Analysis of complex simulation experiments with Gaussian processes

Mickaël Binois
Inria Sophia Antipolis - Acumes
joint works with D. Ginsbourger (UniBE), R. Gramacy (VT), A. Habbal (UCA), V. Picheny (Secondmind), O. Roustant (Insa)

Séminaire Calisto
2 Juillet 2021

Problem description

Let us consider an expensive-to-evaluate **black box** simulator:

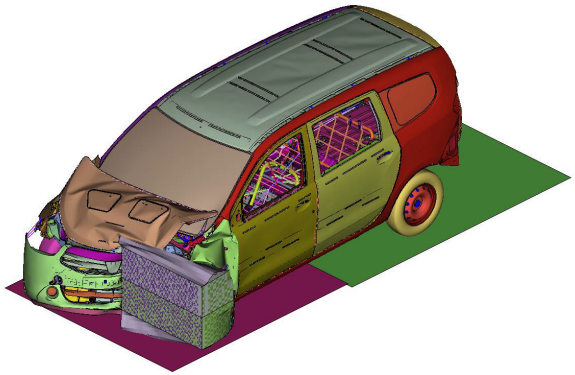
$$f : \mathcal{X} \subset \mathbb{R}^d \to \mathbb{R}^m.$$

Here, $\mathcal{X} = [-1, 1]^d$, corresponding to box constraints. In addition:

- only noisy evaluations of f may be possible;
- some data may be available too.

Common occurrence in engineering, physics, operations research, epidemiology, ML, ...

Examples of (stochastic) simulators:

