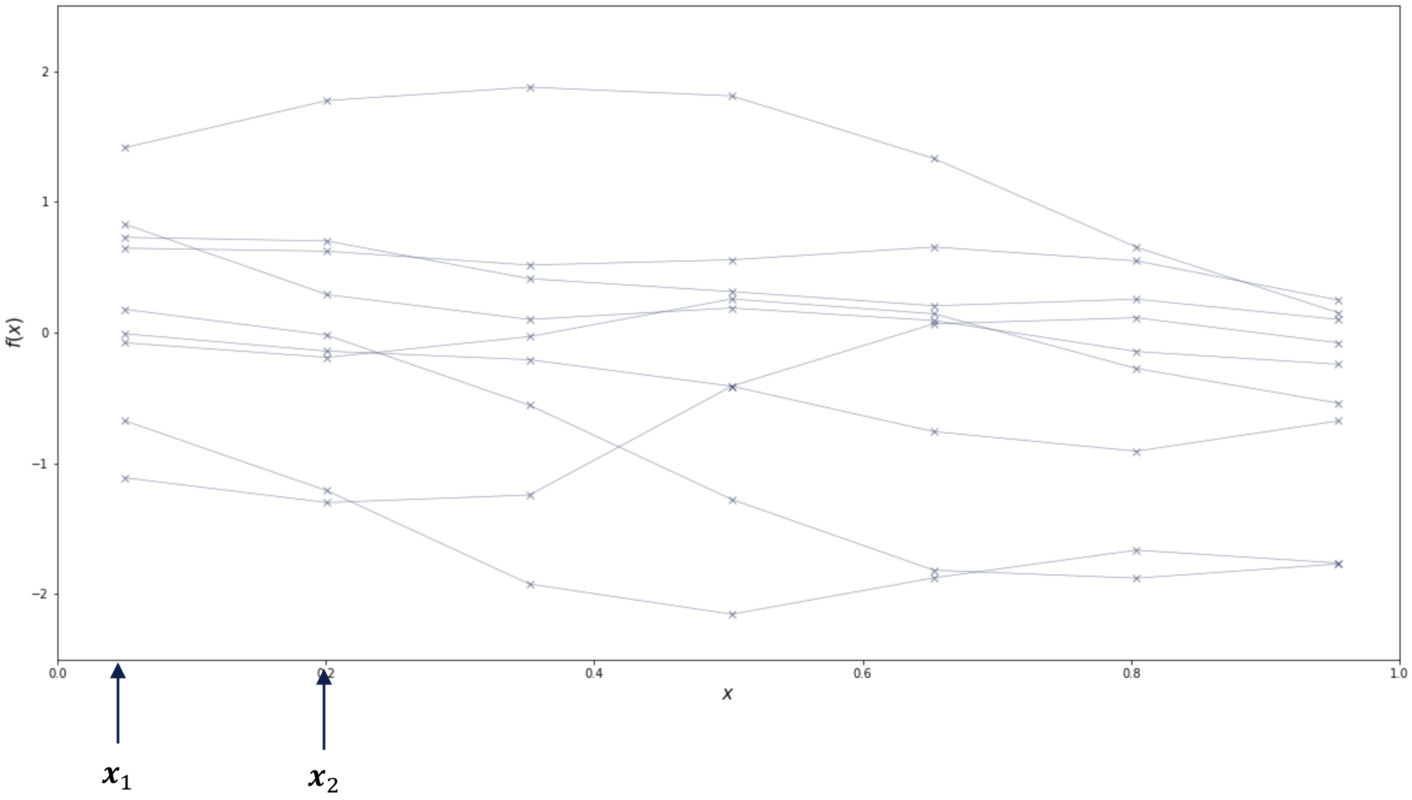




3 Ways to think about GPs
1) A large vector

- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]$

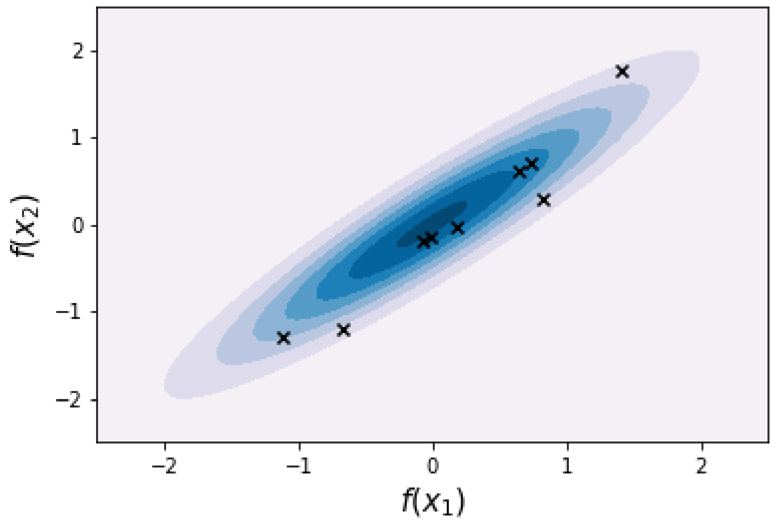


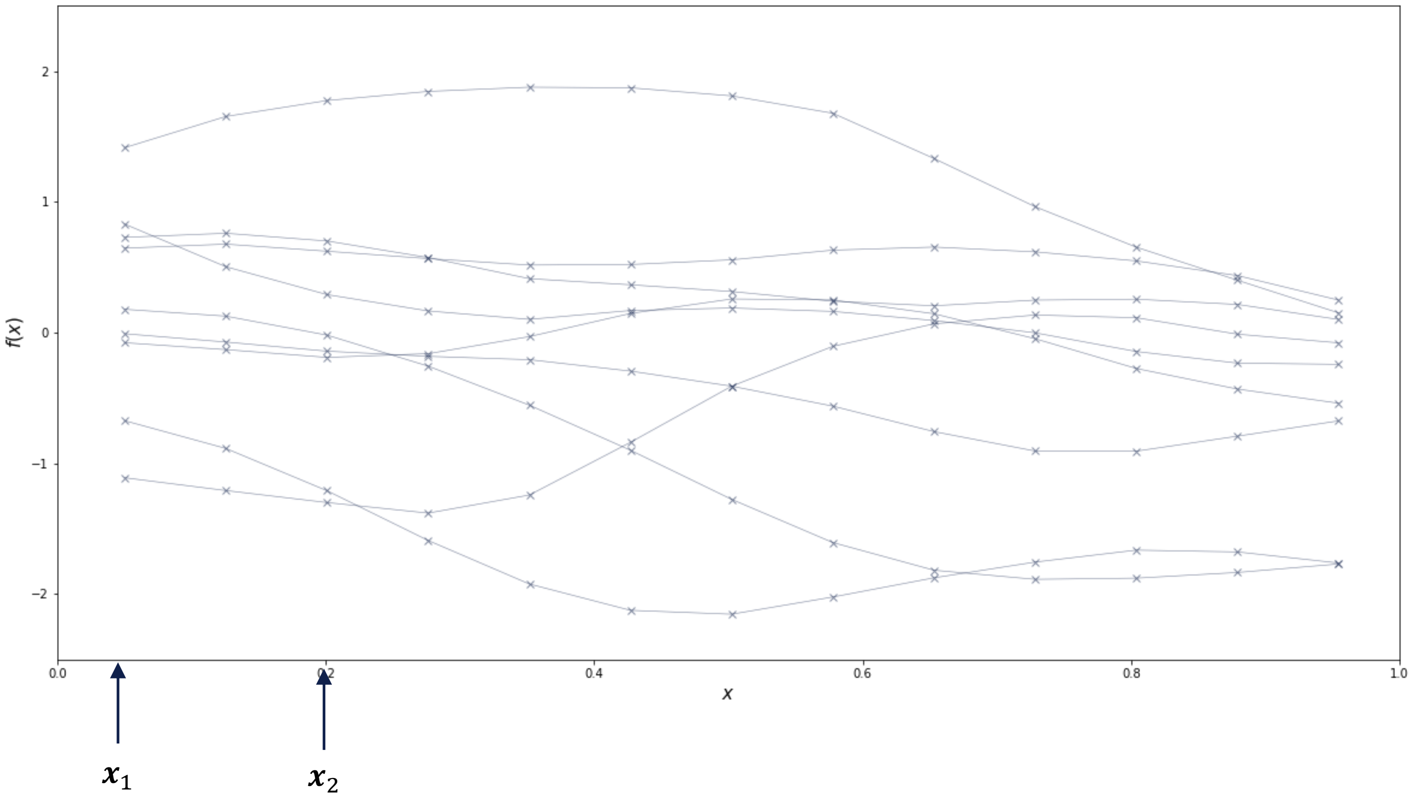


3 Ways to think about GPs

1) A large vector

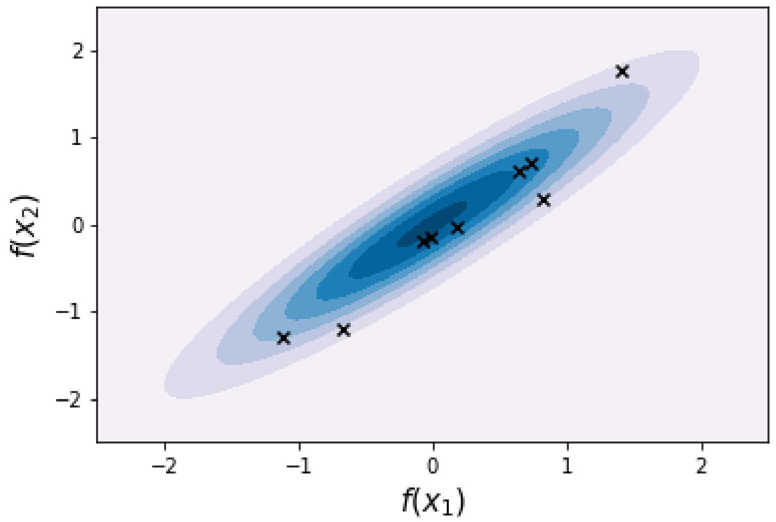
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]$

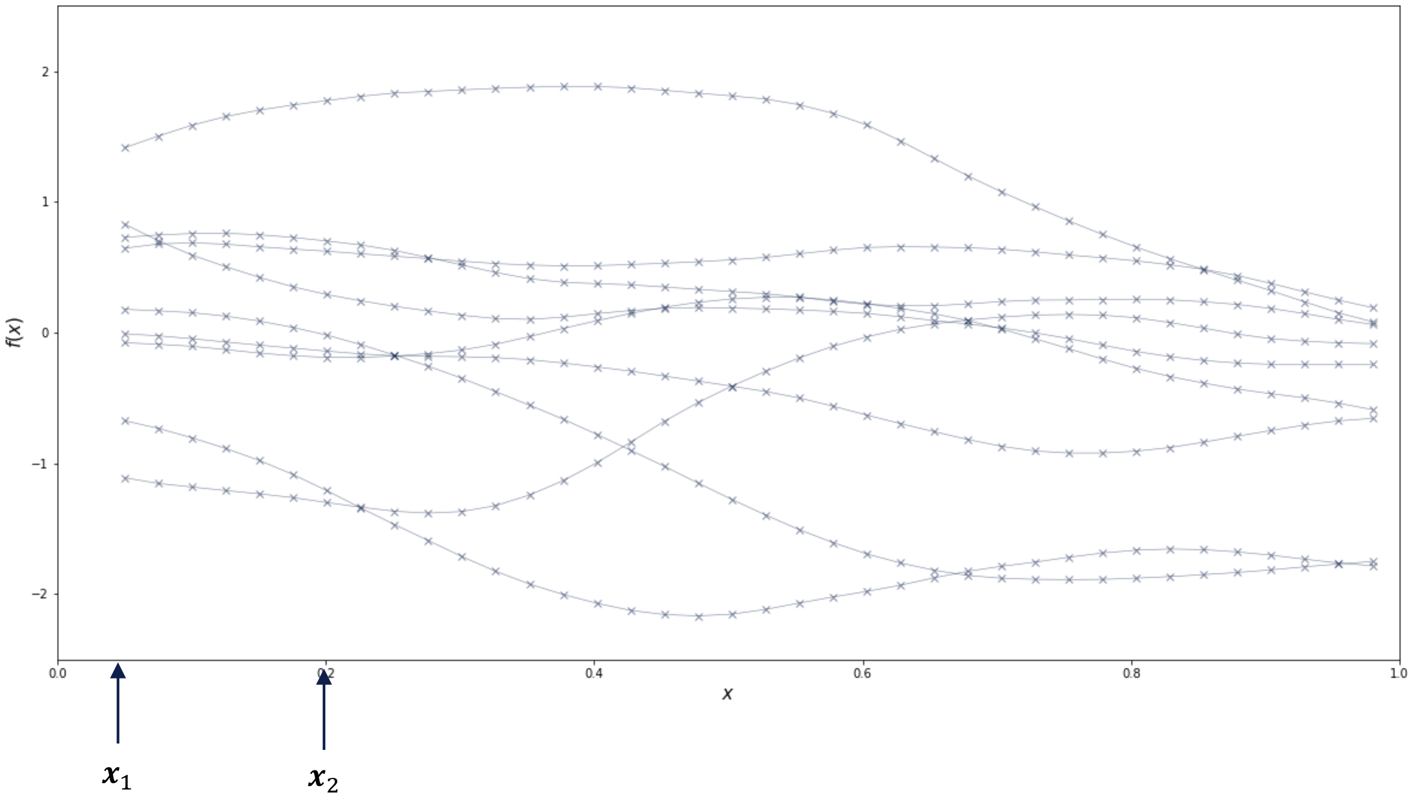




3 Ways to think about GPs
1) A large vector

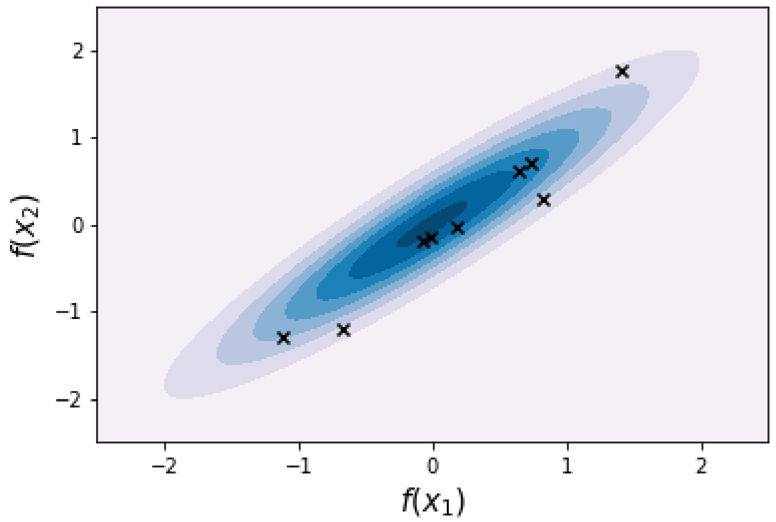
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_N)]$

