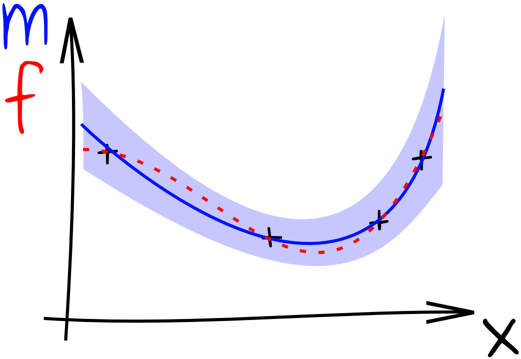
École chercheurs MEXICO, La Rochelle, Mars 2018

Introduction to statistical modelling

Nicolas Durrande, nicolas@prowler.io

PROWLER.io, Cambridge – Mines St-Étienne

How to build **statistical models**?



In the sequel, we will use the following notations :

- The set of observation points is a $n \times d$ matrix X
- The vector of observations is F : $F_i = f(X_i)$ (or $F = f(X)$).

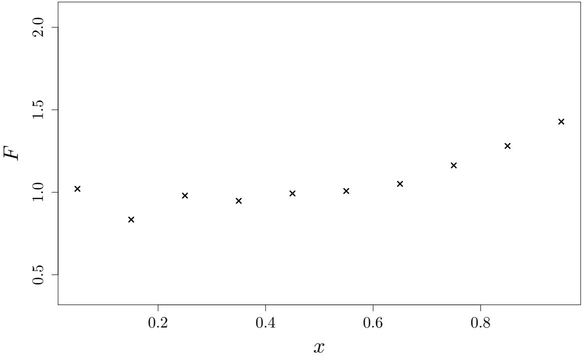
We will now discuss two types of statistical models:

- Linear regression
- Gaussian process regression

Linear Regression

Example

If we consider the following observations:

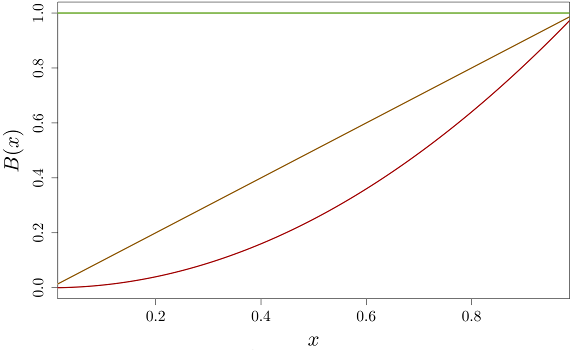


Example

We assume the observations are drawn from

$$F_i = \sum_{k=0}^{2} \beta_k b_k(X_i) + \varepsilon_i \qquad (= B(X_i)\beta + \varepsilon_i)$$

with $b_0(x) = 1,\ b_1(x) = x,\ b_2(x) = x^2$, unknown β_i and i.i.d ε_i.

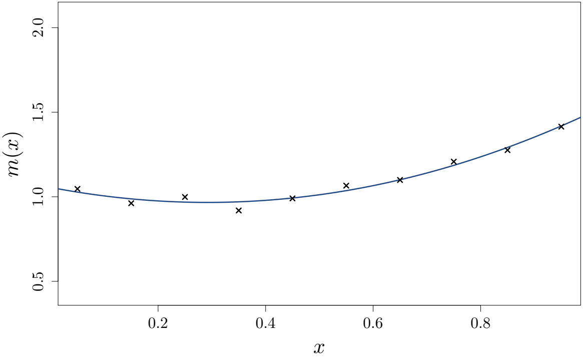


Example

The best linear unbiased estimator of β is

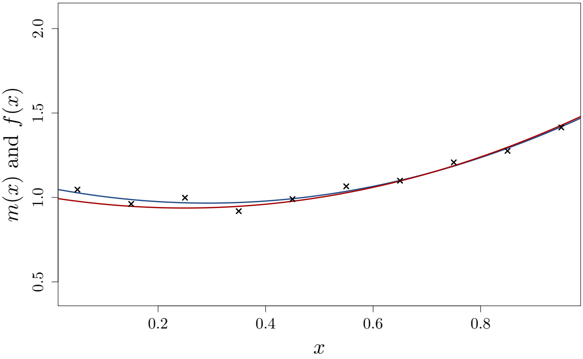
$$\hat{\beta} = (B(X)^t B(X))^{-1} B(X)^t F.$$

We obtain $\hat{\beta} = (1.06, -0.61, 1.04)^T$ and the model is:



Example

There is of course an error between the true generative function and the model



Can this error be quantified?

The estimator can also be seen as a random variable:

$$\hat{\beta} = (B(X)^t B(X))^{-1} B(X)^t (B(X)\beta + \varepsilon).$$

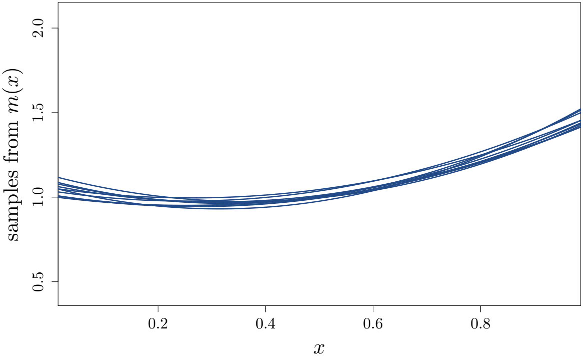
- Its expectation is β $\Rightarrow$ The estimator is unbiased
- Its covariance matrix is

$$(B(X)^t B(X))^{-1} B(X)^t \operatorname{cov}[\varepsilon, \varepsilon^t] B(X) (B(X)^t B(X))^{-1}$$

- If ε is multivariate normal, then $\hat{\beta}$ is also multivariate normal.