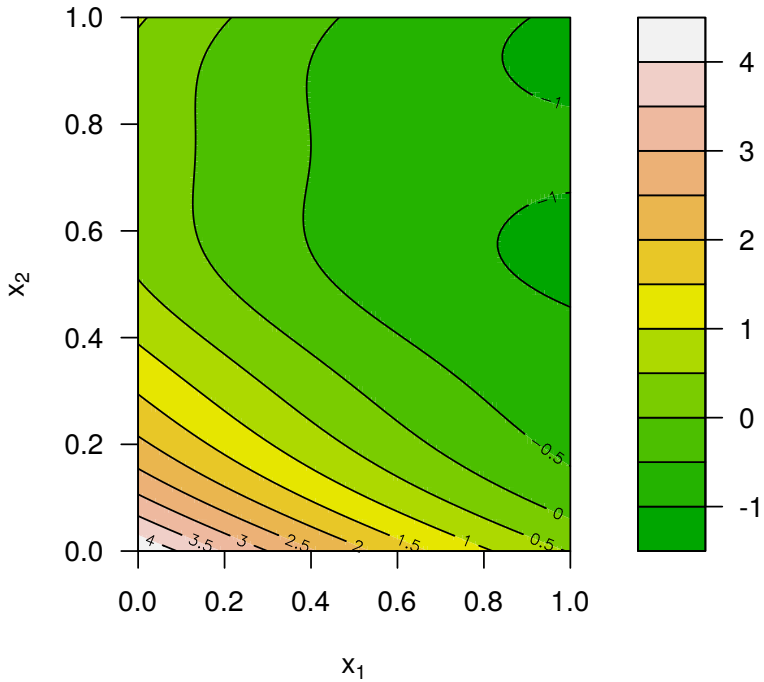


Car crash-worthiness

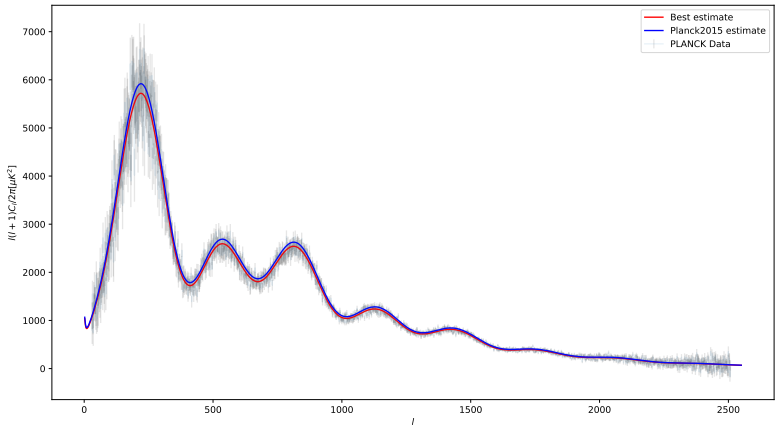
Cosmology

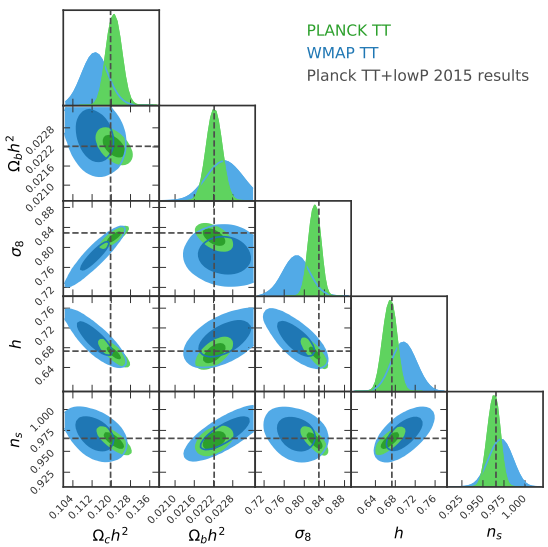
Applications

Optimization or safety



Calibration





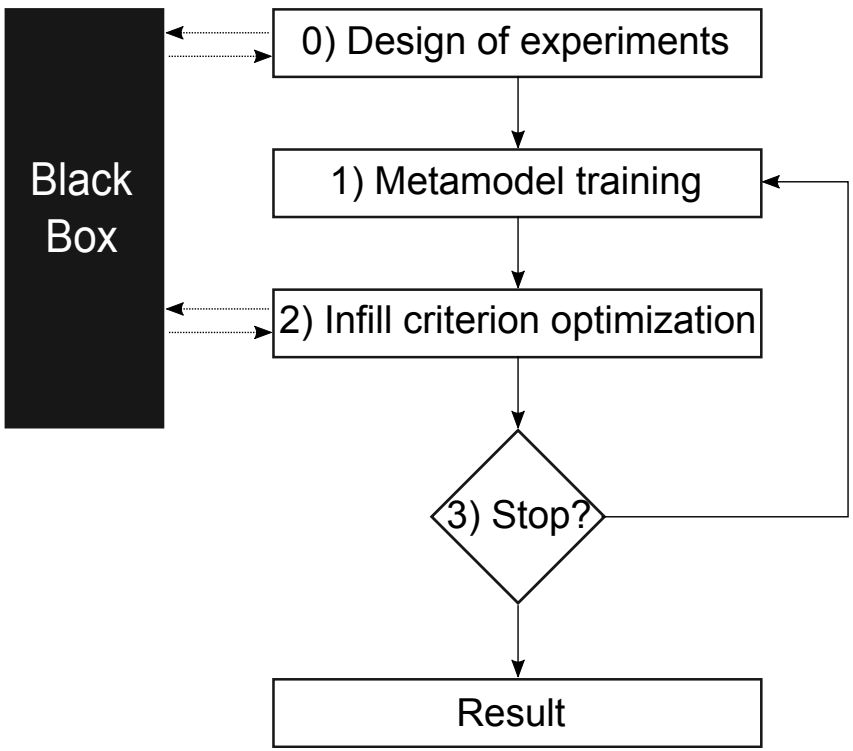
Also: sensitivity analysis, dimension reduction,...

Plan

Surrogate based sequential design procedure

Bayesian optimization [Mockus, 1989]

Sequential design strategy based on a distribution over functions to define an acquisition function.



For instance:

0. Maximin Latin Hypercubes Samples
1. Gaussian process model
2. Expected Improvement
3. Budget

Gaussian processes

Definition (Gaussian vector)

A d-dimensional random vector Y is Gaussian iif $\forall a \in \mathbb{R}^d$, $a^\top Y$ is Gaussian.

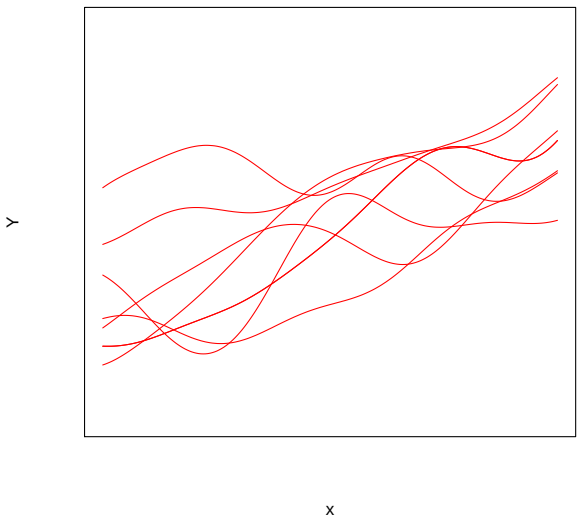
Definition (Gaussian process)

A random process Y indexed by D is said to be Gaussian iif $\forall \mathbf{x}_i \in D, \forall n \in \mathbb{N}$, $(Y(\mathbf{x}_1), \ldots, Y(\mathbf{x}_n))$ is a Gaussian vector.

GPs are fully characterized with their mean and covariance functions.