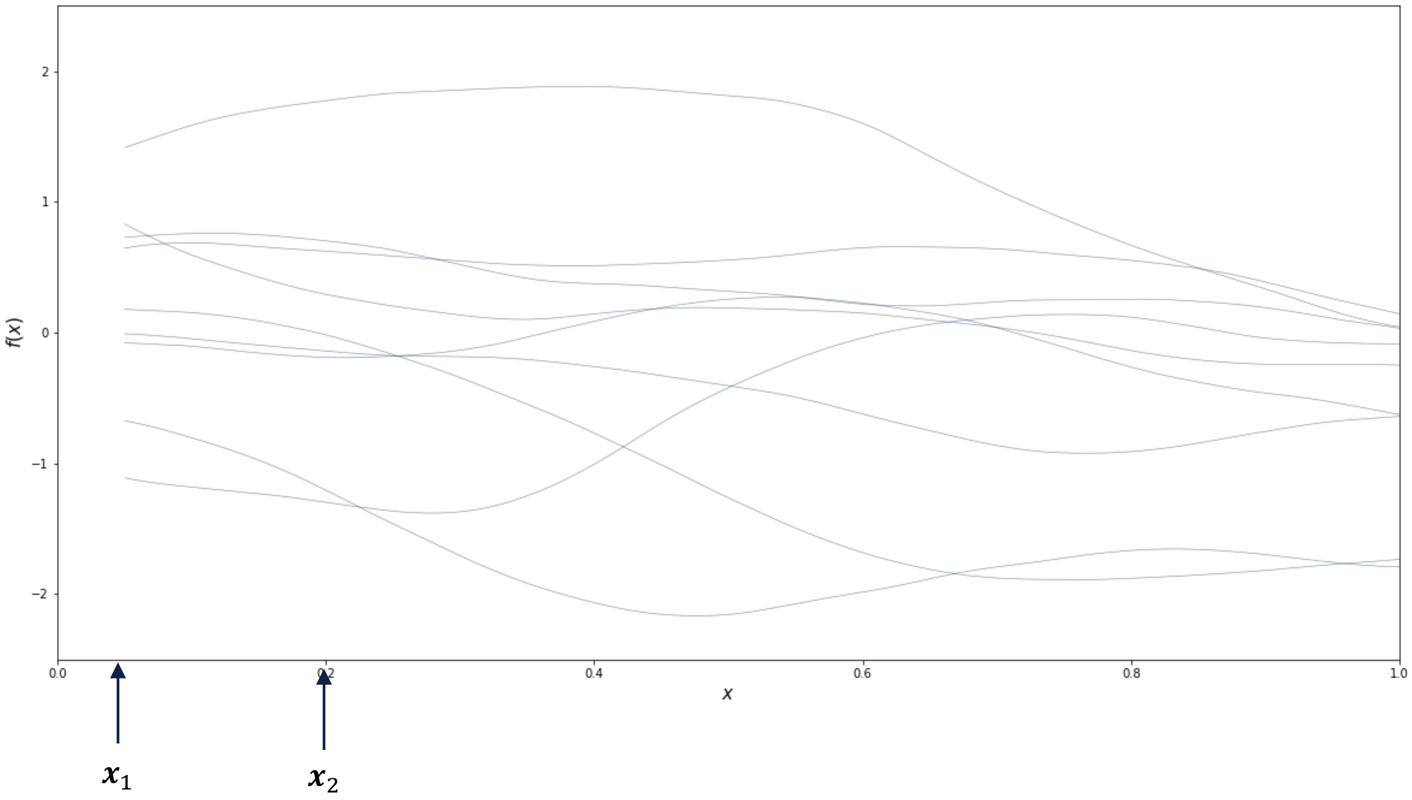




3 Ways to think about GPs
1) A large vector

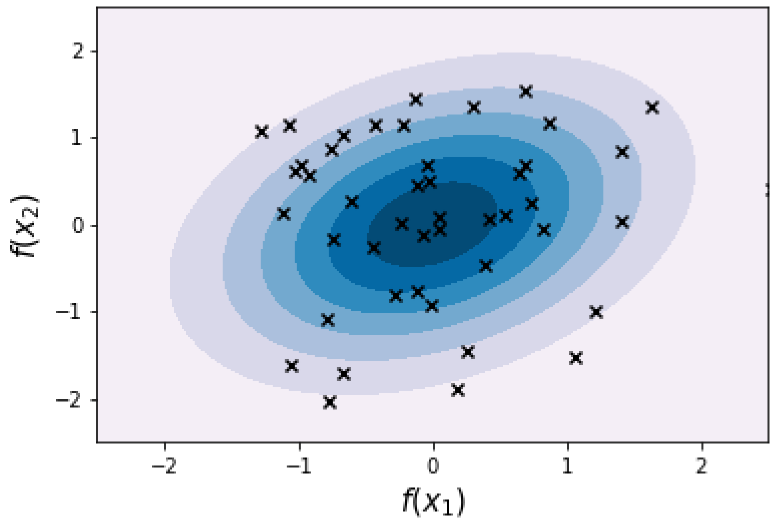
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]$

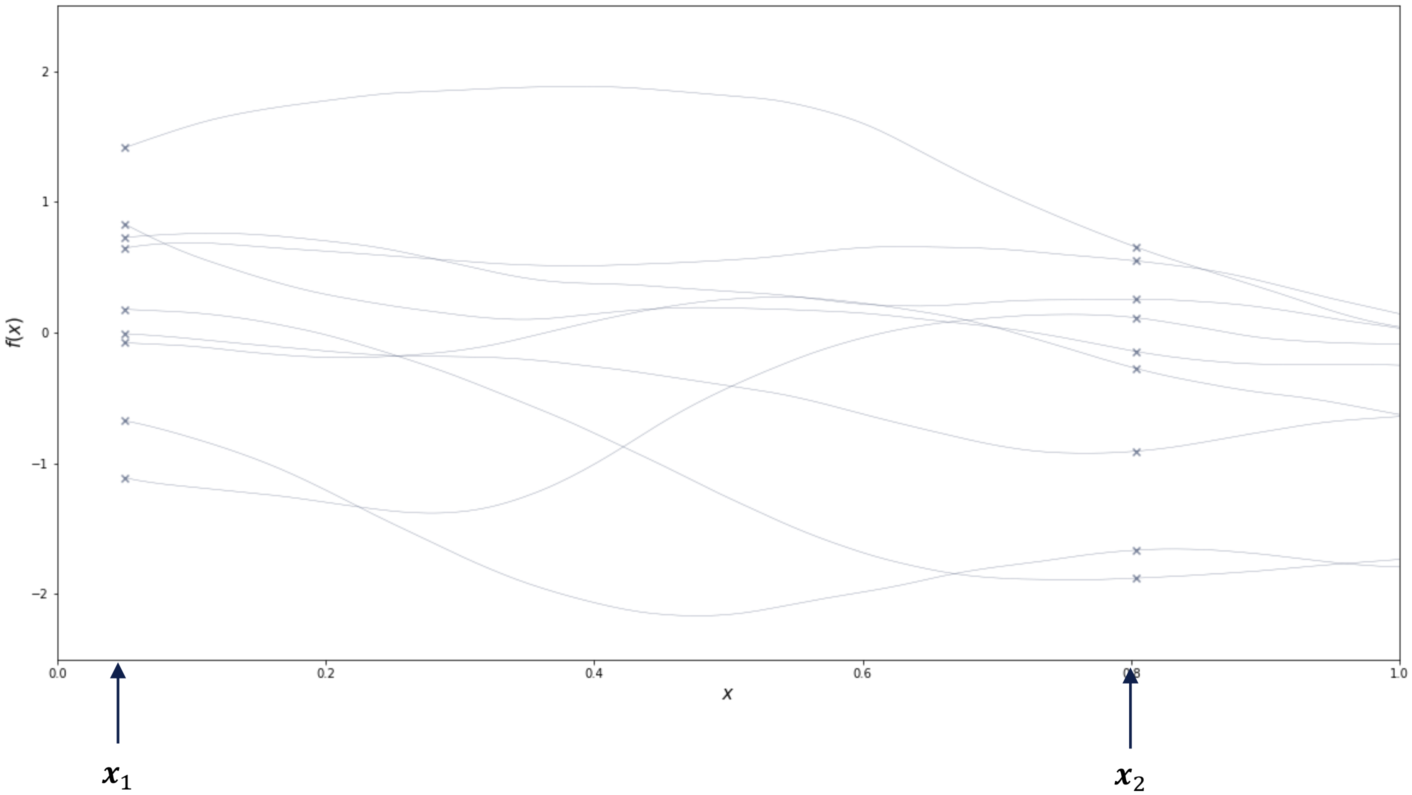




3 Ways to think about GPs
1) A large vector

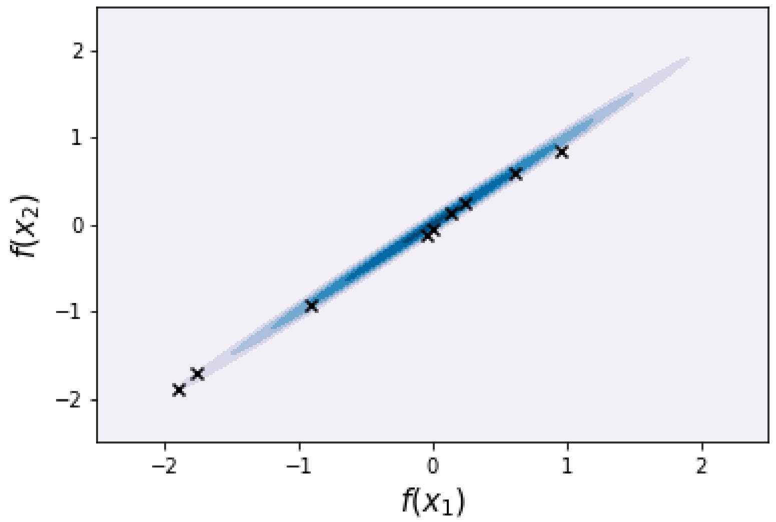
- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_N)]$

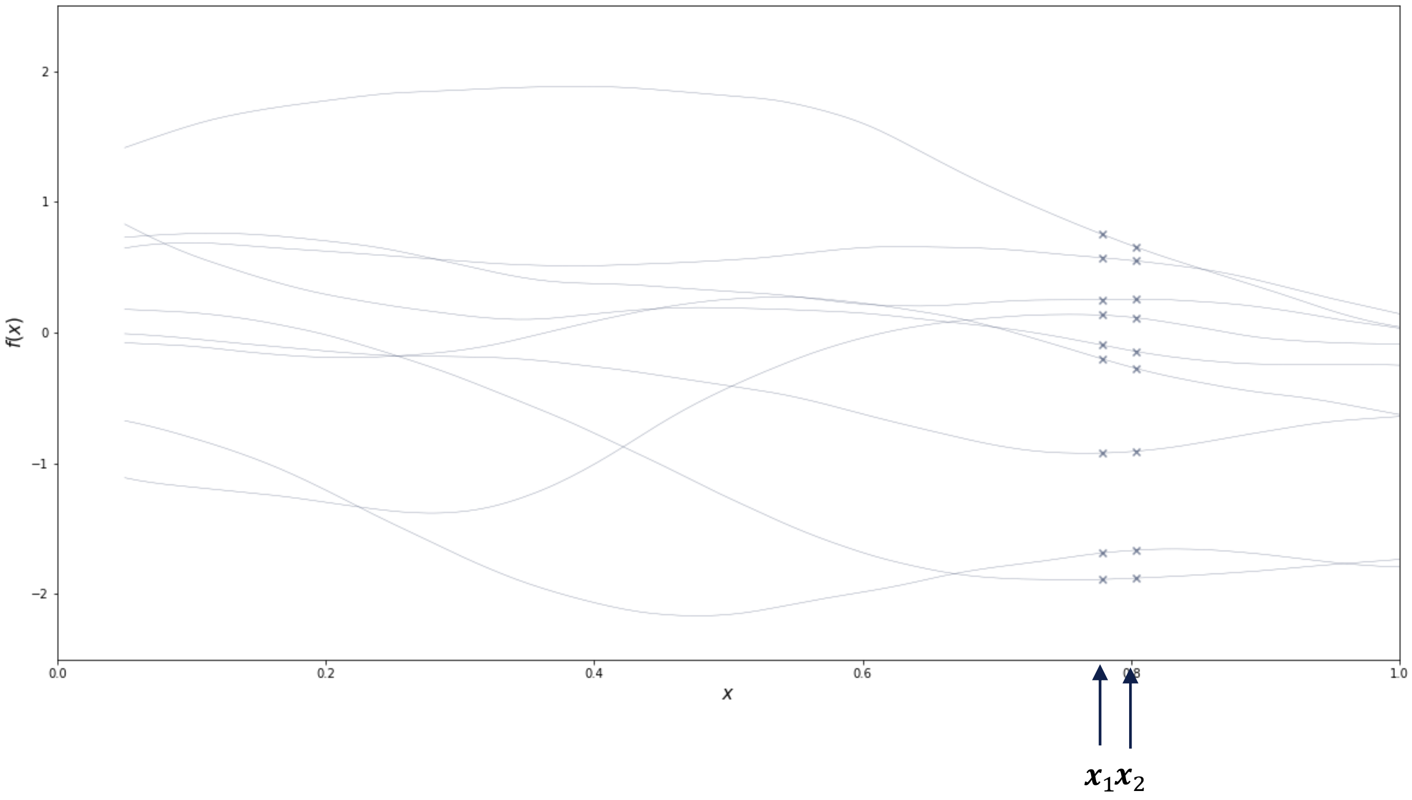




3 Ways to think about GPs
1) A large vector

- Assume $\mathcal{X}$ is finite. $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$
- Then a GP is just a mapping from $\mathcal{X}$ to components of the vector $[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_N)]$





3 Ways to think about GPs
2) A distribution over functions

- The GP is a *function-valued random variable*
- It is a distribution over functions

There is a space of functions associated with the kernel of the GP

This is called the **Reproducing Kernel Hilbert Space**

This connection is very useful for theoretical work!

$$f \sim GP(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$$

Reproducing Kernel Hilbert Space

3 Ways to think about GPs
3) An infinite-parameter model

- Let

$$f(\boldsymbol{x}) = \sum_{i=0}^{N} \lambda_i \xi_i \, \phi_i(\boldsymbol{x}) \qquad (1)$$

where

λ_i = constant

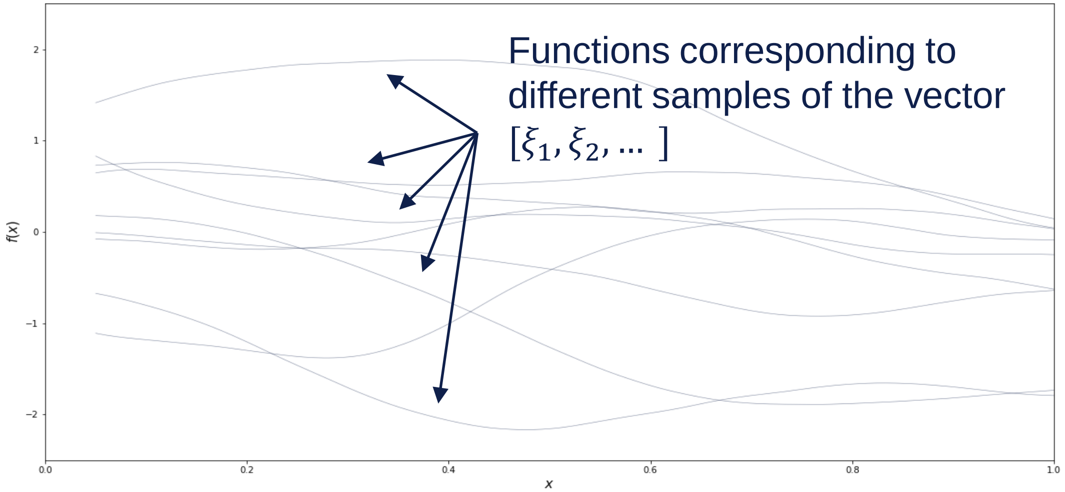
$\phi(\boldsymbol{x})_i$ = deterministic function

ξ_i = Standard normal variable (pairwise independent)

Then f is a GP.