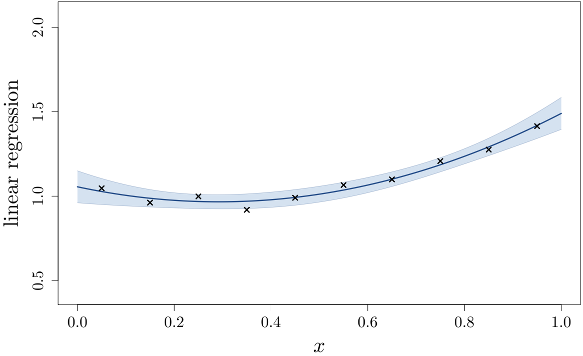
Back to the example

Be obtain uncertainty on the model



Back to the example

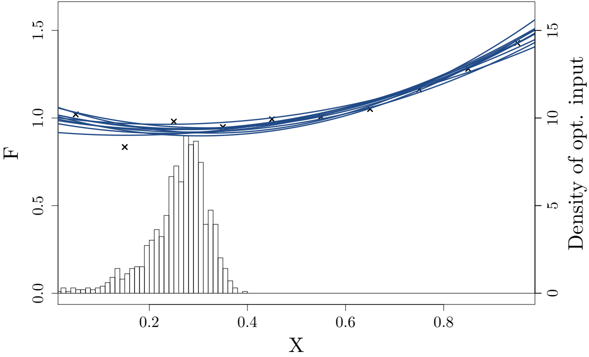
The previous picture can be summarized by showing the mean of *m* and 95% confidence intervals



This statistical model can be used for **uncertainty quantification**:

Back to the example

If we are interested in the value $x^\star$ minimizing $f(x)$:



we obtain a distribution for $x^\star$.

We could dedicate the entire course to linear regression models...

- model validation
- choice of basis functions
- influence of input locations
- ...

We will just stress a few **pros and cons of these models**:

+ provide a good noise filtering

+ are easy to interpret

– are not flexible (need to choose the basis functions)

– do not interpolate

– may explode when using high order polynomials (over-fitting)

Gaussian Process Regression

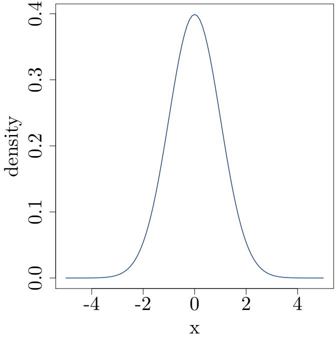
This section is organised in 3 subsections:

1. Univariate and multivariate normal distributions
2. Gaussian processes
3. Gaussian process regression

1D normal distribution

We say that $X \sim \mathcal{N}(\mu, \sigma^2)$ if it has the following pdf:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$



The distribution is characterised by

mean: $\mu = \mathrm{E}[X]$

variance: $\sigma^2 = \mathrm{E}[X^2] - \mathrm{E}[X]^2$

One fundamental property: a linear combination of independant normal distributed random variables is still normal distributed.

Multivariate normal distribution

Definition

We say that a vector $Y = (Y_1, \dots, Y_n)^t$ follows a multivariate normal distribution if any linear combination of Y follows a normal distribution:

$$\forall \alpha \in \mathbb{R}^n,\ \alpha^t Y \sim \mathcal{N}$$

The distribution of a Gaussian vector is characterised by

- a **mean vector** $\mu = \mathrm{E}[Y]$
- a **covariance matrix** $\Sigma = \mathrm{E}[YY^t] - \mathrm{E}[Y]\,\mathrm{E}[Y]^t$

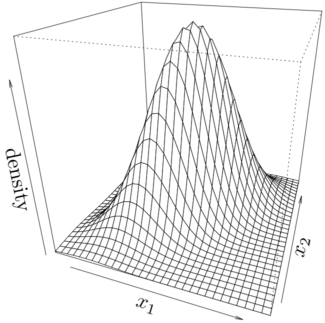
Property:

A covariance matrix is

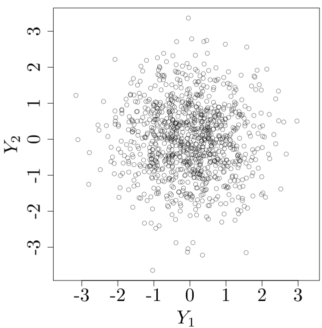
- symmetric $K_{i,j} = K_{j,i}$
- positive semi-definite $\forall \alpha \in \mathbb{R}^n, \alpha^t K \alpha \geq 0$.

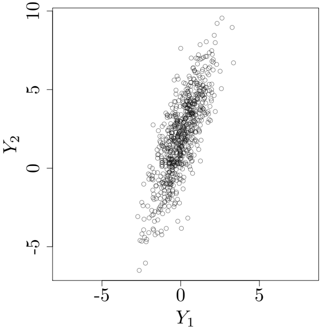
The pdf of a multivariate Gaussian is:

$$f_Y(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^t K^{-1}(x-\mu)\right).$$



Samples from a multivariate normal

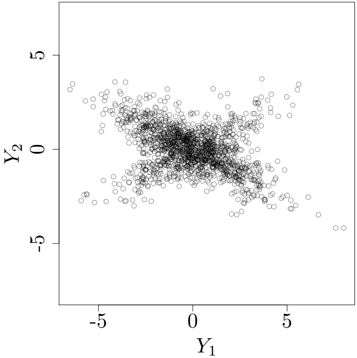




Exercise

- For $X = (X_1, \dots, X_n)$ with X_i independent and $\mathcal{N}(0, 1)$, and a $n \times n$ matrix A, what is the distribution of AX?
- For a given covariance matrix K and independent $\mathcal{N}(0, 1)$ samples, how can we generate $\mathcal{N}(\mu, K)$ random samples?

Counter example



Y_1 and Y_2 are normally distributed but **the couple** (Y_1, Y_2) **is not**.

Conditional distribution

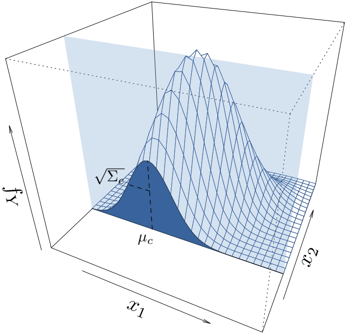
2D multivariate Gaussian conditional distribution:

$$p(y_1|y_2=\alpha) = \frac{p(y_1,\alpha)}{p(\alpha)}$$

$$= \frac{\exp(\text{quadratic in } y_1 \text{ and } \alpha)}{\text{const}}$$

$$= \frac{\exp(\text{quadratic in } y_1)}{\text{const}}$$

$$= \text{Gaussian distribution!}$$



The conditional distribution is still Gaussian!