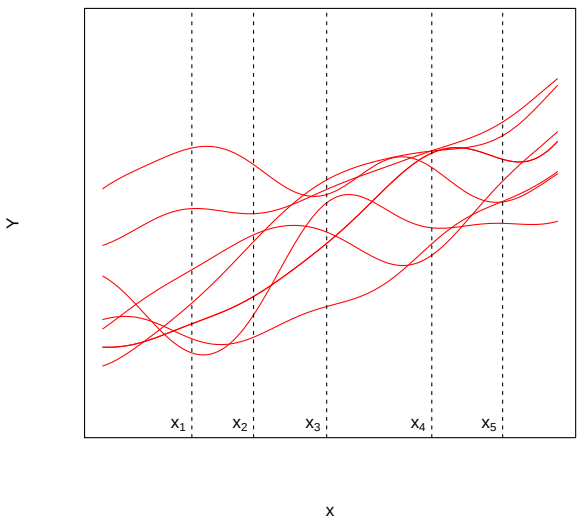
Gaussian processes

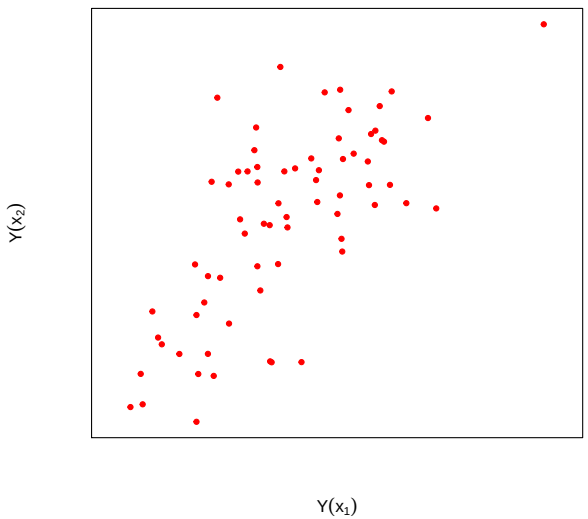
Same with images:



Gaussian processes

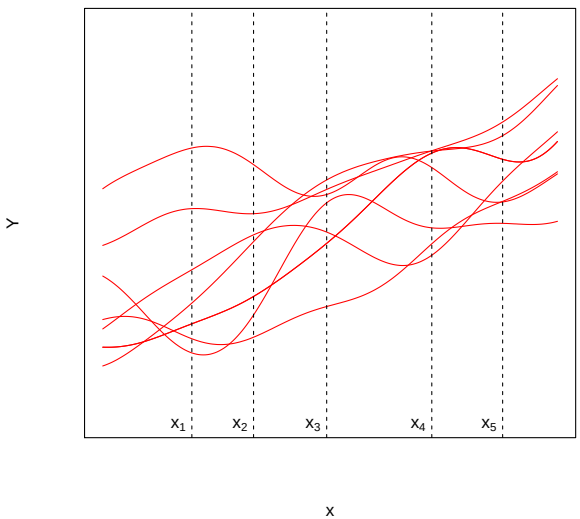
Same with images:

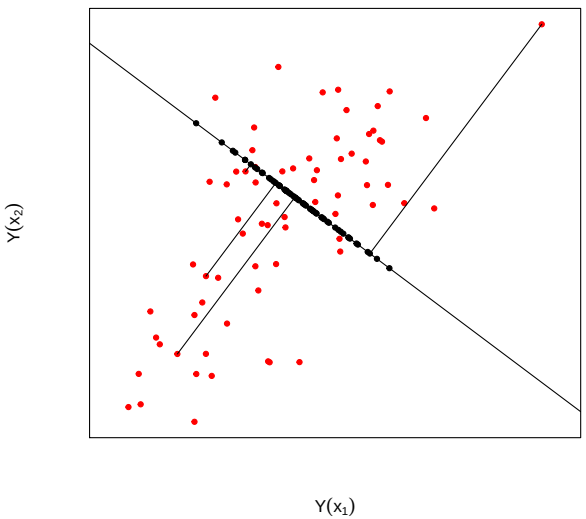




Gaussian processes

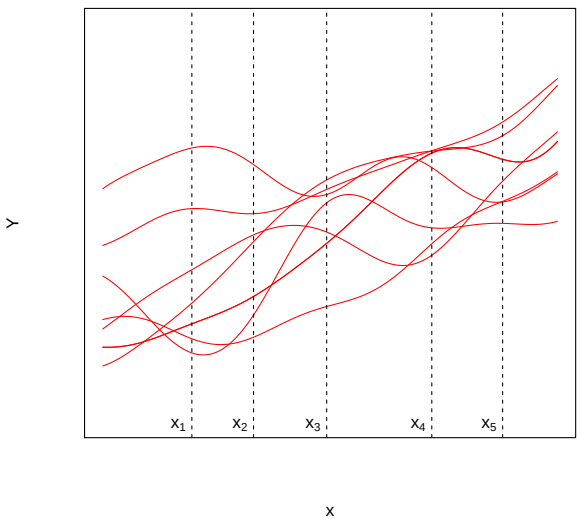
Same with images:

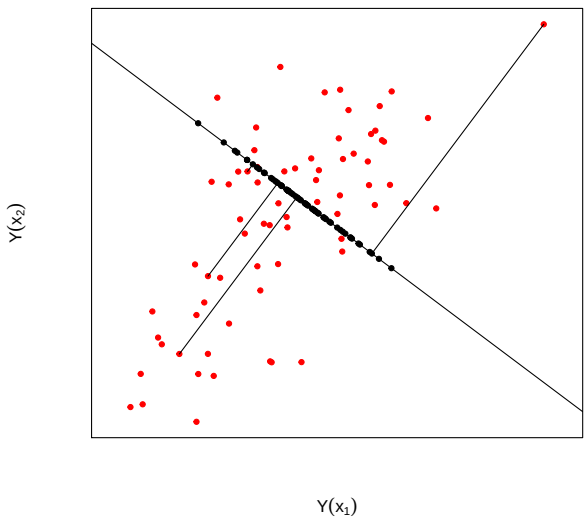


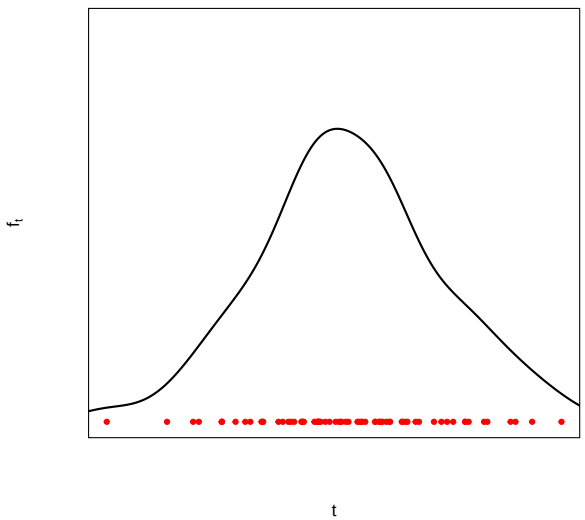


Gaussian processes

Same with images:







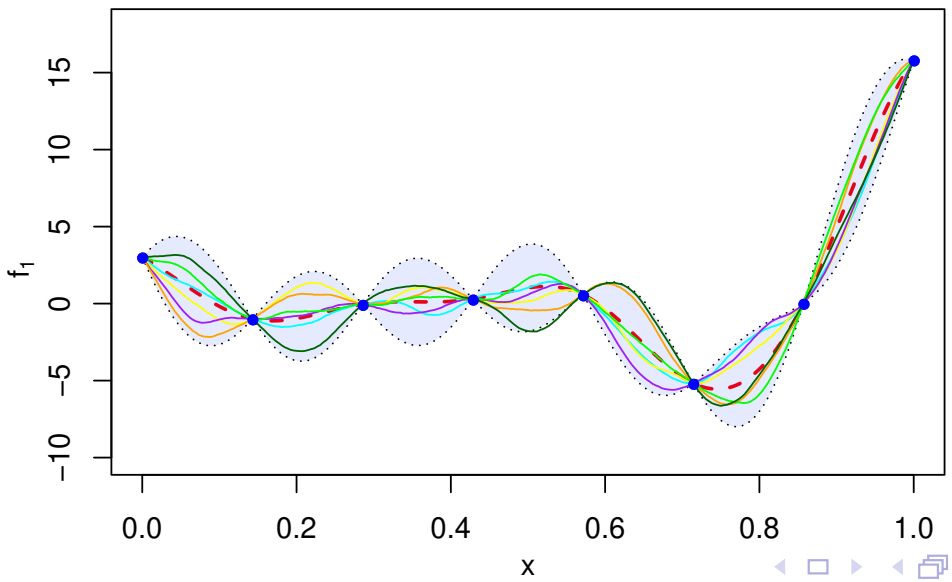
Gaussian process regression

We use a zero mean GP prior on y, with covariance k: $Y \sim \mathcal{GP}(0, k)$.

MVN conditional identities give directly the result on $(\mathbf{x}_i, y_i)_{1 \leq i \leq N}$:

$$Y|\mathbf{y} \sim \mathcal{GP}(\mu, \sigma^2) \text{ with}$$
$$m_n(\mathbf{x}) = \mathbb{E}(Y(\mathbf{x})|\mathbf{y}) = \mathbf{k}(\mathbf{x})^\top \mathbf{K}_N^{-1} \mathbf{y},$$
$$s_n^2(\mathbf{x}) = \mathbb{V}\mathrm{ar}(Y(\mathbf{x})|\mathbf{y}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}(\mathbf{x})^\top \mathbf{K}_N^{-1} \mathbf{k}(\mathbf{x}), \text{ where}$$

$\mathbf{k}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), \ldots, k(\mathbf{x}, \mathbf{x}_N))^\top$ and $\mathbf{K}_N = (k(\mathbf{x}_i, \mathbf{x}_j)_{1 \leq i,j \leq n}$.



GP training

GPs have their own hyperparameters, mostly for the kernel function.
Most popular kernels are stationary, e.g., the Gaussian kernel:
$k(x, x'|\tau^2, \theta) = \tau^2 \exp(-(x - x')^2/\theta) = \tau^2 c(\text{abs}(x - x')|\tau^2, \theta)$.