Karhunen-Loève expansion:

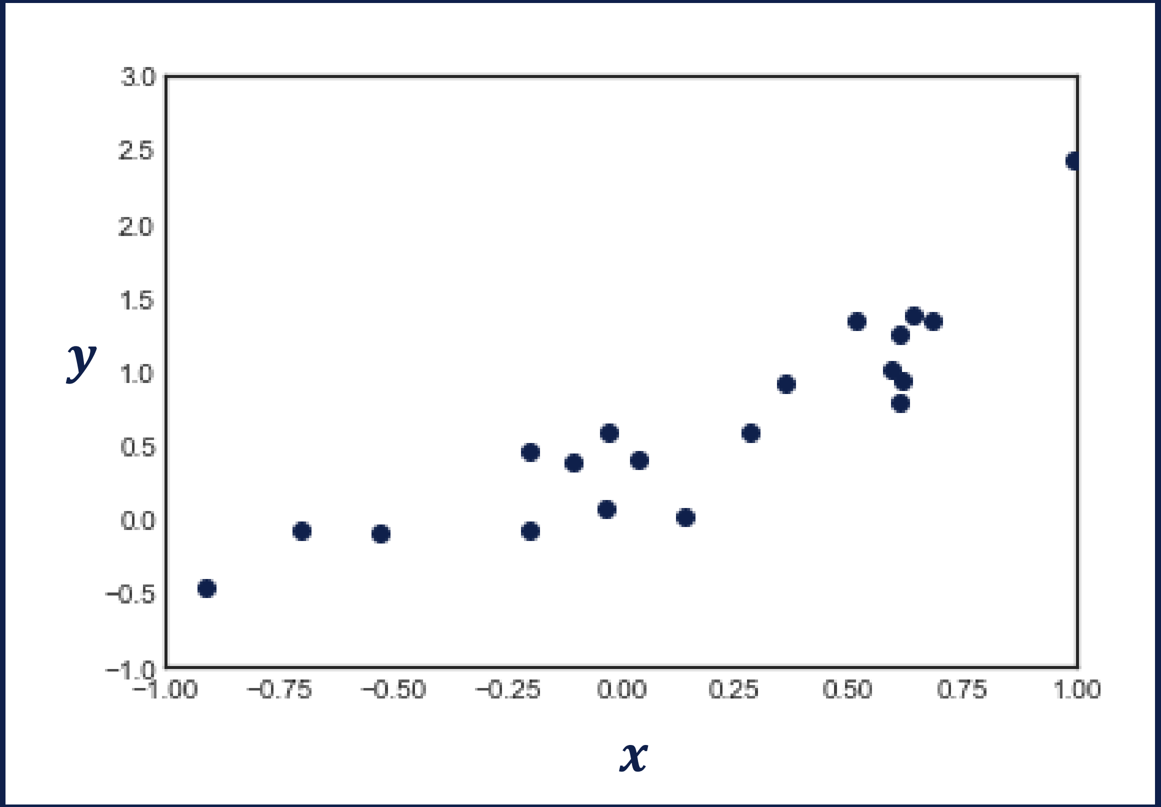
Any GP can be written as (1) with $N = \infty$