Conditional distribution

Let (Y_1, Y_2) be a Gaussian vector (Y_1 and Y_2 may both be vectors):

$$\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} = \mathcal{N}\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \right).$$

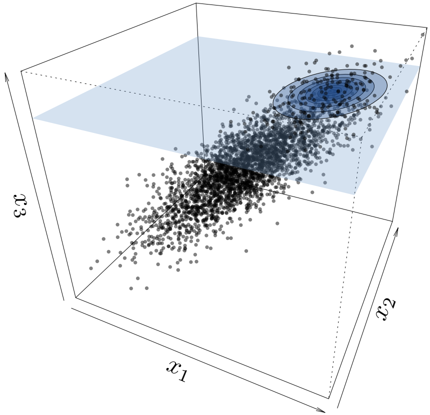
The conditional distribution of Y_1 given Y_2 is:

$$Y_1 | Y_2 \sim \mathcal{N}(\mu_{\mathrm{cond}}, \Sigma_{\mathrm{cond}})$$

with
$$\begin{aligned} \mu_{\mathrm{cond}} &= \mathrm{E}[Y_1 | Y_2] = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(Y_2 - \mu_2) \\ \Sigma_{\mathrm{cond}} &= \mathrm{cov}[Y_1, Y_1 | Y_2] = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \end{aligned}$$

3D Example

3D multivariate Gaussian conditional distribution:



2. Gaussian processes

The multivariate Gaussian distribution can be generalised to random processes:

Definition

A random process Z over $D \subset \mathbb{R}^d$ is said to be Gaussian if

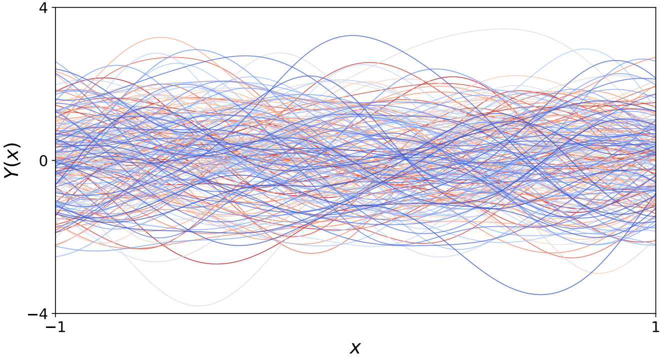
$$\forall n \in \mathbb{N}, \forall x_i \in D, (Z(x_1), \dots, Z(x_n)) \text{ is a Gaussian vector.}$$

The distribution of a GP is fully characterised by:

- its mean function m defined over D
- its covariance function (or kernel) k defined over $D \times D$: $k(x, y) = \mathrm{cov}(Z(x), Z(y))$

We will use the notation $Z \sim \mathcal{N}(m(.), k(., .))$.

Let’s look at the sample paths of a Gaussian Process!



Exercise: Simulating sample paths

Let X be a set 100 regularly spaced points over the input space of Z.

- What is the distribution of $Z(X)$?
- How to simulate samples from $Z(X)$?

A kernel satisfies the following properties:

- It is symmetric: $k(x, y) = k(y, x)$
- It is positive semi-definite (psd):

$$\forall n \in \mathbb{N}, \forall x_i \in D, \forall \alpha \in \mathbb{R}^n, \ \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \geq 0$$

Furthermore any symmetric psd function can be seen as the covariance of a Gaussian process. This equivalence is known as the Loeve theorem.

There are a lot of functions that have already been proven psd:

constant	$k(x,y) = \sigma^2$
white noise	$k(x,y) = \sigma^2 \delta_{x,y}$
Brownian	$k(x,y) = \sigma^2 \min(x,y)$
exponential	$k(x,y) = \sigma^2 \exp\left(-\lvert x-y\rvert/\theta\right)$
Matern 3/2	$k(x,y) = \sigma^2 \left(1+\lvert x-y\rvert\right) \exp\left(-\lvert x-y\rvert/\theta\right)$
Matern 5/2	$k(x,y) = \sigma^2 \left(1+\lvert x-y\rvert/\theta + 1/3\lvert x-y\rvert^2/\theta^2\right) \exp\left(-\lvert x-y\rvert/\theta\right)$
squared exponential	$k(x,y) = \sigma^2 \exp\left(-(x-y)^2/\theta^2\right)$
$\vdots$	

The parameter σ^2 is called the **variance** and θ the **length-scale**.

⇒ Shiny App:

`https://github.com/NicolasDurrande/shinyApps`

Here is a list of the most common kernels in higher dimension:

constant	$k(x,y) = \sigma^2$
white noise	$k(x,y) = \sigma^2 \delta_{x,y}$
exponential	$k(x,y) = \sigma^2 \exp\left(-\|\|x-y\|\|_\theta\right)$
Matern 3/2	$k(x,y) = \sigma^2 \left(1+\sqrt{3}\|\|x-y\|\|_\theta\right) \exp\left(-\sqrt{3}\|\|x-y\|\|_\theta\right)$
Matern 5/2	$k(x,y) = \sigma^2 \left(1+\sqrt{5}\|\|x-y\|\|_\theta + \frac{5}{3}\|\|x-y\|\|_\theta^2\right) \exp\left(-\sqrt{5}\|\|x-y\|\|_\theta\right)$
Gaussian	$k(x,y) = \sigma^2 \exp\left(-\frac{1}{2}\|\|x-y\|\|_\theta^2\right)$

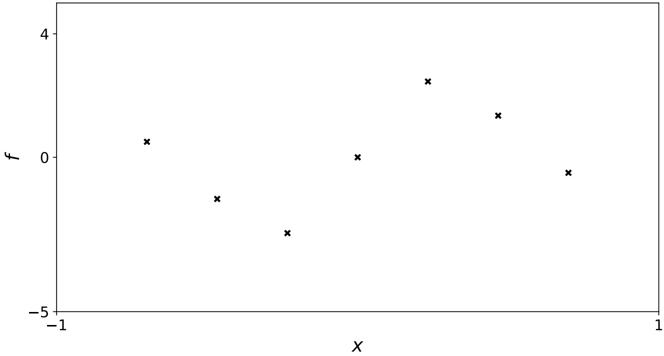
where

$$||x-y||_\theta = \left(\sum_{i=1}^{d} \frac{(x_i - y_i)^2}{\theta_i^2} \right)^{1/2}.$$

⇒ R demo

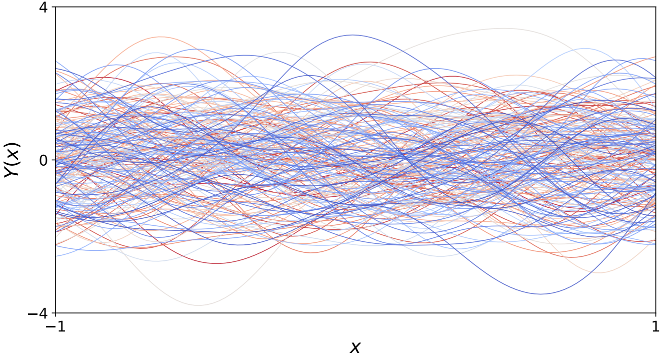
Gaussian process regression

We assume we have observed a function f for a set of points $X = (X_1, \dots, X_n)$:



The vector of observations is $F = f(X)$ (ie $F_i = f(X_i)$).