Hyperparameter estimation can be based on:

- model error (i.e., cross validation, training/testing sets)
- variogram analysis
- **likelihood**

Likelihood, i.e., multivariate normal density:

$$L = \frac{1}{(2\pi)^{N/2}|\mathbf{K}|^{1/2}} \exp\left(-\frac{1}{2}\mathbf{y}^\top \mathbf{K}^{-1}\mathbf{y}\right).$$

Alternatives include maximum-likelihood estimation and more Bayesian versions with various degrees of approximation.

Gaussian processes are increasingly popular

For reasons including:

- probabilistic model with spatial dependence
  $\rightarrow$ provide realistic uncertainty in sampled/unsampled areas;
- conditional distributions can be analytical (e.g., conditional variance)
  $\rightarrow$ key to define efficient active learning strategies;
- parameterized by two functions (mean and covariance kernel)
  $\rightarrow$ flexibility
- simple implementation (and many libraries)
  $\rightarrow$ easy to test
- theoretical background
  $\rightarrow$ Stochastic process, random fields, Reproducing Kernel Hilbert Spaces (RKHS), positive definite functions

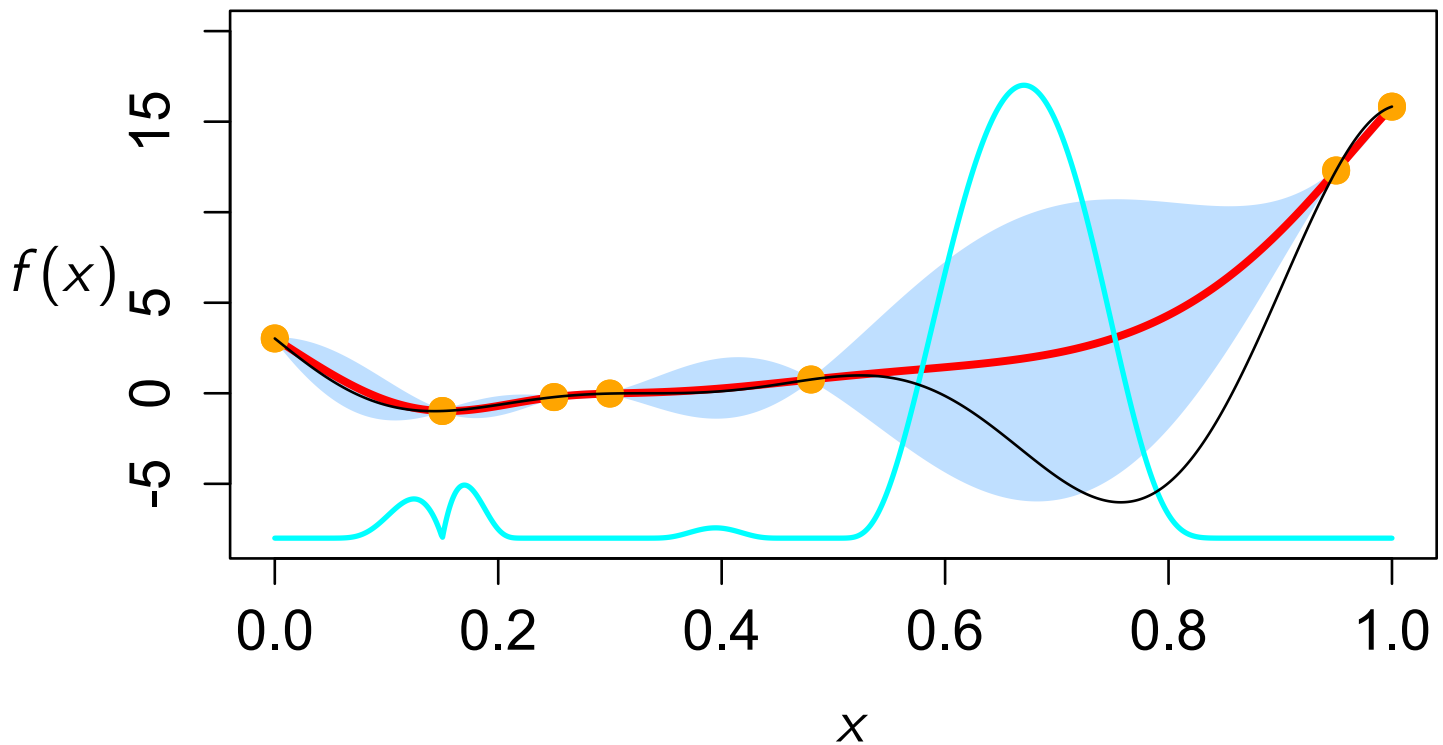
2) Infill criterion - Expected Improvement

Improvement: $I : \mathbf{x} \in \mathcal{X} \rightarrow \max\{f^* - Y(\mathbf{x}), 0\} \in \mathbb{R}$, $y^* = \min_{1 \leq i \leq N} y_i$,