Conditioning on data

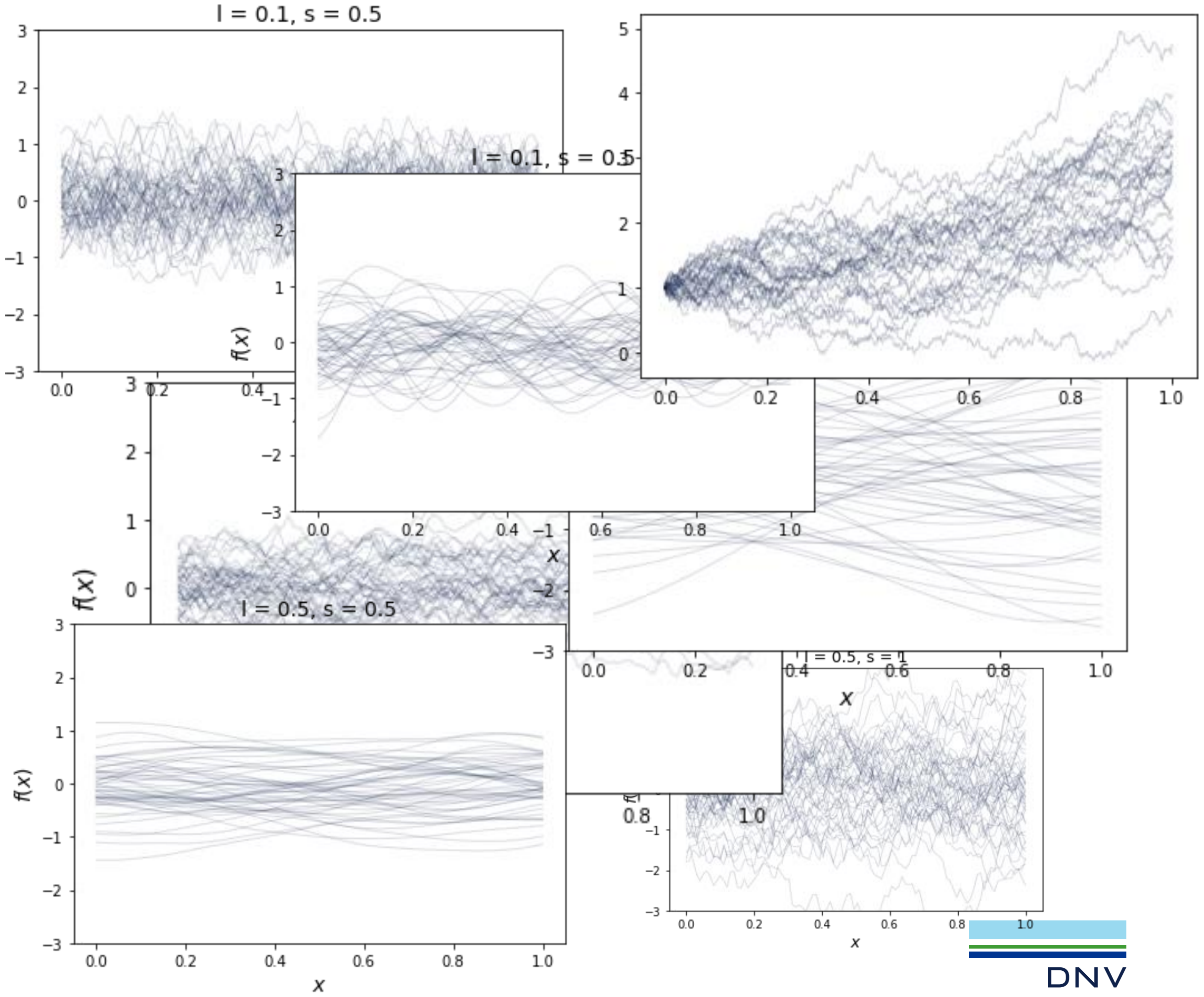
Data

$(x_1, y_1), (x_2, y_2), \ldots$



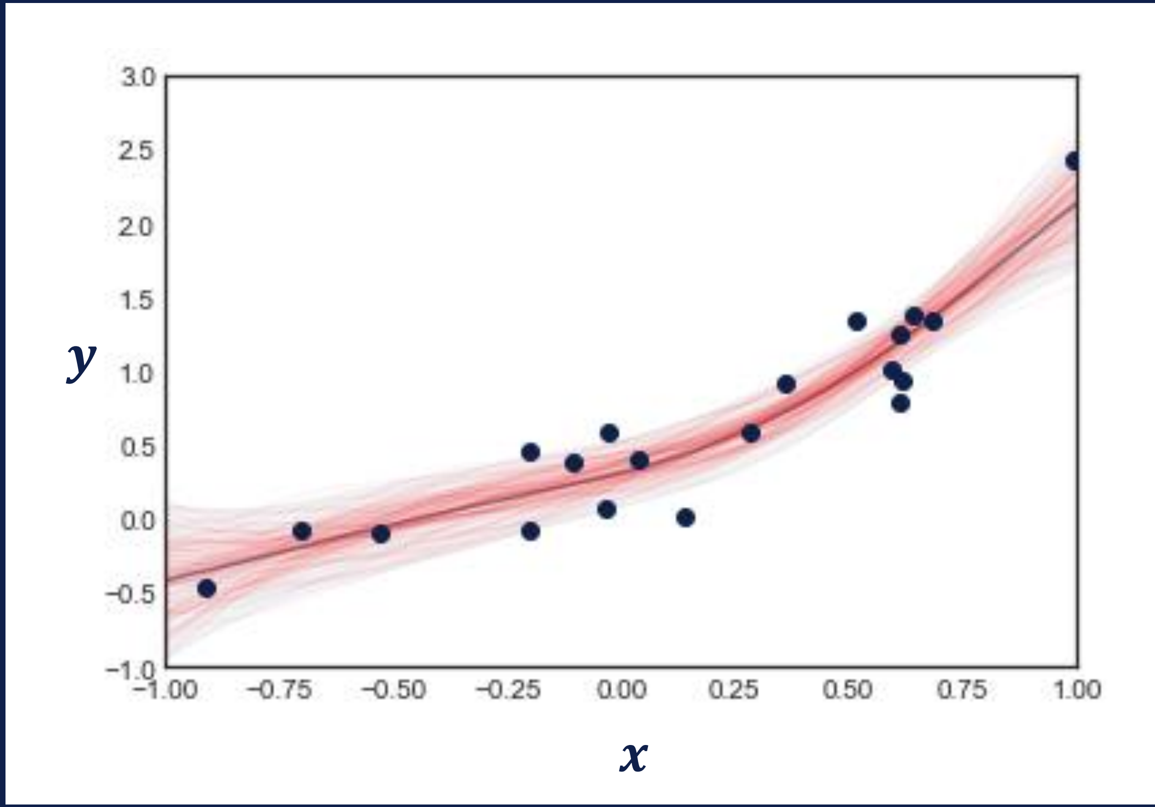
Model

Some different Gaussian process priors



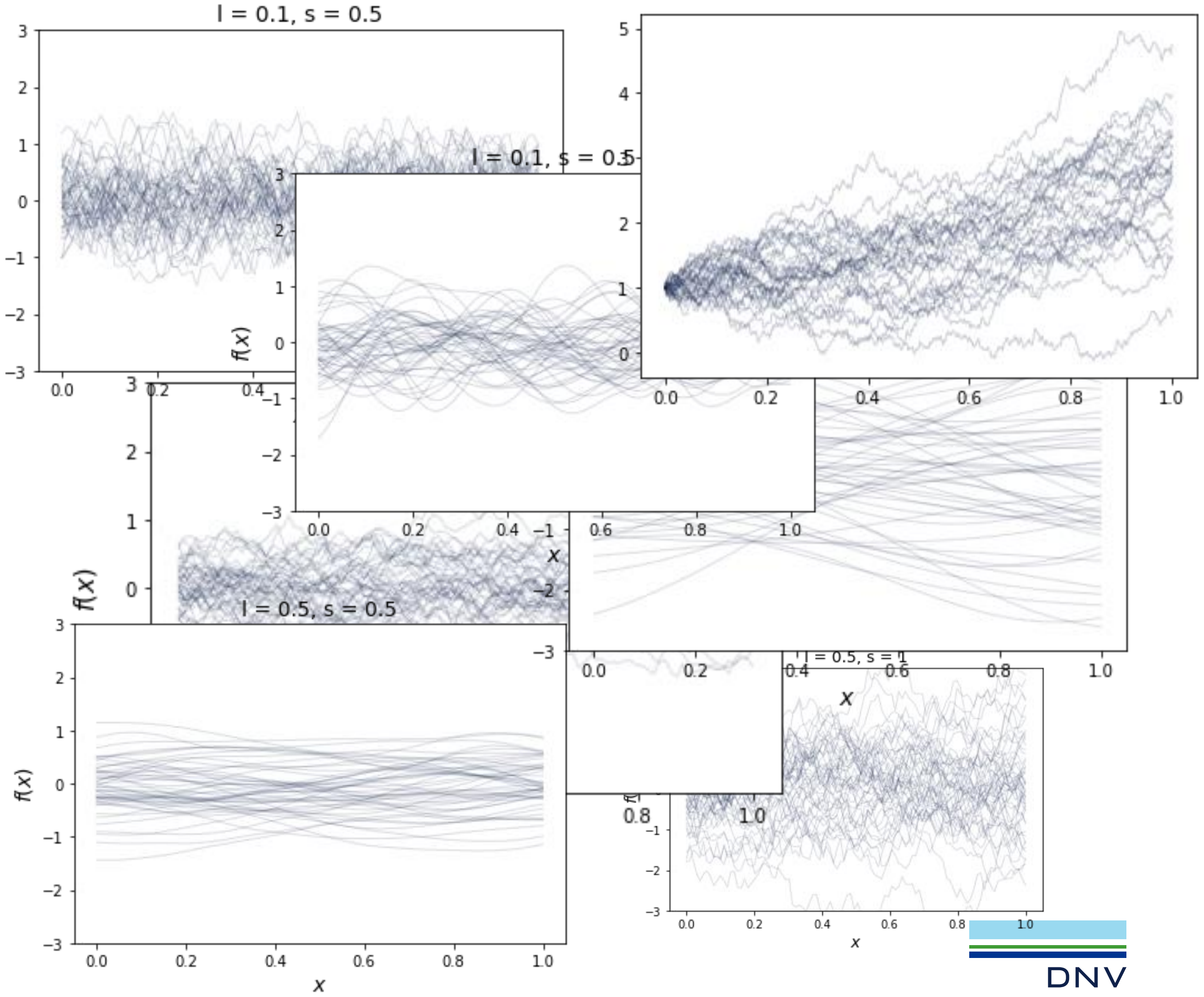
Data

$(x_1, y_1), (x_2, y_2), \ldots$



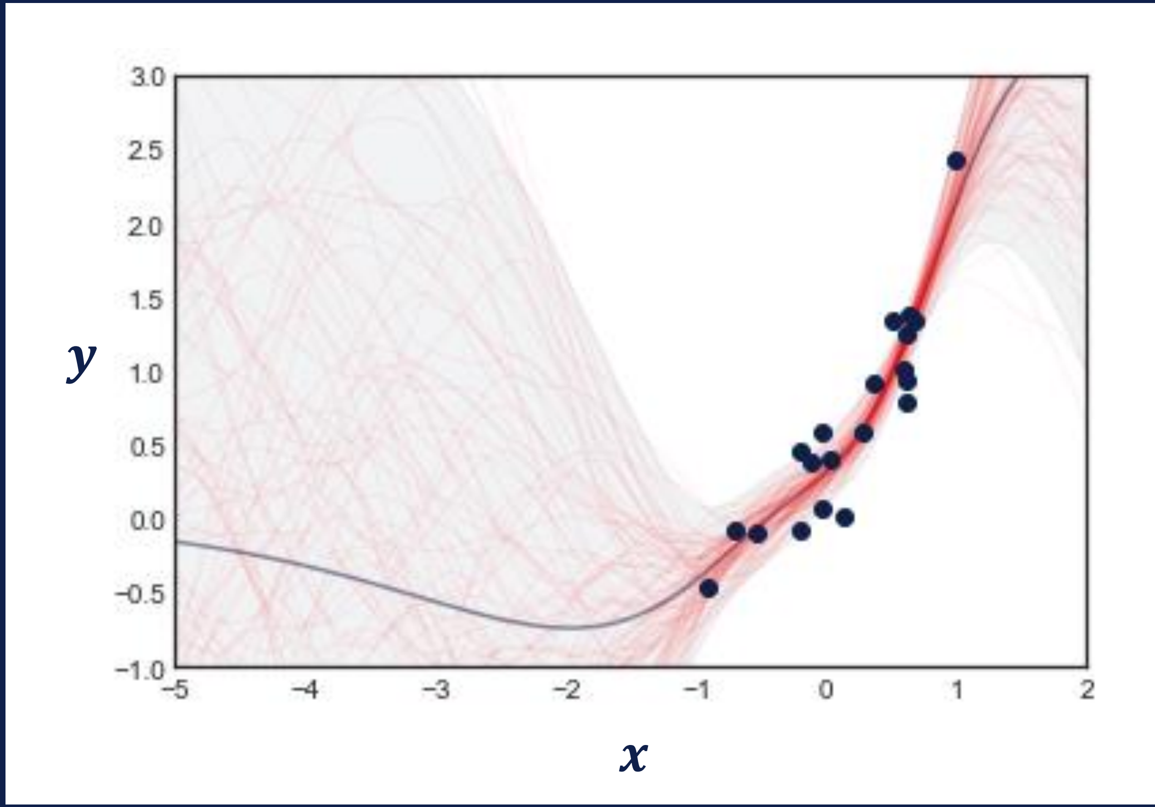
Model

Some different Gaussian process priors



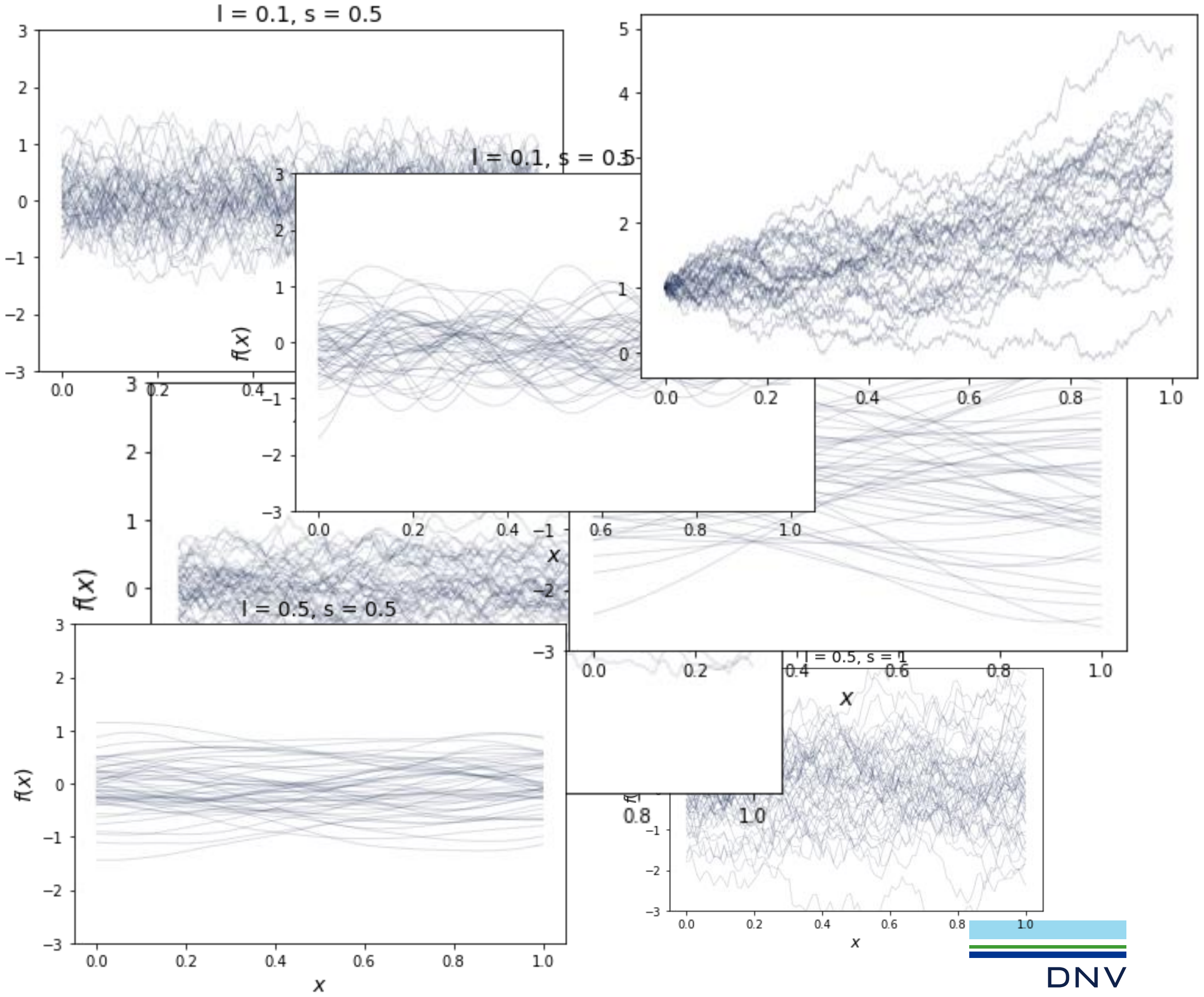
Data

$(x_1, y_1), (x_2, y_2), \ldots$



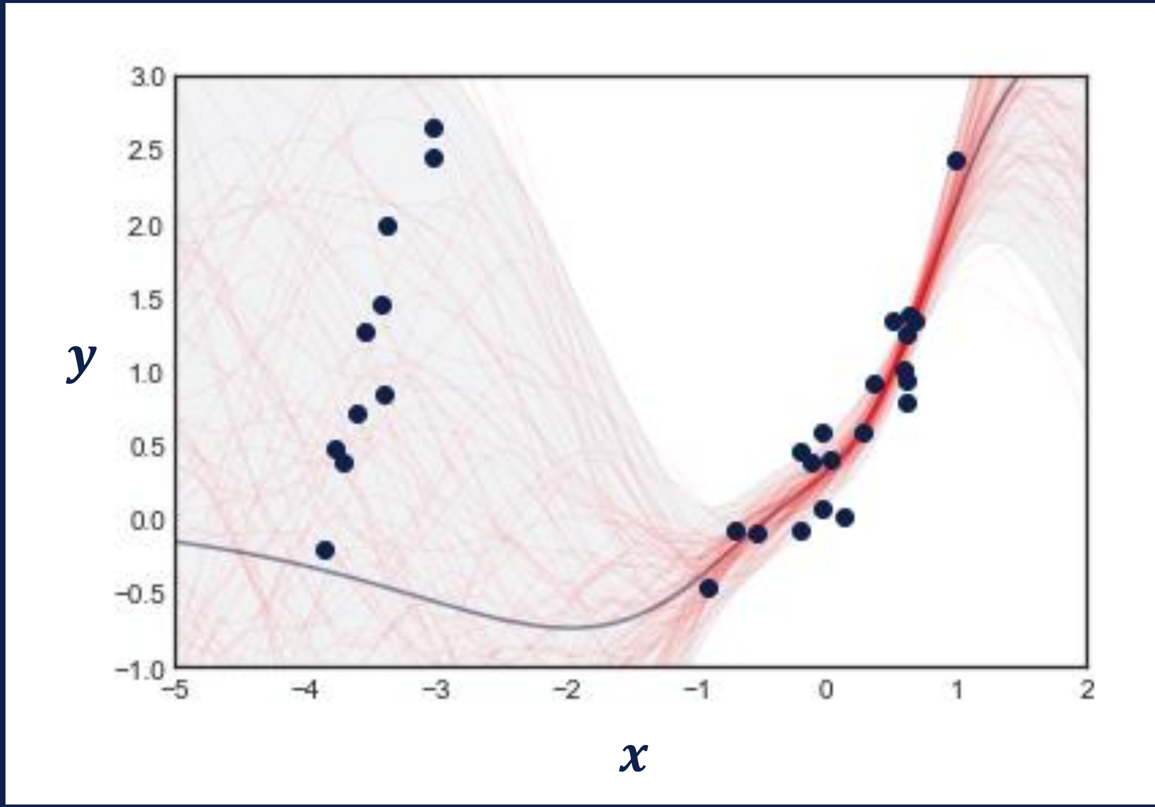
Model

Some different Gaussian process priors



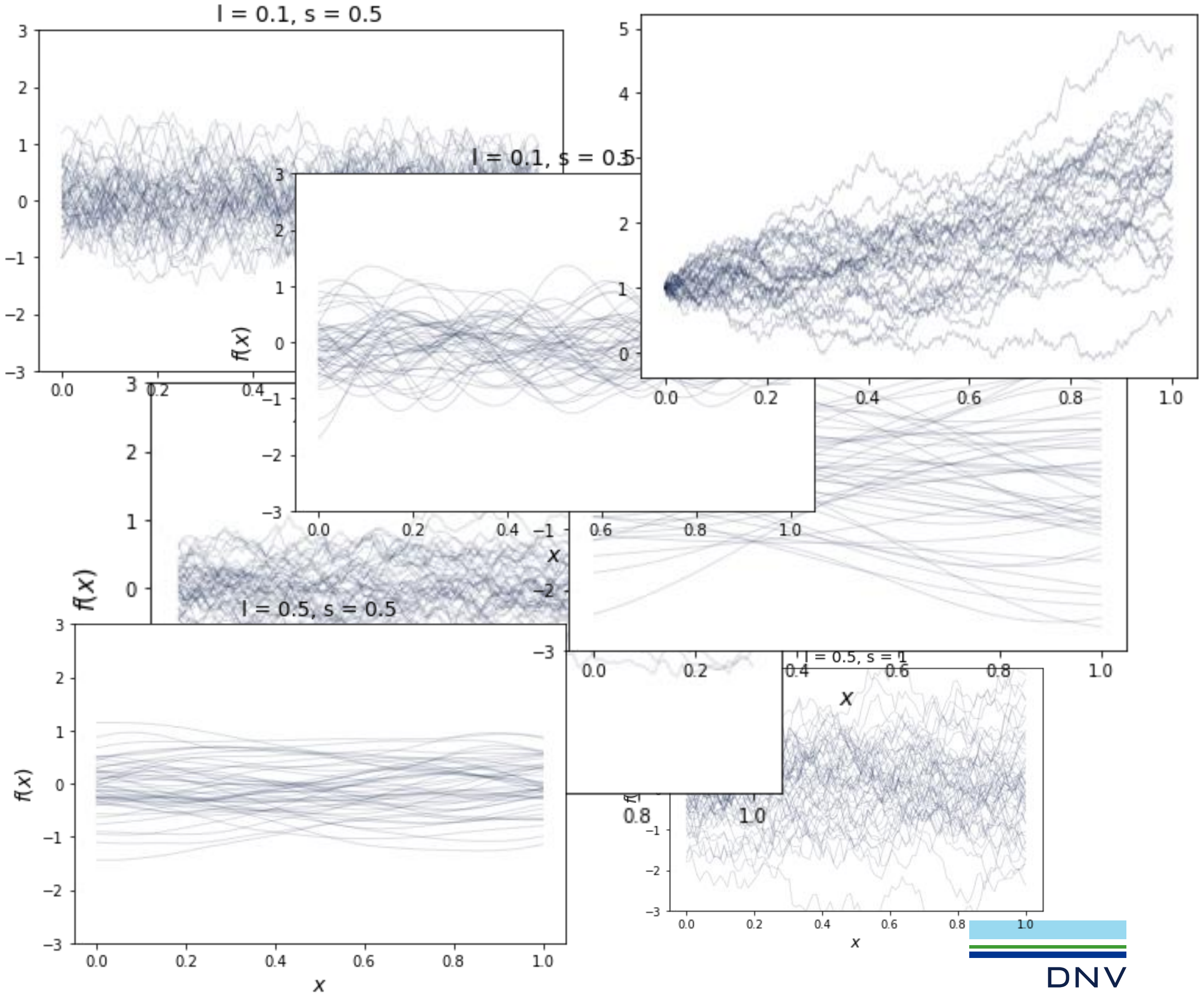
Data

$(x_1, y_1), (x_2, y_2), \ldots$



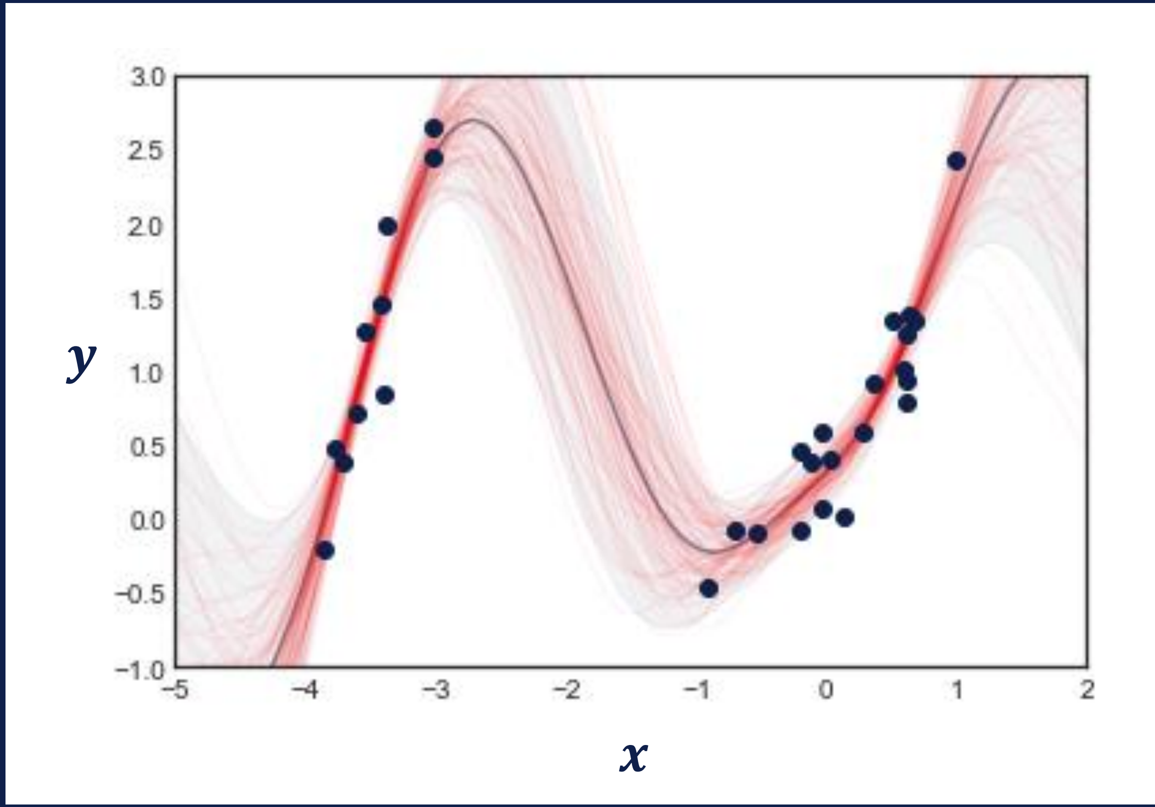
Model

Some different Gaussian process priors



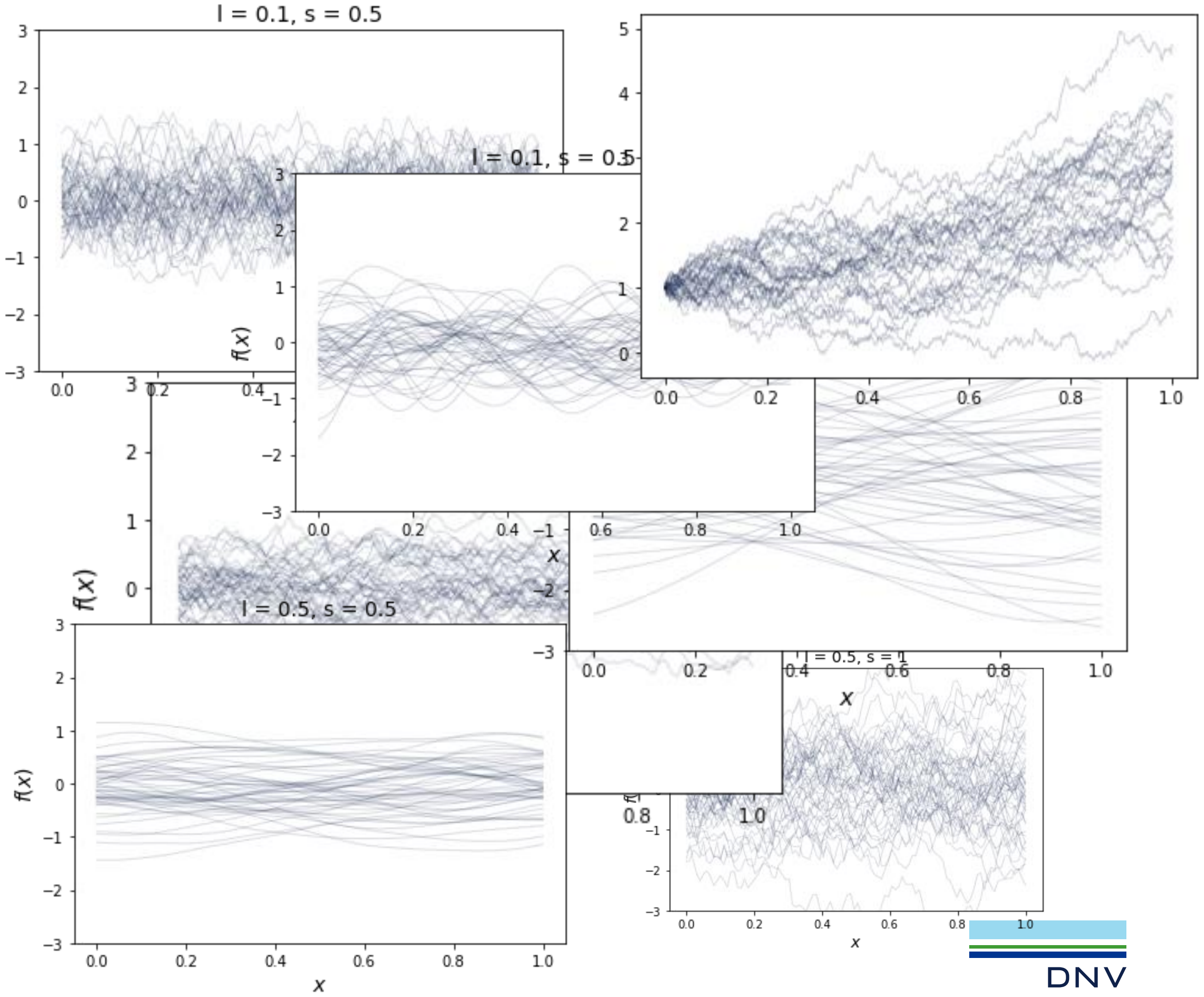
Data

$(x_1, y_1), (x_2, y_2), \ldots$

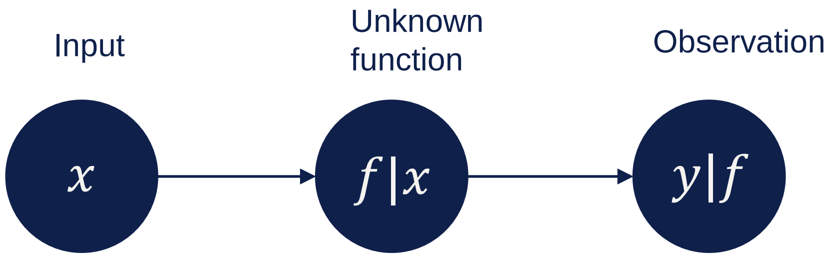


Model

Some different Gaussian process priors



Conditioning on a set of observation



Goal

Infer the function $f(x)$, given a set of observations D=$\{(x_1, y_1), \dots, (x_n, y_n)\}$

Canonical case

Input and output: $x \in \mathbb{R}^d$, $f(x) \in \mathbb{R}$, $y \in \mathbb{R}$
Observations: $y_i = f(x_i) + \varepsilon_i$, ε_i = noise

Standard GP regression

- Assume the noise terms are i.i.d. $\varepsilon_i \sim N(0, \sigma^2)$.
- Let $f \sim GP(\mu, k)$.

For a new set of input locations, $x_1^*, \dots, x_M^*$, let $f^*|D$ denote the posterior process evaluated at each new input, $f^*|D = [f(x_1^*), \dots, f(x_M^*)] | D$.

- Then $f^*|D \sim N(\mu_{f^*|D}, \Sigma_{f^*|D})$ with