Since f in unknown, we make the general assumption that it is the sample path of a Gaussian process $Z \sim \mathcal{N}(0, k)$:



The posterior distribution $Y(\cdot)|Y(X)=F$:

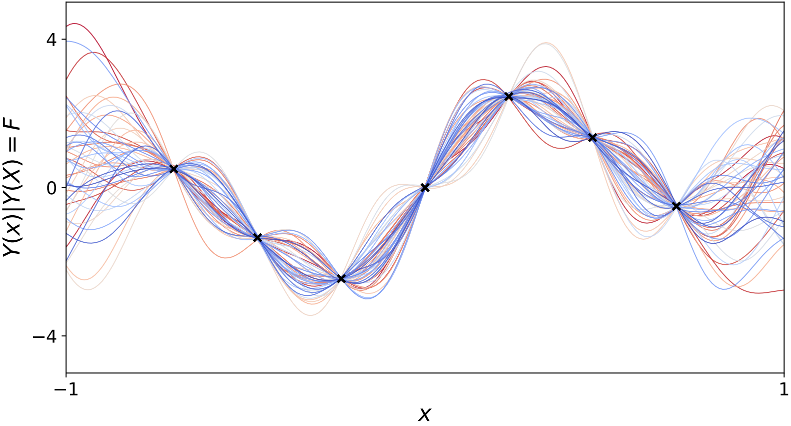
- Is still Gaussian
- Can be computed analytically

It is $\mathcal{N}(m(\cdot), c(\cdot,\cdot))$ with:

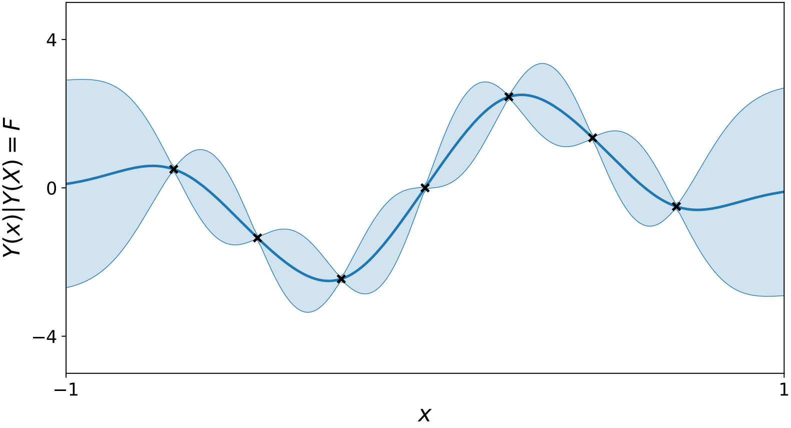
$$\begin{aligned} m(x) &= \mathrm{E}[Y(x)|Y(X){=}F] \\ &= k(x,X)k(X,X)^{-1}F \end{aligned}$$

$$\begin{aligned} c(x,y) &= \mathrm{cov}[Y(x),Y(y)|Y(X){=}F] \\ &= k(x,y) - k(x,X)k(X,X)^{-1}k(X,y) \end{aligned}$$

Samples from the posterior distribution



It can be summarized by a mean function and 95% confidence intervals.



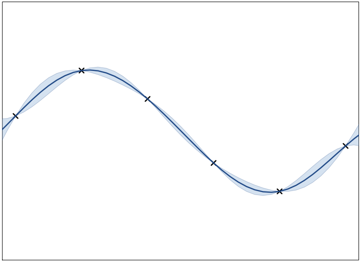
A few remarkable properties of GPR models

- They (can) interpolate the data-points
- The prediction variance does not depend on the observations
- The mean predictor does not depend on the variance parameter
- They (usually) come back to zero when we are far away from the observations.

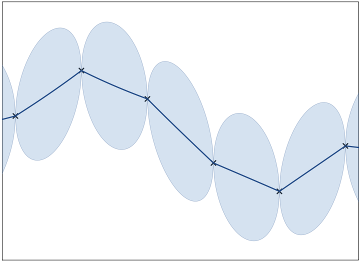
Can we prove them?

Changing the kernel has a huge impact on the model:

Gaussian kernel:

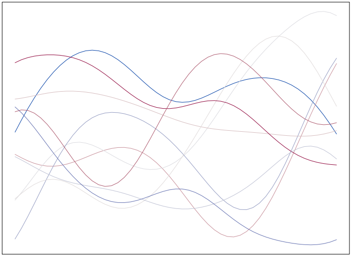


Exponential kernel:

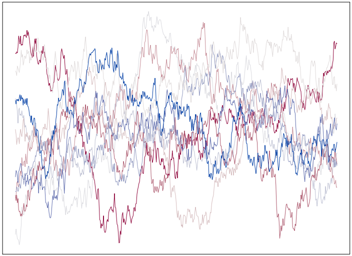


This is because changing the kernel means changing the prior on f

Gaussian kernel:

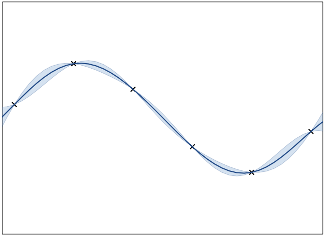


Exponential kernel:

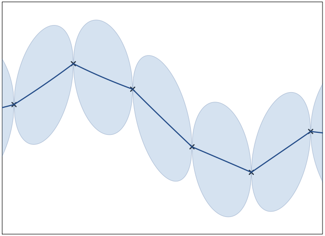


There is no kernel that is intrinsically better... it depends on data!