Expected Improvement [Mockus et al., 1978]

$$E[I(\mathbf{x})|\mathbf{y}] = (f^* - m_N(\mathbf{x}))\,\Phi\left(\frac{f^* - m_N(\mathbf{x})}{s_N(\mathbf{x})}\right) + s_N(\mathbf{x})\phi\left(\frac{f^* - m_N(\mathbf{x})}{s_N(\mathbf{x})}\right)$$

$\rightarrow$ balance between exploration and exploitation



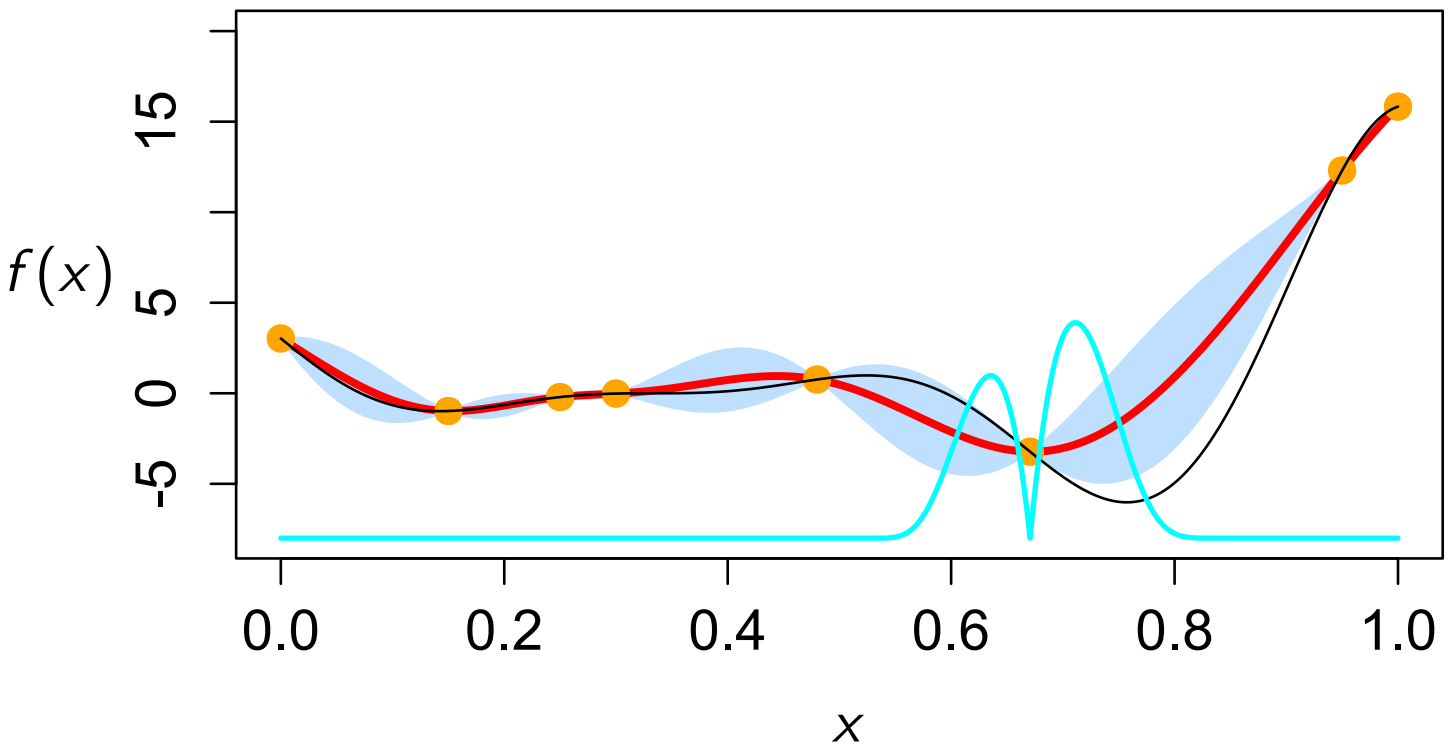
2) Infill criterion - Expected Improvement

Improvement: $I : \mathbf{x} \in \mathcal{X} \rightarrow \max\{f^* - Y(\mathbf{x}), 0\} \in \mathbb{R}$, $y^* = \min_{1 \leq i \leq N} y_i$,

Expected Improvement [Mockus et al., 1978]

$$E[I(\mathbf{x})|\mathbf{y}] = (f^* - m_N(\mathbf{x}))\,\Phi\left(\frac{f^* - m_N(\mathbf{x})}{s_N(\mathbf{x})}\right) + s_N(\mathbf{x})\phi\left(\frac{f^* - m_N(\mathbf{x})}{s_N(\mathbf{x})}\right)$$