$$\mu_{f^*|D} = \mu^* + K^*(K + \sigma^2 I)^{-1}(Y - \mu)$$

$$\Sigma_{f^*|D} = K^{**} - K^*(K + \sigma^2 I)^{-1}(K^*)^T$$

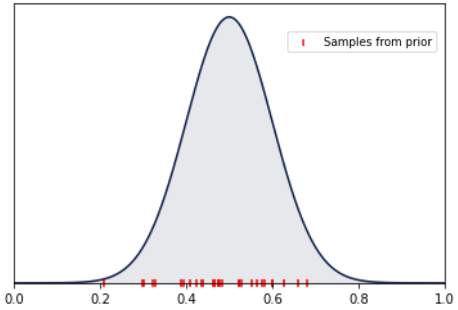
$O(n^3)$ computation
$O(n^2)$ memory
!

where $(\mu)_i = \mu(x_i)$, $(K)_{i,j} = k(x_i, x_j)$, $(K^*)_{i,j} = k(x_i^*, x_j)$ etc.

GP as a prior over functions

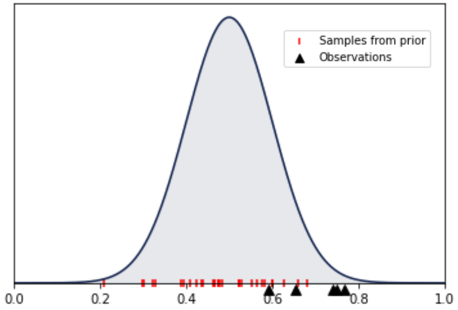
Prior distribution
$p(x|\theta)$

Prior

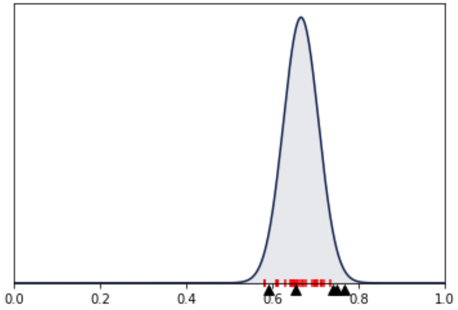


Observations

$y_i = x_{true} + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma^2)$



Posterior

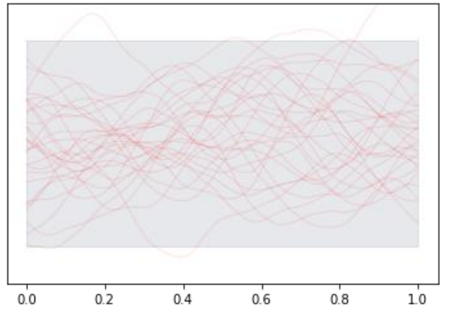


Bayesian inference

(find x_{true})

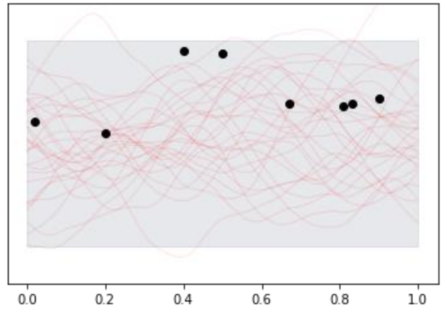
Prior process
$GP(\mu(x|\theta), k(x, x'|\theta))$

Prior

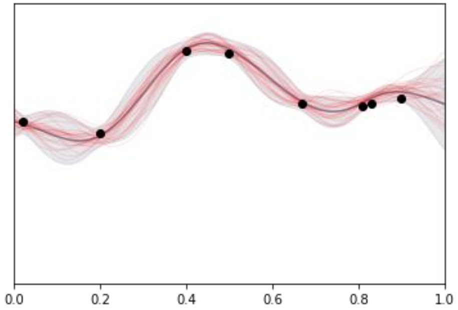


Observations

$y_i = f_{true}(x_i) + \varepsilon_i, \ \varepsilon_i \sim N(0, \sigma^2)$



Posterior



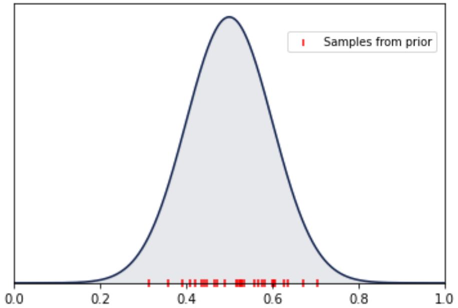
Bayesian inference over functions

(find f_{true})

GP as a prior over functions

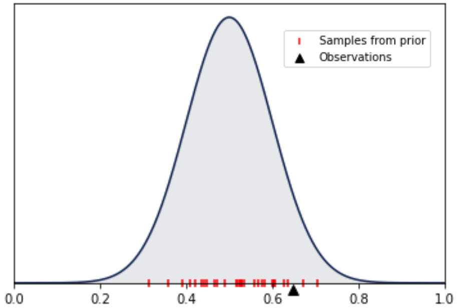
Prior distribution
$p(x|\theta)$

Prior

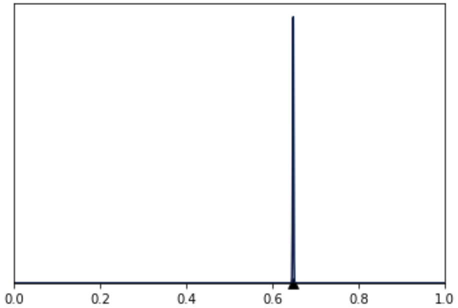


Observations

$y_i = x_{true} + \varepsilon_i, \quad \varepsilon_i = 0$



Posterior

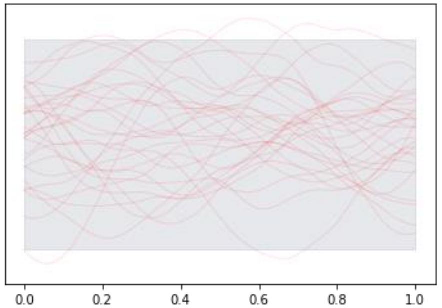


Bayesian inference

(find x_{true})

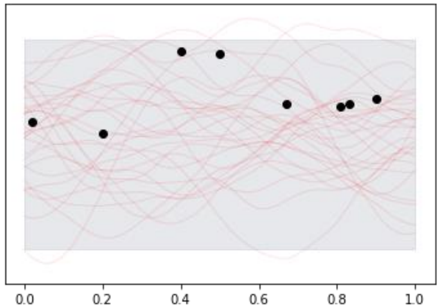
Prior process
$GP(\mu(x|\theta), k(x, x'|\theta))$

Prior

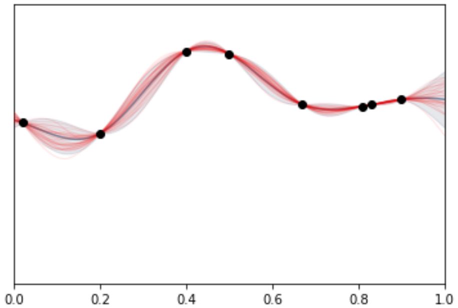


Observations

$y_i = f_{true}(x_i) + \varepsilon_i, \ \varepsilon_i = 0$



Posterior



Bayesian inference over functions

(find f_{true})