Gaussian kernel:



Exponential kernel:



The kernel has to be chosen accordingly to our prior belief on the behaviour of the function to study:

- is it continuous, differentiable, how many times?
- is it stationary ?
- ...

$\Rightarrow$ R volcano demo

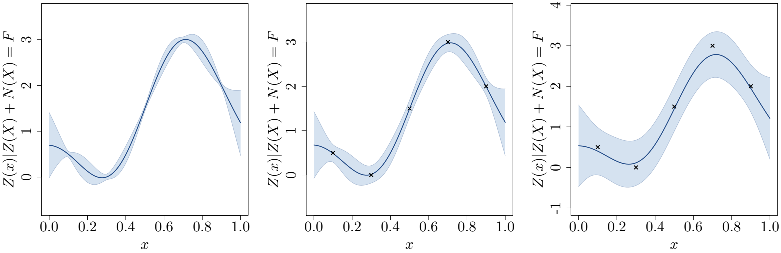
We are not always interested in models that interpolate the data. For example, if there is some observation noise: $F = f(X) + \varepsilon$. Let

N be a process $\mathcal{N}(0, n(.,.))$ that represent the observation noise. The expressions of GPR with noise are

$$\begin{aligned} m(x) &= \mathrm{E}[Z(x)|Z(X) + N(X){=}F] \\ &= k(x,X)(k(X,X) + n(X,X))^{-1}F \end{aligned}$$

$$\begin{aligned} c(x,y) &= \mathrm{cov}[Z(x), Z(y)|Z(X) + N(X){=}F] \\ &= k(x,y) - k(x,X)(k(X,X) + n(X,X))^{-1}k(X,y) \end{aligned}$$

Examples of models with observation noise for $n(x,y) = \tau^2 \delta_{x,y}$:



The values of τ^2 are respectively 0.001, 0.01 and 0.1.

Parameter estimation

We have seen previously that the choice of the kernel and its parameters have a great influence on the model.

In order to choose a prior that is suited to the data at hand, we can consider:

- minimising the model error
- Using maximum likelihood estimation

We will now detail the second one.

Definition

The **likelihood** of a distribution with a density f_X given some observations $X_1, \dots, X_p$ is:

$$L = \prod_{i=1}^{p} f_X(X_i)$$

This quantity can be used to measure the adequacy between observations and a distribution.

In the GPR context, we often have only **one observation** of the vector F. The likelihood is then:

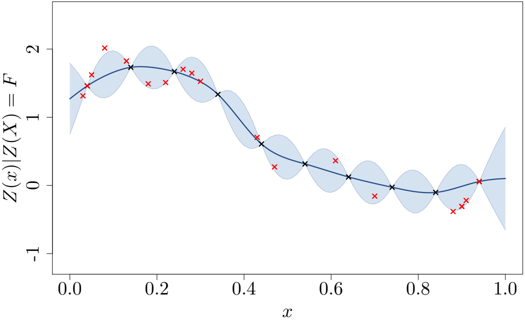
$$L = f_{Z(X)}(F) = \frac{1}{(2\pi)^{n/2}|k(X,X)|^{1/2}} \exp\left(-\frac{1}{2}F^t k(X,X)^{-1} F\right).$$

It is thus possible to maximise L – or $\log(L)$ – with respect to the kernel's parameters in order to find a well suited prior.

⇒ R demo

Model validation

The idea is to introduce new data and to compare the model prediction with reality



Since GPR models provide a mean and a covariance structure for the error they both have to be assessed.

Let X_t be the test set and $F_t = f(X_t)$ be the associated observations.

The accuracy of the mean can be measured by computing:

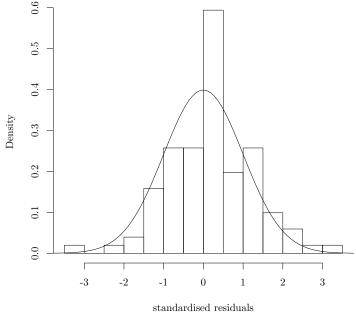
$$\begin{aligned} \text{Mean Square Error} \qquad & MSE = \mathrm{mean}((F_t - m(X_t))^2) \\ \text{A ``normalised'' criterion} \qquad & Q_2 = 1 - \frac{\sum (F_t - m(X_t))^2}{\sum (F_t - \mathrm{mean}(F_t))^2} \end{aligned}$$

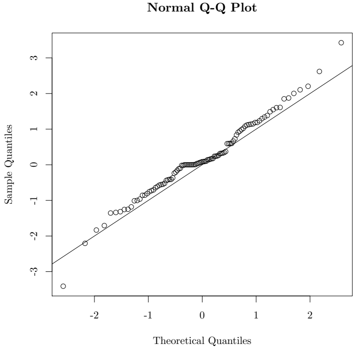
On the above example we get $MSE = 0.038$ and $Q_2 = 0.95$.

The predicted distribution can be tested by normalising the residuals.

According to the model, $F_t \sim \mathcal{N}(m(X_t), c(X_t, X_t))$.

$c(X_t, X_t)^{-1/2}(F_t - m(X_t))$ should thus be independents $\mathcal{N}(0, 1)$:




Normal Q-Q Plot

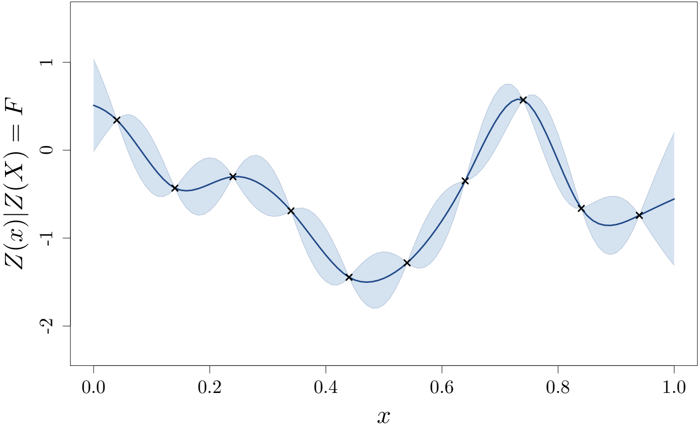
When no test set is available, another option is to consider cross validation methods such as leave-one-out.