$\rightarrow$ balance between exploration and exploitation



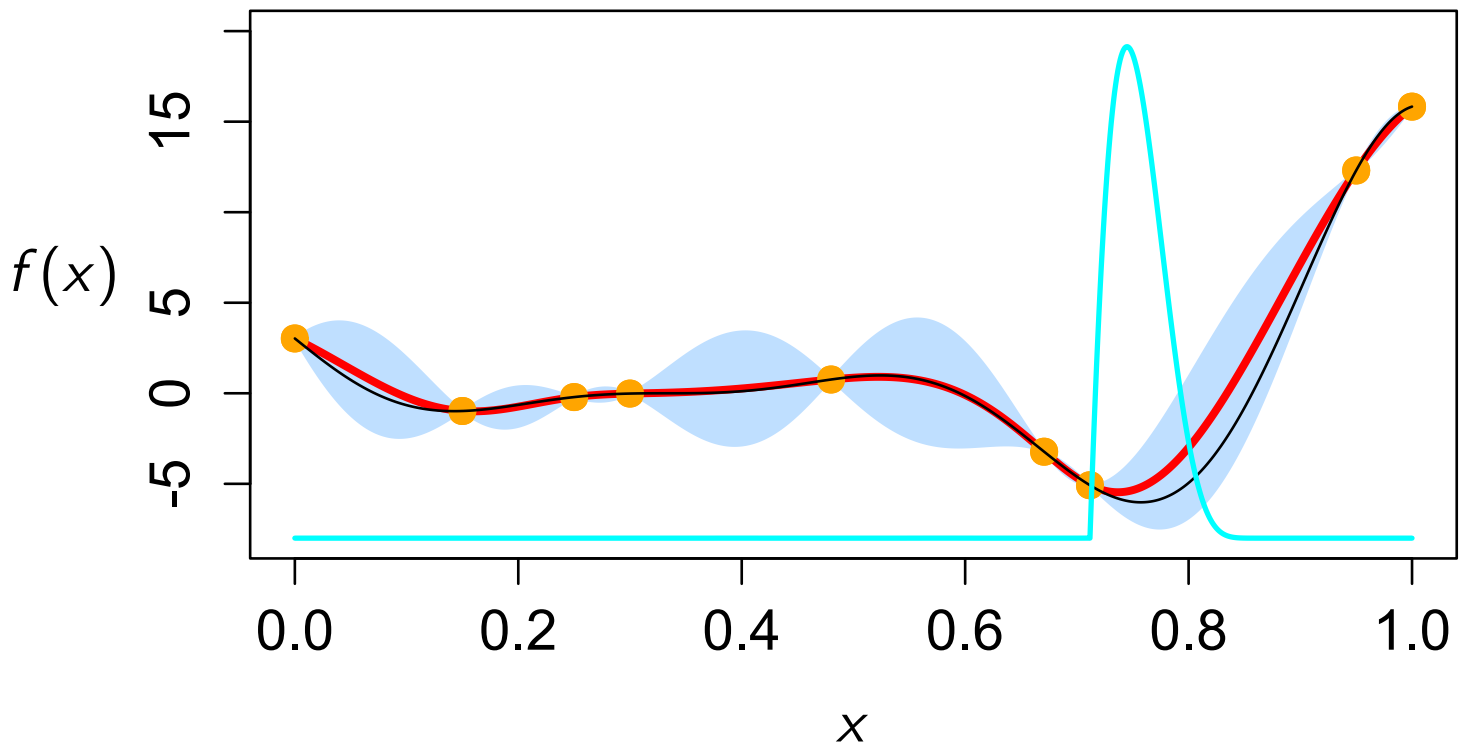
2) Infill criterion - Expected Improvement

Improvement: $I : \mathbf{x} \in \mathcal{X} \rightarrow \max\{f^* - Y(\mathbf{x}), 0\} \in \mathbb{R}$, $y^* = \min_{1 \leq i \leq N} y_i$,

Expected Improvement [Mockus et al., 1978]

$$E[I(\mathbf{x})|\mathbf{y}] = (f^* - m_N(\mathbf{x}))\,\Phi\left(\frac{f^* - m_N(\mathbf{x})}{s_N(\mathbf{x})}\right) + s_N(\mathbf{x})\phi\left(\frac{f^* - m_N(\mathbf{x})}{s_N(\mathbf{x})}\right)$$

$\rightarrow$ balance between exploration and exploitation



Beyond EI and implementations

Many different infill criteria:

- probability of improvement;
- UCB [Srinivas et al., 2009];
- entropy: conditional [Villemonteix et al., 2009], mutual information [Contal and Vayatis, 2013], PESC [Hernández-Lobato et al., 2014], max value [Wang and Jegelka, 2017];

Implementations:
`Matlab`: DACE, UQLab

`R`: `DiceKriging`, `DiceOptim`, `mlrMBO`, `tgp`, `laGP`, `hetGP`, ...