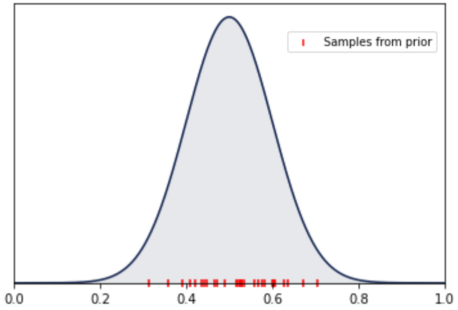
GP as a prior over functions

Prior distribution
$p(x|\boldsymbol{\theta})$

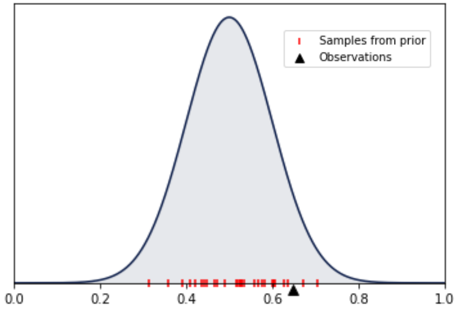
The prior depends on a parameter θ

Prior process
$GP(\mu(x|\boldsymbol{\theta}), k(x, x'|\boldsymbol{\theta}))$

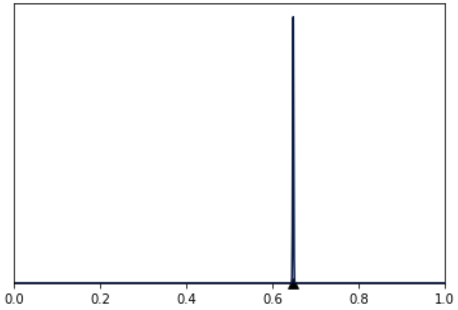
Prior



Observations
$y_i = x_{true} + \varepsilon_i, \quad \varepsilon_i = 0$



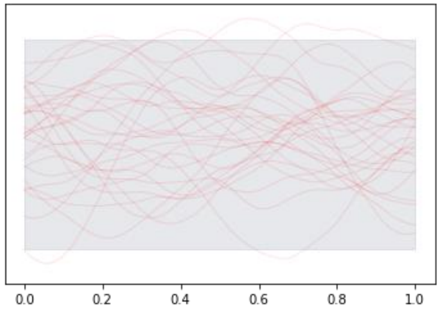
Posterior



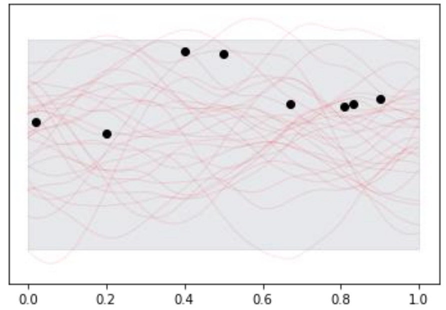
Bayesian inference

(find x_{true})

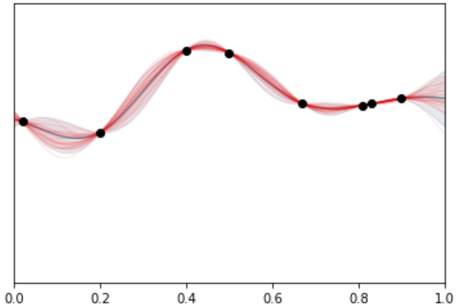
Prior



Observations
$y_i = f_{true}(x_i) + \varepsilon_i, \; \varepsilon_i = 0$



Posterior



Bayesian inference over functions

(find f_{true})

Hyperparameter estimation

The covariance function

Recall: $f \sim GP(\mu, k)$

- Assume $\mu = 0$
  (Or that we work with $f - \mu$. This is not a restrictive assumption)

- Assume k is stationary: $k(x_1, x_2)$ can be written as $k(x_1 - x_2)$
  (Needed for theoretical analysis. This is often used in practice)

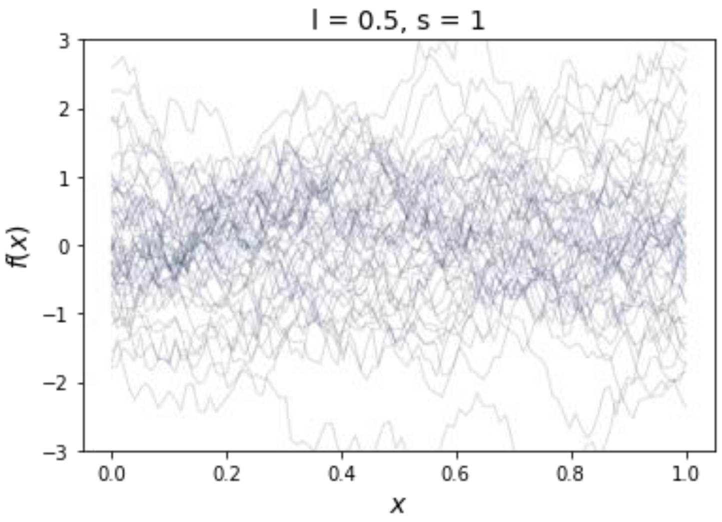
How do we choose an appropriate covariance function k ?

- Let $\{k_\theta \mid \theta \in \Theta\}$ be a set of covariance functions parameterised by θ

- Below are some examples with $\theta = (s, l)$ and $r = \|x_1 - x_2\| \,/\, l$

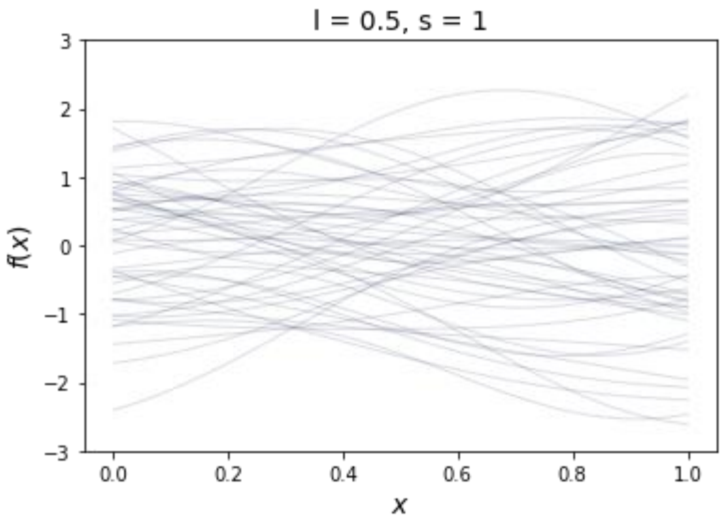
Exponential

$$k_\theta(x_1, x_2) = s^2 e^{-r}$$



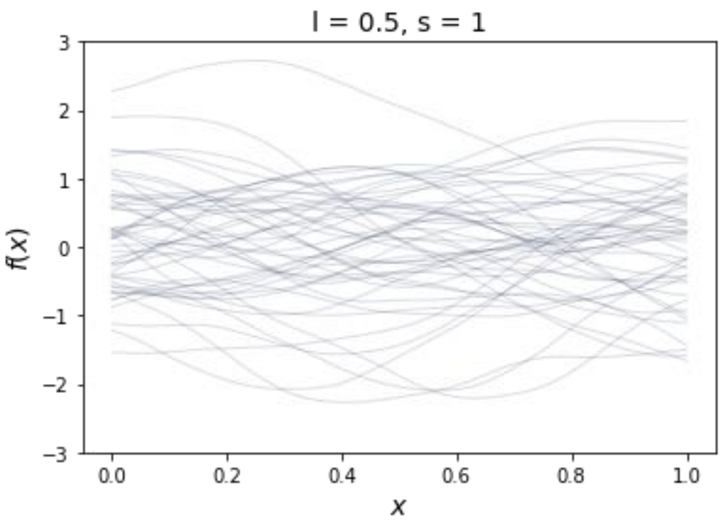
Gaussian

$$k_\theta(x_1, x_2) = s^2 e^{-\frac{1}{2}r^2}$$



Matérn 5/2

$$k_\theta(x_1, x_2) = s^2 \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right) e^{-\sqrt{5}r}$$



The covariance function

Recall: $f \sim GP(\mu, k)$

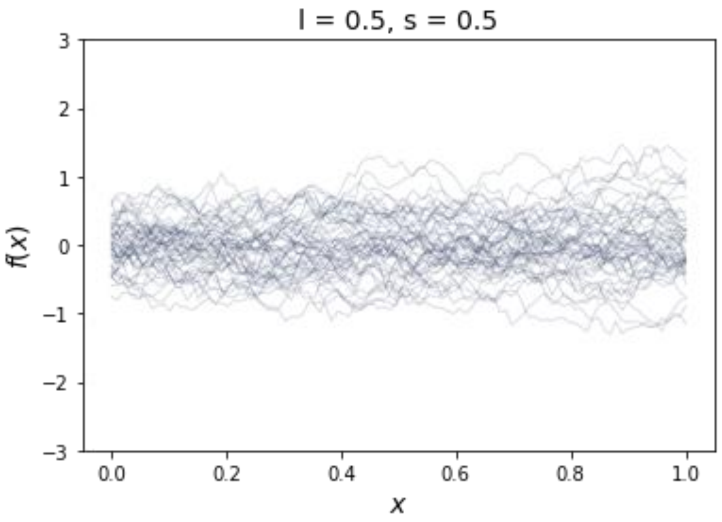
- Assume $\mu = 0$
  (Or that we work with $f - \mu$. This is not a restrictive assumption)
- Assume k is stationary: $k(x_1, x_2)$ can be written as $k(x_1 - x_2)$
  (Needed for theoretical analysis. This is often used in practice)

How do we choose an appropriate covariance function $\boldsymbol{k}$?

- Let $\{k_\theta \mid \theta \in \Theta\}$ be a set of covariance functions parameterised by θ
- Below are some examples with $\theta = (s, l)$ and $r = \|x_1 - x_2\| \,/\, l$

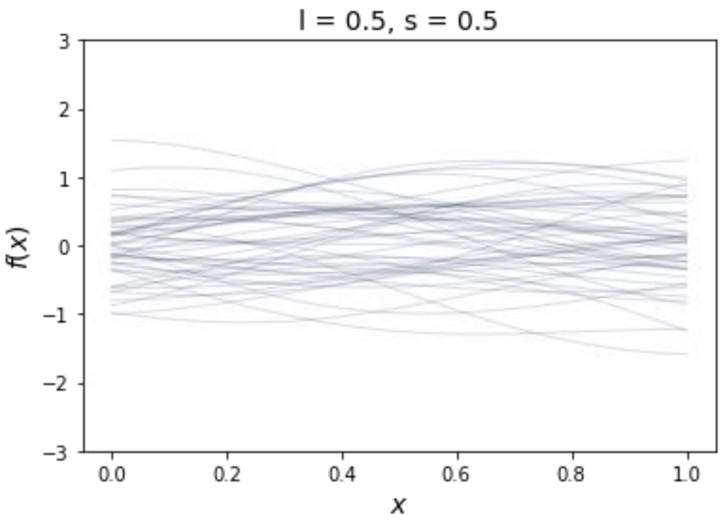
Exponential

$$k_\theta(x_1, x_2) = s^2 e^{-r}$$



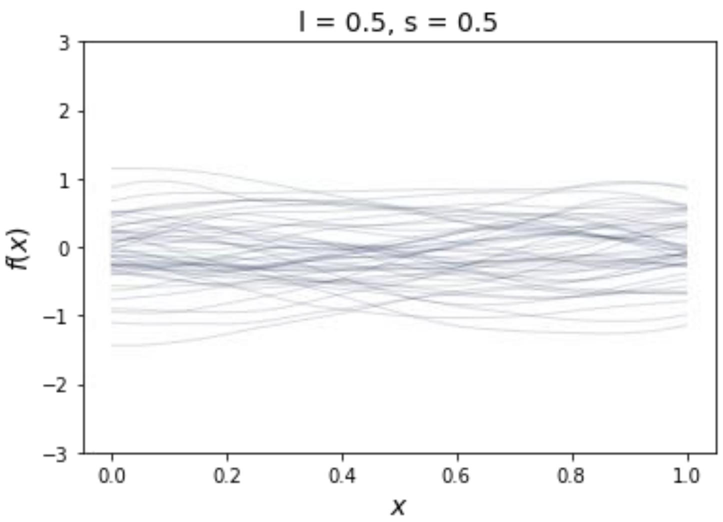
Gaussian

$$k_\theta(x_1, x_2) = s^2 e^{-\frac{1}{2}r^2}$$



Matérn 5/2

$$k_\theta(x_1, x_2) = s^2 \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right) e^{-\sqrt{5}r}$$



The covariance function

Recall: $f \sim GP(\mu, k)$

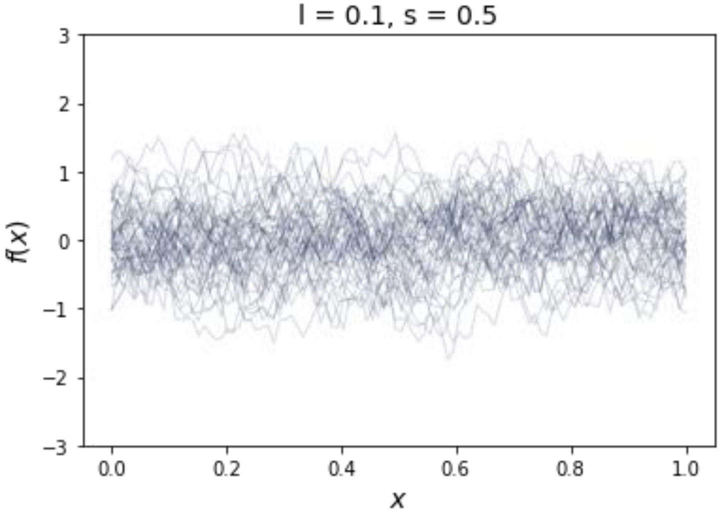
- Assume $\mu = 0$
  (Or that we work with $f - \mu$. This is not a restrictive assumption)
- Assume k is stationary: $k(x_1, x_2)$ can be written as $k(x_1 - x_2)$
  (Needed for theoretical analysis. This is often used in practice)

How do we choose an appropriate covariance function k ?