The steps are:

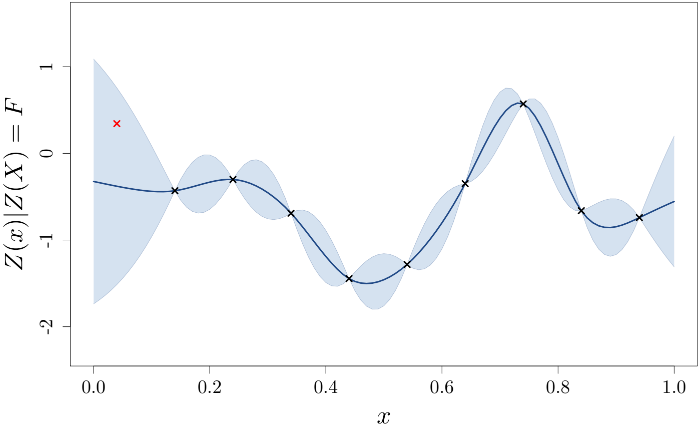
1. build a model based on all observations except one
2. compute the model error at this point

This procedure can be repeated for all the design points in order to get a vector of error.

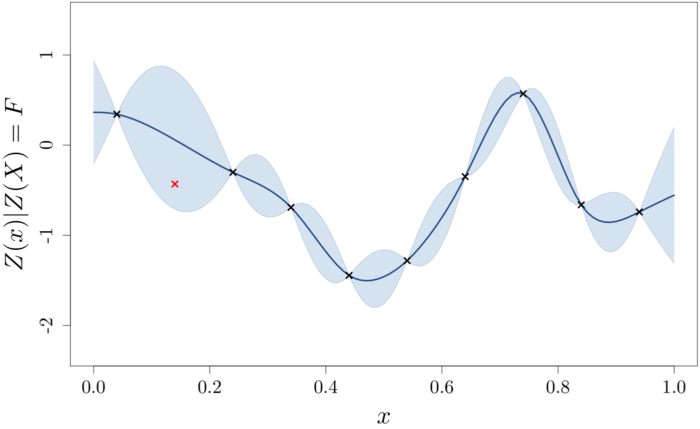
Model to be tested:



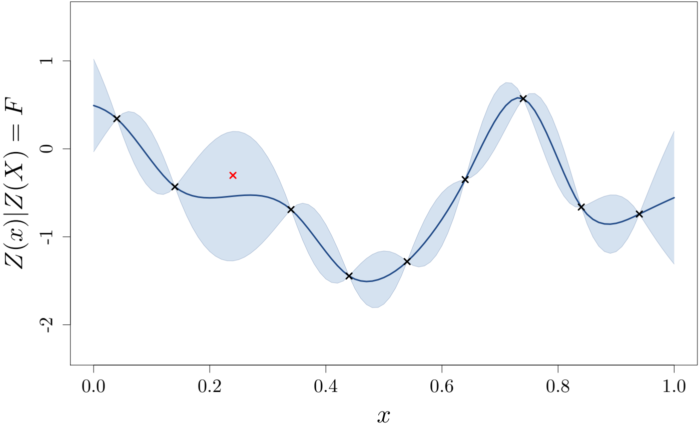
Step 1:



Step 2:



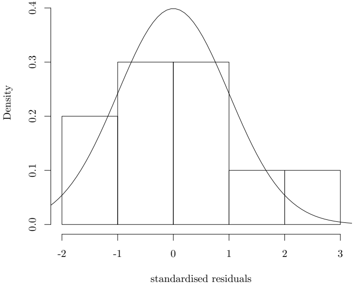
Step 3:

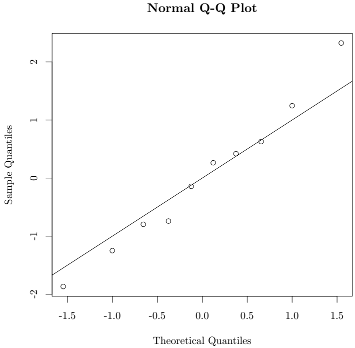


We finally obtain:

$$MSE = 0.24 \text{ and } Q_2 = 0.34.$$

We can also look at the residual distribution. For leave-one-out, there is no joint distribution for the residuals so they have to be standardised independently.

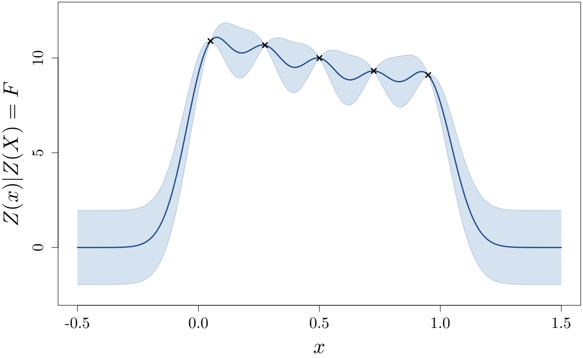




GPR with trend

We have seen that GPR models go back to zero if we consider a centred prior.

This behaviour is not always wanted

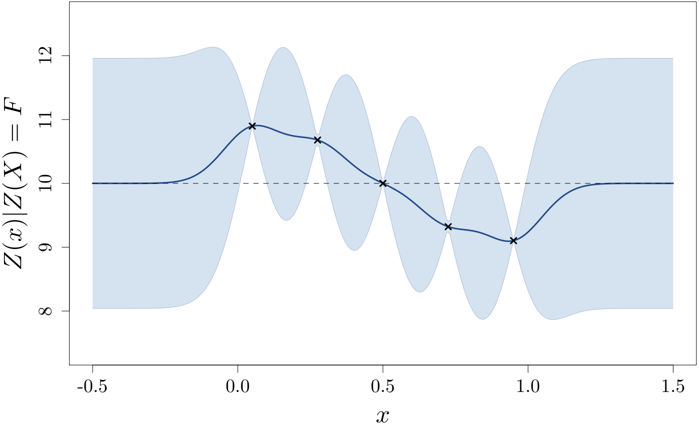


If the trend $t(.)$ is known, the usual formulas for multivariate normal conditional distribution apply:

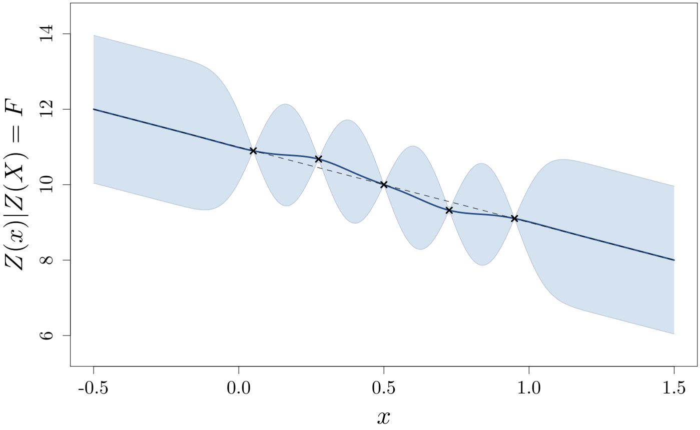
$$\begin{aligned} m(x) &= \mathrm{E}[Z(x)|Z(X){=}F] \\ &= t(x) + k(x,X)k(X,X)^{-1}(F - t(X)) \\ c(x,y) &= \mathrm{cov}[Z(x), Z(y)|Z(X){=}F] \\ &= k(x,y) - k(x,X)k(X,X)^{-1}k(X,y) \end{aligned}$$

We can see that the trend is subtracted first and then added in the end.

In the previous example, we can consider that trend is constant $t(x) = 10$:



We can also try a linear trend $t(x) = 11 - 2x$:

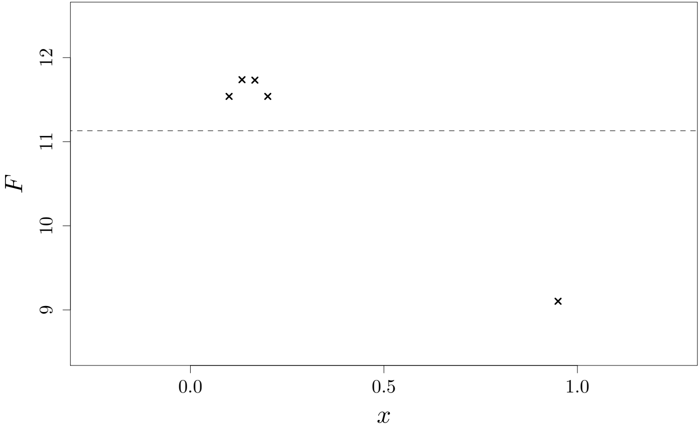


In practice, the trend is often unknown... The question is then how to estimate it.

We will distinguish:

- **simple kriging**: there is no trend or it is known
- **ordinary kriging**: the trend is a constant
- **universal kriging**: the trend is given by basis functions

First idea: considering the mean value $t(x) = \text{mean}(F)$... this is not ideal!



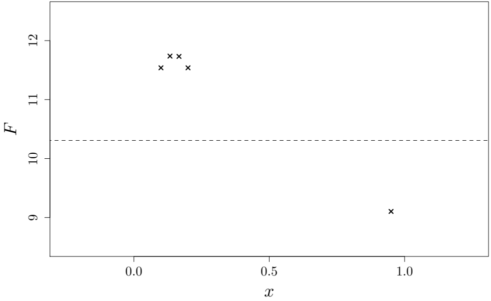
Any other idea?

In practice the right thing to to is to find the maximum likelihood value:

$$L(t) = \frac{1}{(2\pi)^{n/2}|k(X,X)|^{1/2}} \exp\left(-\frac{1}{2}(F - t\mathbf{1})^t k(X,X)^{-1}(F - t\mathbf{1})\right)$$

We obtain:

$$\hat{t} = \frac{\mathbf{1}^t k(X,X)^{-1} F}{\mathbf{1}^t k(X,X)^{-1}\mathbf{1}}$$



The expression of the **best predictor** is given by the usual conditioning of a GP:

$$m(x) = \mathrm{E}[Z(x)|Z(X) = F] = \hat{t} - k(x, X)k(X, X)^{-1}(F - \hat{t})$$

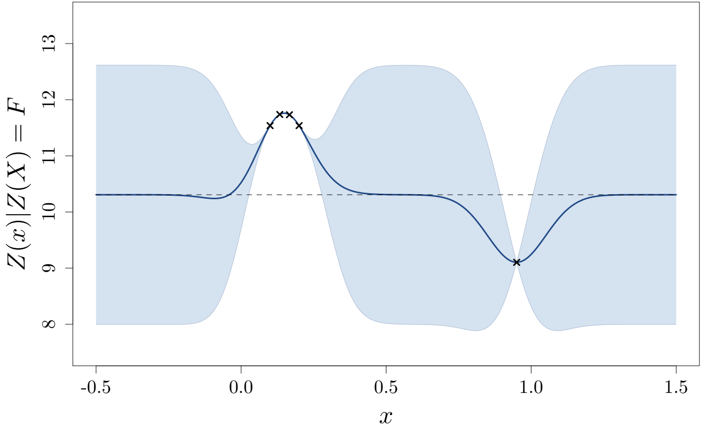
Regarding the **model variance**, it must account for the estimator's variance. We will use the law of total Variance :

$$\mathrm{var}[X] = \mathrm{E}[\mathrm{var}(X|Y)] + \mathrm{var}[\mathrm{E}(X|Y)]$$

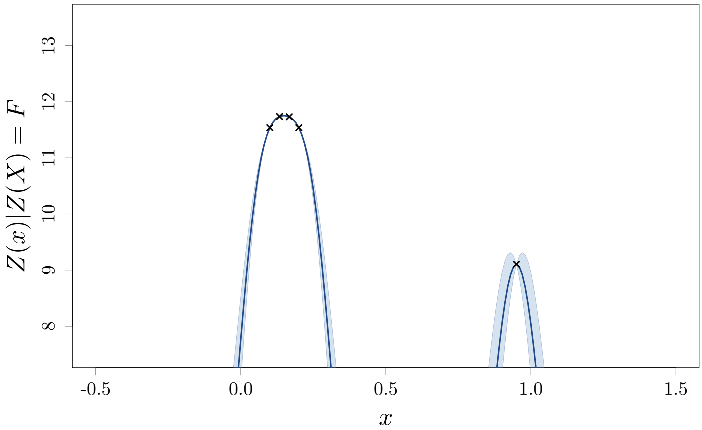
If we apply this to the GPR variance prediction we get:

$$\begin{aligned}\mathrm{var}[Z(x)|Z(X)] = {} & k(x, x) - k(x, X)k(X, X)^{-1}k(X, x) \\ & + \frac{(\mathbf{1} + k(x, X)k(X, X)^{-1}\mathbf{1})^t(\mathbf{1} + k(x, X)k(X, X)^{-1}\mathbf{1})}{\mathbf{1}^t k(X, X)^{-1}\mathbf{1}}\end{aligned}$$

On the previous example we obtain:



it can be compared with simple kriging



If the trend is not constant but linear, quadratic, etc. it is interesting to consider the following probabilistic model for the prior:

$$Z(x) = Y(x) + \sum_i \beta_i h_i(x)$$

where the $h_i(x)$ are basis functions and the β_i are unknown scalars.