- Let $\{k_\theta \mid \theta \in \Theta\}$ be a set of covariance functions parameterised by θ
- Below are some examples with $\theta = (s, l)$ and $r = \|x_1 - x_2\| \,/\, l$

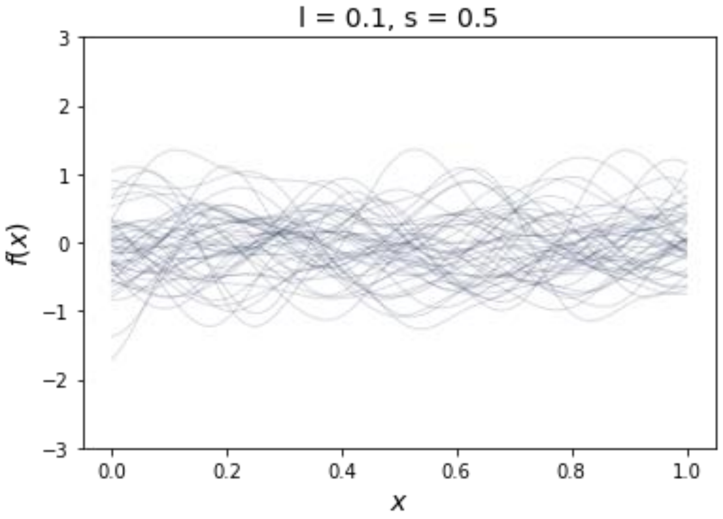
Exponential

$$k_\theta(x_1, x_2) = s^2 e^{-r}$$



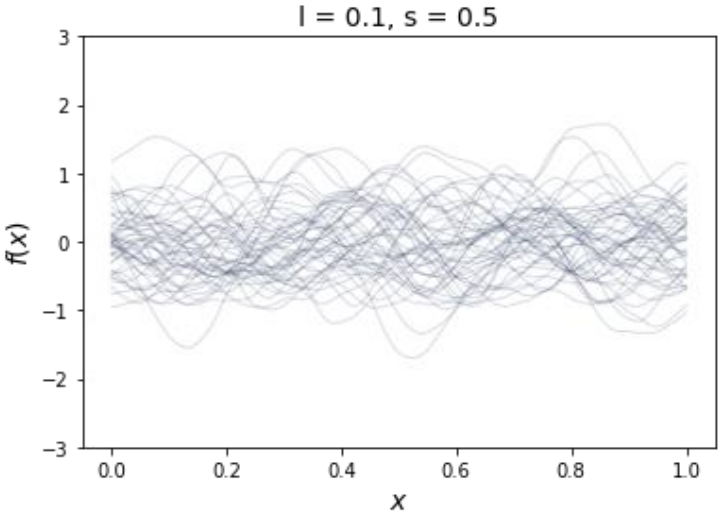
Gaussian

$$k_\theta(x_1, x_2) = s^2 e^{-\frac{1}{2}r^2}$$



Matérn 5/2

$$k_\theta(x_1, x_2) = s^2 \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right) e^{-\sqrt{5}r}$$



Tree ways of estimating θ

1. Maximum likelihood (ML)
   - Most common alternative
2. Cross validation (CV)
   - Leave-one-out predictions can be made efficient
3. Bayesian
   - MAP estimates
   - Full Bayesian treatment with MCMC to sample from $p(\theta|D)$[1]
   - Some use within Uncertainty Quantification[2]

Most common to use one of these and the plug-in approach

The plug-in approach

(Also called Type-II maximum likelihood)

- Compute a fixed estimate $\hat{\theta}$
- Treat $\hat{\theta}$ as the “true” value and compute the posterior GP for $k_{\hat{\theta}}$

Maximum likelihood (ML)

We have

- GP prior: $f \sim GP(0, k)$
- Data $\{(x_i, y_i)\}_{i=1}^{n}$ where: $y_i = f(x_i)$
- This means that

$$Y \sim N(0, \mathrm{K})$$

with $\mathrm{K}_{i,j} = k\left(x_i, x_j\right)$, $Y_i = y_i$

The log likelihood:

$$L(\theta) = \underbrace{-\frac{1}{2} Y^T \mathrm{K}_\theta^{-1} Y}_{\text{Data fit}} \; - \; \underbrace{\frac{1}{2} \log|\mathrm{K}_\theta|}_{\text{Model complexity}} \; - \; \frac{n}{2} \log 2\pi$$

The covariance matrix that depends on θ

- $\mathrm{K} = \mathrm{K}_\theta$ depends on some parameter θ

Remark

If $y_i = f(x_i) + \varepsilon_i$, $\varepsilon_i \sim N(0, \sigma^2)$, we would make us of the covariance matrix $\Sigma_\theta = (K_\theta + \sigma^2 I)$

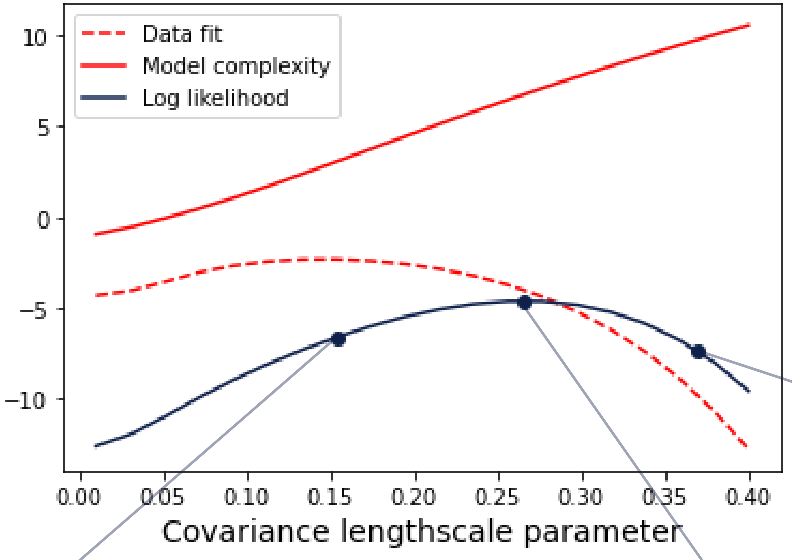
- The noise variance σ^2 could also be estimated together with θ
- We call $L(\theta)$ the *log marginal likelihood*

Recall the Gaussian density

$$p(Y|X, \theta) = \frac{1}{(2\pi)^{n/2} \sqrt{|\mathrm{K}_\theta|}} e^{-\frac{1}{2} Y^T \mathrm{K}_\theta^{-1} Y}$$

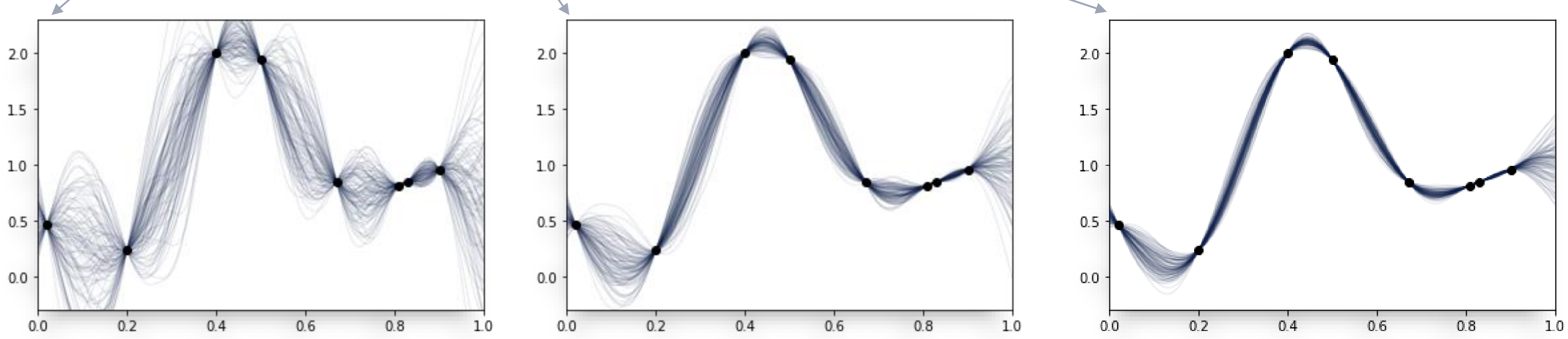
(3) C. E. Rasmussen and C. K. I. Williams. Gaussian Processes for Machine Learning. MIT Press, 2006.

Maximum likelihood (ML)



The log likelihood:

$$L(\theta) = \underbrace{-\frac{1}{2} Y^T \mathrm{K}_\theta^{-1} Y}_{\text{Data fit}} \; - \; \underbrace{\frac{1}{2} \log|\mathrm{K}_\theta|}_{\text{Model complexity}} \; - \; \frac{n}{2} \log 2\pi$$



Putting it all together

- Gaussian process regression

Gaussian process regression

Prior

1) Select model
2) Select likelihood
3) Optimize hyperparameters

Posterior

4) Condition on data
5) Make predictions

Gaussian process regression

Prior

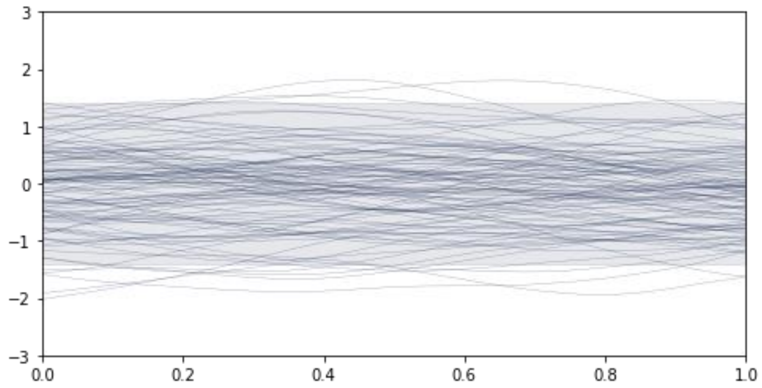
1) Select model
2) Select likelihood
3) Optimize hyperparameters

Posterior

4) Condition on data
5) Make predictions

Select mean $\mu(\boldsymbol{x})$ and covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$

Tip: Start with $\mu = 0$ and k = Matérn



Gaussian process regression

Prior

1) Select model
2) Select likelihood
3) Optimize hyperparameters

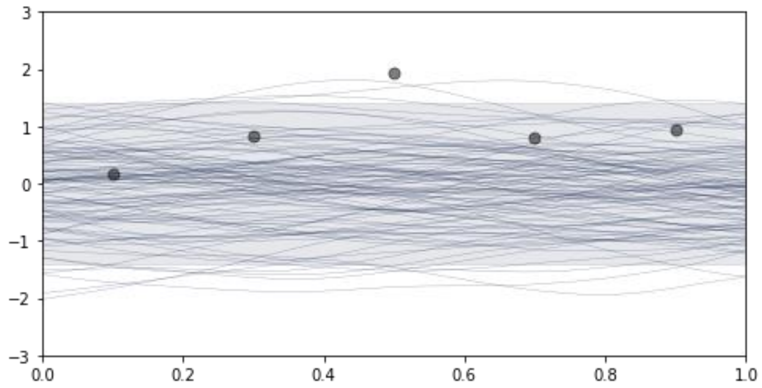
Posterior

4) Condition on data
5) Make predictions

Assume additive Gaussian noise: $y = f(\boldsymbol{x}) + \varepsilon$, $\varepsilon \sim N(0, \sigma^2)$.

Decide if σ^2 is fixed or unknown.

Tip: Set $\sigma^2 = 10^{-6} \approx 0$ *for noiseless data.*



Gaussian process regression

Prior

1) Select model
2) Select likelihood
3) Optimize hyperparameters

Posterior

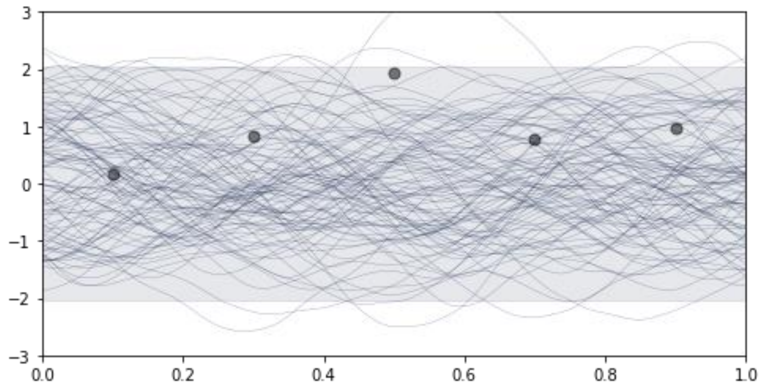
4) Condition on data
5) Make predictions

Identify which parameters of the mean function, covariance function, and likelihood to optimize:

$$\mu(\boldsymbol{x}|\beta),\ k(\boldsymbol{x},\boldsymbol{x}'|\theta),\ \varepsilon \sim N(0,\sigma^2).$$

Optimize using e.g. maximum likelihood

$$(\beta,\theta,\sigma) \in argmax\ L(\beta,\theta,\sigma)$$



Gaussian process regression

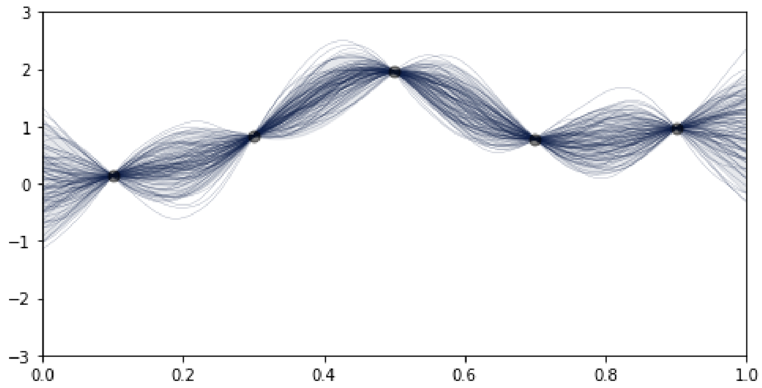
Prior
1) Select model
2) Select likelihood
3) Optimize hyperparameters

Posterior
4) Condition on data
5) Make predictions

The posterior GP can be computed analytically

$$\mu_{f^*|D} = \mu^* + K^*(K + \sigma^2 I)^{-1}(Y - \mu)$$

$$\Sigma_{f^*|D} = K^{**} - K^*(K + \sigma^2 I)^{-1}(K^*)^T$$



Gaussian process regression

Prior

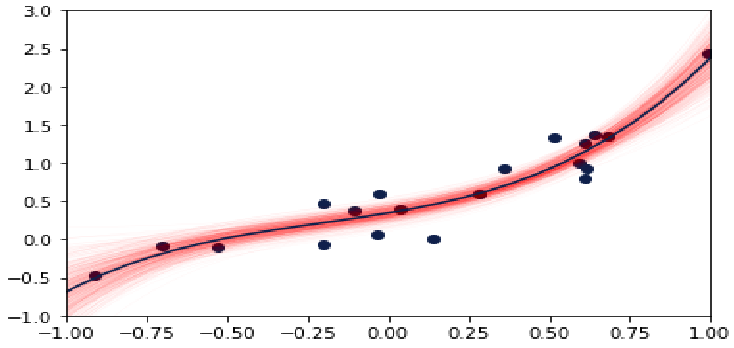
1) Select model
2) Select likelihood
3) Optimize hyperparameters

Posterior

4) Condition on data
5) Make predictions

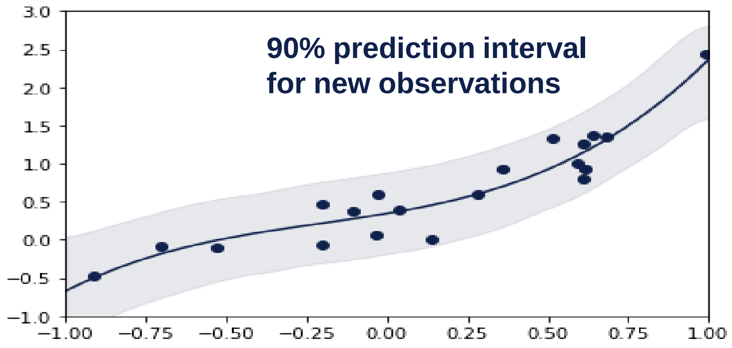
Predict the latent function

$$f(\boldsymbol{x})$$



Predict the next observation

$$f(\boldsymbol{x}) + \varepsilon$$



Gaussian process regression

Generalisations:

We have focused on functions $f: \mathbb{R} \to \mathbb{R}$ with i.i.d. noise.

- Extending to $f: \mathbb{R}^N \to \mathbb{R}^M$ is trivial
- Extending to general Gaussian noise is trivial

Limitations:

- Non-Gaussian noise
- Large datasets (due to cubic time complexity)

For this, special methods are needed.

Software

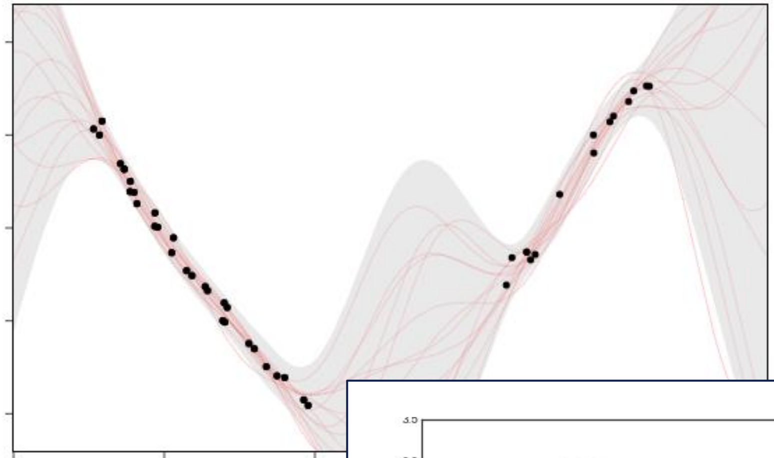
There are many ML and UQ software packages for GPs

Some Python alternatives:

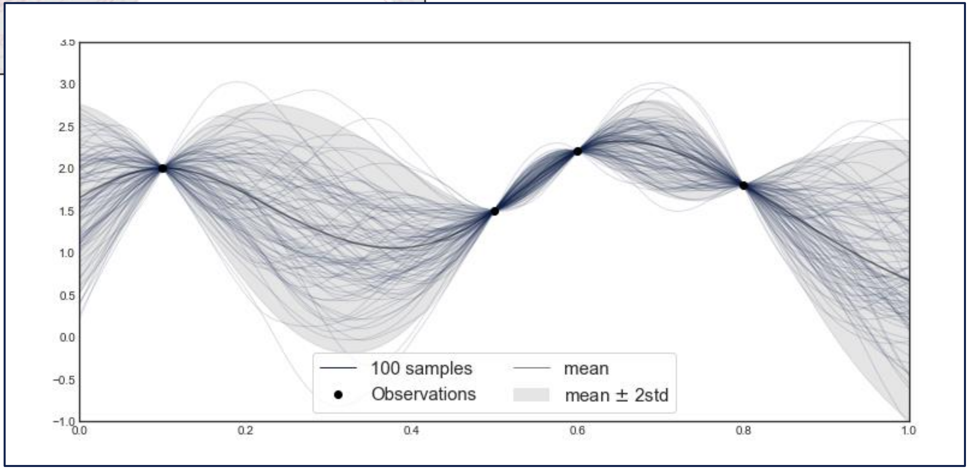
- scikit-learn
- GPy
- GPyTorch
- GPflow

DOE

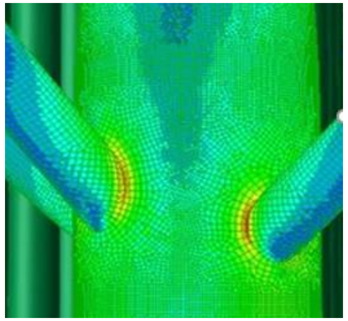
Design of experiments



Active learning with Gaussian processes for expensive data

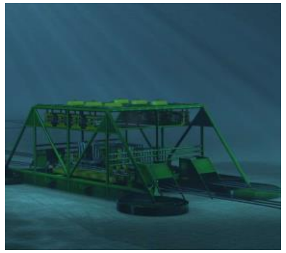


Computer experiment

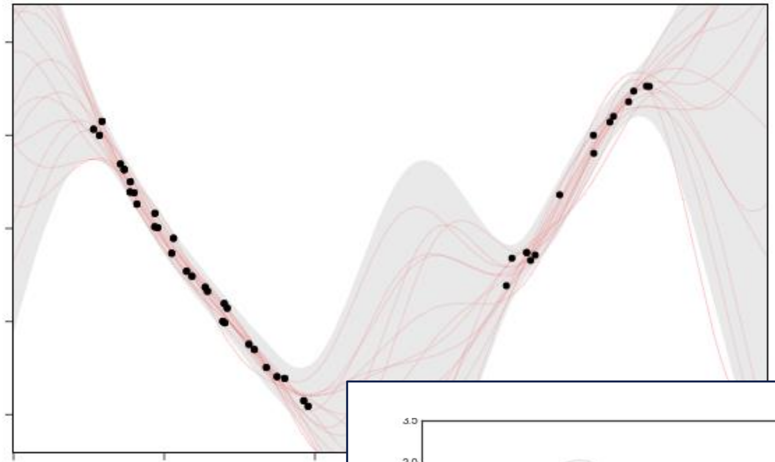


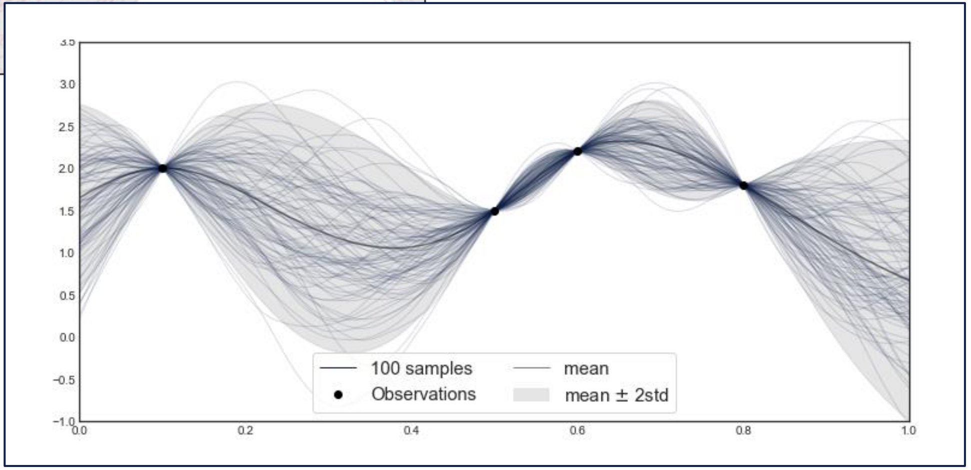
Lab experiment

Inspection



Design of experiments





Active learning with Gaussian processes for expensive data

To find the maximum of the function that has generated the data, we can use e.g.

Upper confidence band

$$x \in \operatorname{argmax}(E[f(x)] + \lambda \cdot Std[f(x)])$$

Expected improvement

$$x \in \operatorname{argmax} E[\max(f(x) - f(x^+), 0]$$

Where $f(x)$ is the GP (or any other object with epistemic uncertainty that depends on x)

Bayesian optimization

Expected improvement

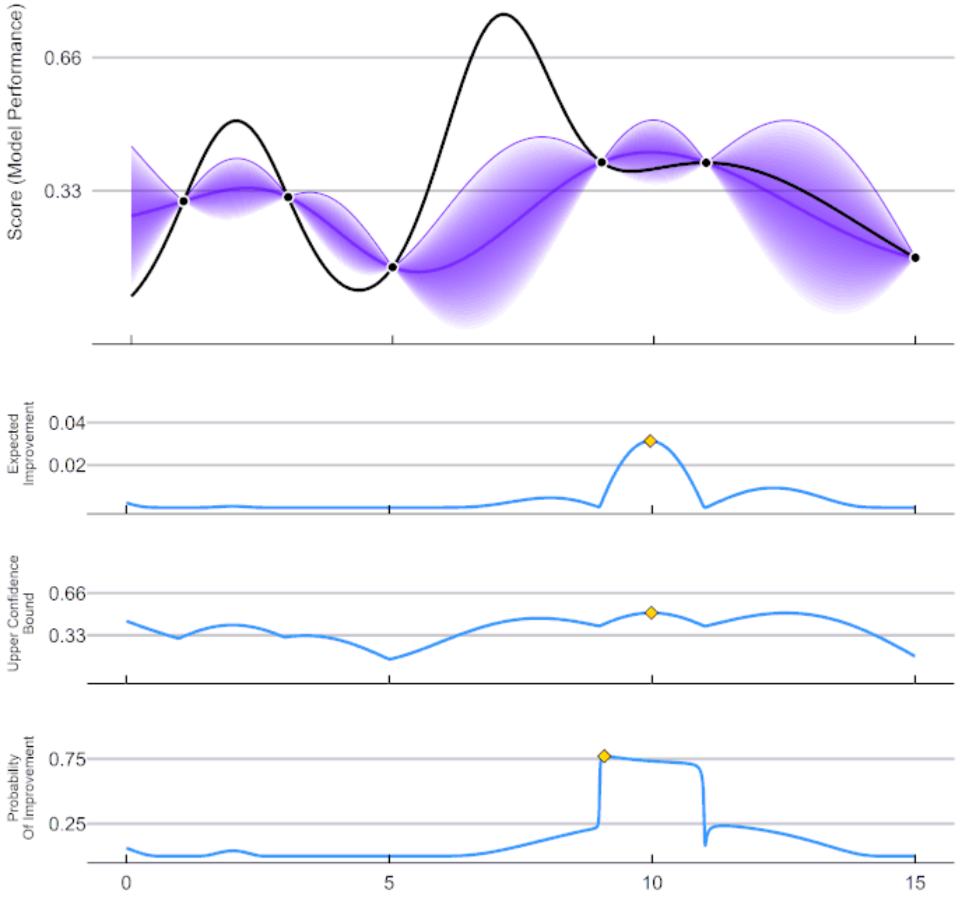
$$E[\max(f(x) - f(x^+), 0]$$

Upper confidence band

$$E[f(x)] + \lambda \cdot Std[f(x)]$$

Probability of improvement

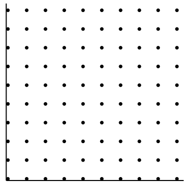
$$P(f(x) > f(x^+))$$

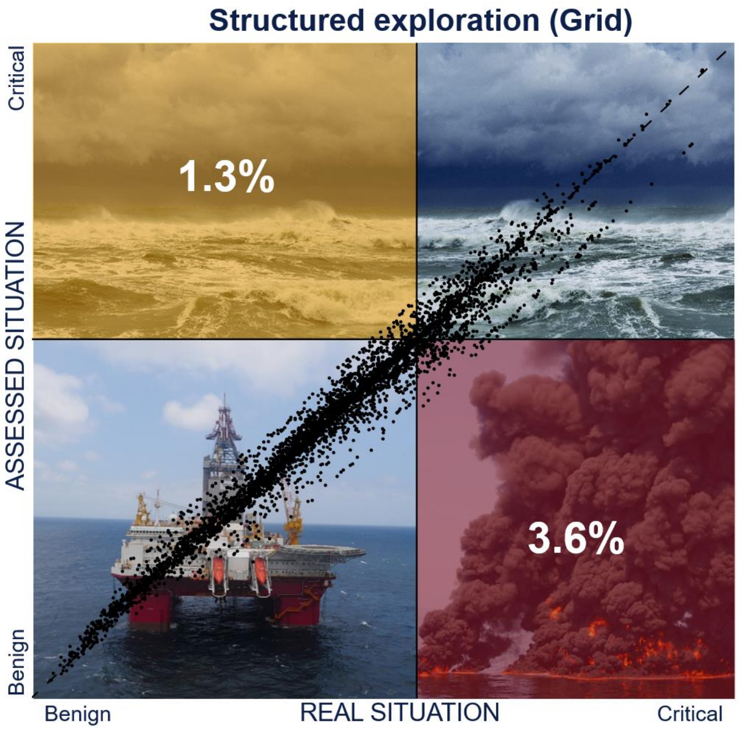


Example - Risk-based design of experiments

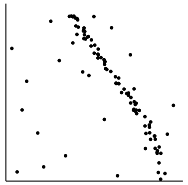
Eldevik, S. and Sætre, S. (2020) *Offshore Workover Operations: Reducing Uncertainty of Critical Weather Scenarios by Optimal Use of Simulations and Probabilistic Machine Learning.* ESREL 2020.

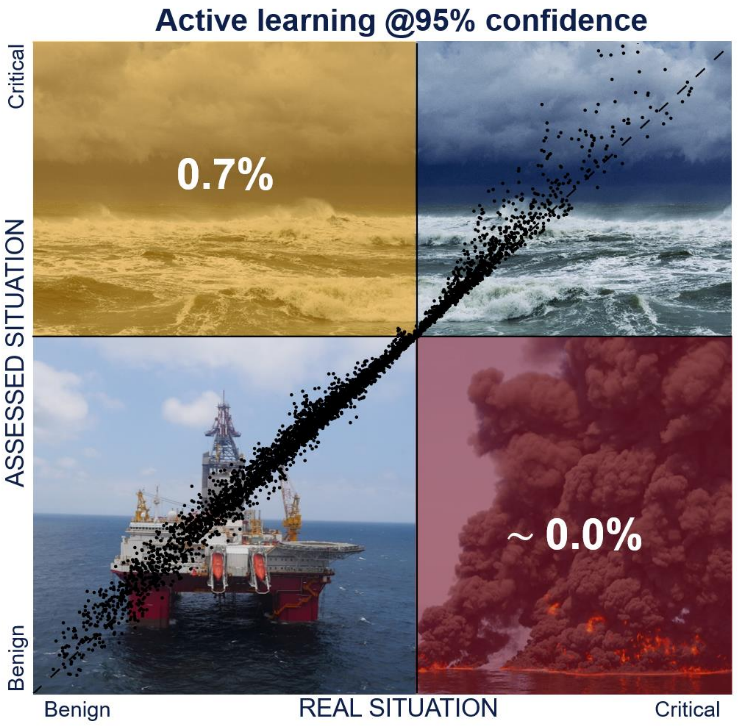
Standard approach





Optimized for safety-critical decisions





WHEN TRUST MATTERS

www.dnv.com