As previously, we can consider the maximum likelihood estimator

$$\hat{\beta} = (H^t k(X,X)^{-1} H)^{-1} H^t k(X,X)^{-1} F$$

where H is the matrix of general term $H_{i,j} = h_j(X_i)$.

The final equations are very similar to ordinary kriging:

Universal kriging

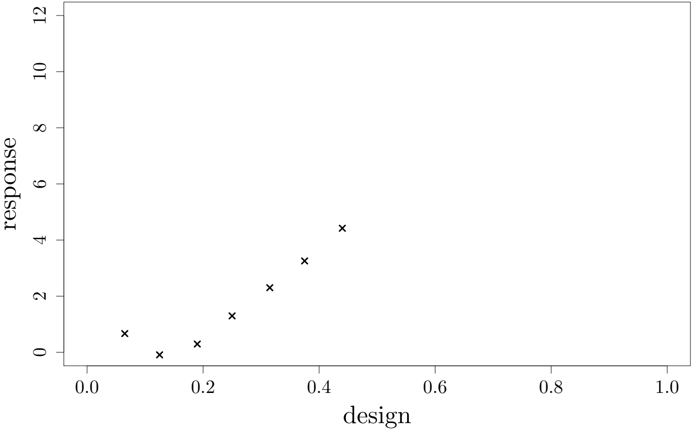
$$\begin{aligned} m(x) &= h(x)^t\hat{\beta} - k_x K^{-1}(F - h(X)^t\hat{\beta}) \\ c(x,y) &= k(x,y) - k_x K^{-1} k_y^t \\ &\qquad + (h(x)^t + k_x K^{-1} H)^t (H^t K^{-1} H)^{-1} (h(y)^t + k_y K^{-1} H) \end{aligned}$$

where $k_x = k(x, X)$ and $K = k(X, X)$.

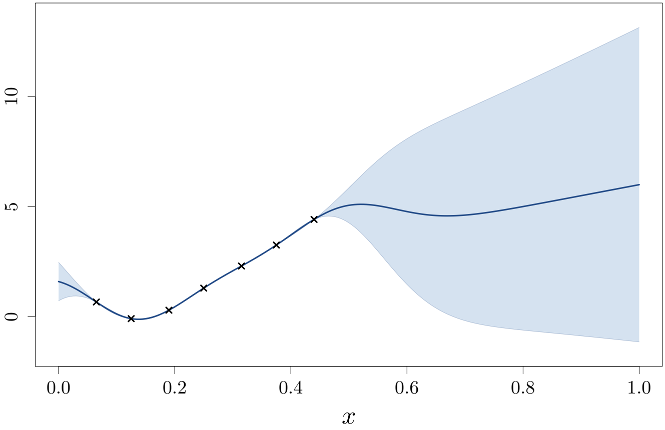
Remarks

- Ordinary kriging is a special case of universal kriging with only one constant basis function.
- The model always interpolates whatever $\hat{\beta}$ is.
- the trend part can be seen as generalised least square (regression with correlated residuals)

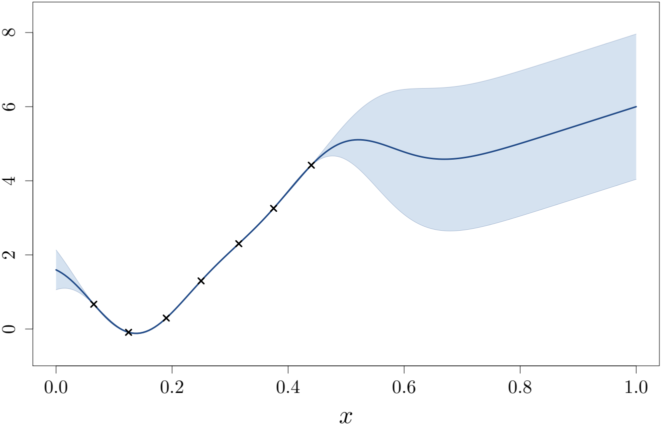
We consider the following example



Universal kriging model with linear trend: $h_1(x) = 1$, $h_2(x) = x$.



It can be compared to simple kriging with known trend



GPR in practice

The various steps for building a GPR model are:

1. Create a DoE
   - What is the overall evaluation budget?
   - What is my model for?
2. Choose a kernel
3. Estimate the parameters
   - Maximum likelihood
   - Cross-validation
   - Multi-start
4. Validate the model
   - Test set
   - Leave-one-out to check mean and confidence intervals
   - Leave-*k*-out to check predicted covariances

Remarks

- It is common to iterate over steps 2, 3 and 4.

In practice, the following errors may appear:

- Error: the matrix is not invertible
- Error: the matrix is not positive definite

In practice, invertibility issues may arise if observations points are close-by.

This is specially true if

- the kernel corresponds to very regular sample paths (squared-exponential for example)
- the range (or length-scale) parameters are large

In order to avoid numerical problems during optimization, one can:

- add a (very) small observation noise
- impose a maximum bound to length-scales
- impose a minimal bound for noise variance
- avoid the Gaussian kernel

A few words on GPR **Complexity**

- Storage footprint: We have to store the covariance matrix which is $n \times n$.
- Complexity: We have to invert the covariance matrix, which requires is $\mathcal{O}(n^3)$.

Storage footprint is often the first limit to be reached.

The maximal number of observation points is between 1000 and 10 000.

Note that the complexity do not depend on the dimension of the input space!

Conclusion

Important points:

- Statistical models are useful when little data is available. they allow to
  - interpolate or approximate functions
  - Compute quantities of interests (such as mean value, optimum, ...)
  - Get an error measure
- GPR is similar to linear regression but the assumption is much weaker (not a finite dimensional space)

Reference

Carl Edward Rasmussen and Chris Williams, *Gaussian processes for machine learning*, MIT Press, 2006. (free version online).