`Python`: GPy, GPflow, GPyTorch

Overview: [Erickson et al., 2017]

GP based sequential design is versatile

Active research directions on extensions include:

- batched versions of BO, e.g., with multi-point EI or local models
- look-ahead criteria
- noise on inputs/outputs, heteroskedasticity, non-Gaussian noise
- complex inputs/outputs (images, graphs, functions, ...)
- multi-fidelity and variable cost
- multi/many objective, multi-task, constrained optimization

with some practical limitations:

- GP training can be expensive: the vanilla version is $\mathcal{O}(N^3)$ in time complexity (but can be reduced to $\mathcal{O}(N)$ with approximations)
- d must remain in the low tens
- non-stationarity is harder to model

Plan

Gaussian process regression with noisy observations

Observation model: $y(\mathbf{x}_i) = f(\mathbf{x}_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, r(\mathbf{x}_i))$

For a zero mean GP with kernel k, MVN conditional identities give:

$$Y|\mathbf{y} \sim \mathcal{GP}(\mu, \sigma^2) \text{ with}$$

$$m_N(\mathbf{x}) = \mathbb{E}(Y(\mathbf{x})|\mathbf{y}) = \mathbf{k}(\mathbf{x})^\top (\mathbf{K}_N + \mathbf{\Sigma}_N)^{-1}\mathbf{y},$$
$$s_N^2(\mathbf{x}) = \mathbb{V}\mathrm{ar}(Y(\mathbf{x})|\mathbf{y}) = k(\mathbf{x},\mathbf{x}) - \mathbf{k}(\mathbf{x})^\top (\mathbf{K}_N + \mathbf{\Sigma}_N)^{-1}\mathbf{k}(\mathbf{x}) + r(\mathbf{x})$$

where $\mathbf{y} = (y(\mathbf{x}_i))_{1\le i\le N}^\top$, $\mathbf{k}(\mathbf{x}) = (k(\mathbf{x},\mathbf{x}_i))_{1\le i\le N}^\top$, $\mathbf{K}_N = (k(\mathbf{x}_i,\mathbf{x}_j))_{1\le i,j\le N}$, $\mathbf{\Sigma}_N = \mathrm{Diag}(r(\mathbf{x}_1),\ldots,r(\mathbf{x}_N))$

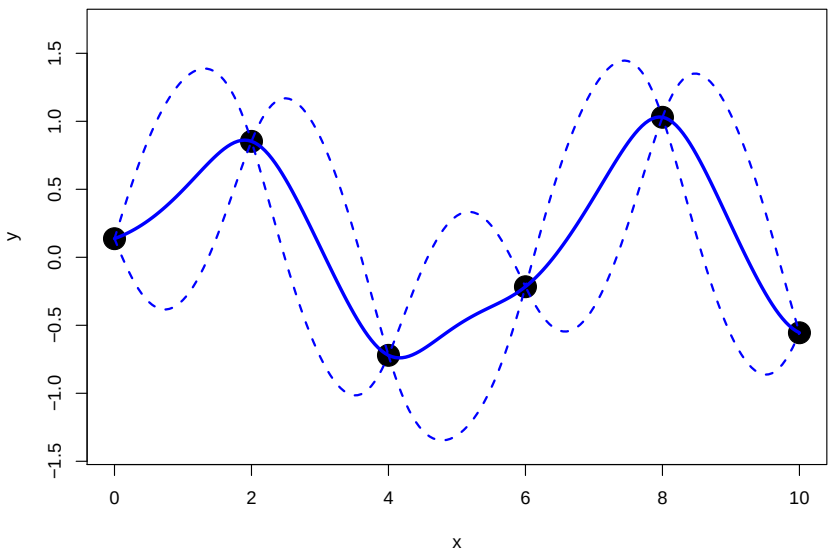
Remark 1: interest also in $P(y(\mathbf{x})|data)$, not only $P(f(\mathbf{x})|data)$
Remark 2: alternative noise distributions are possible, but losing analytical tractability

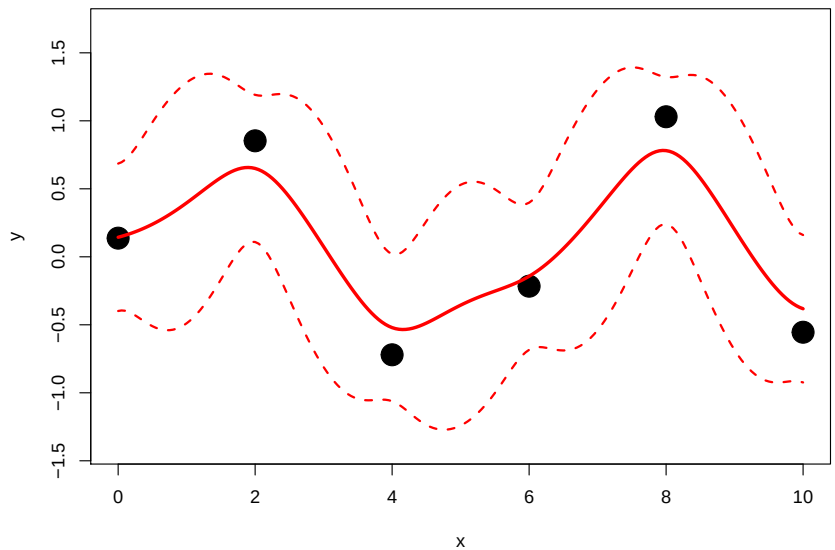
Gaussian process regression (2)

$$\boldsymbol{\Sigma}_N = \text{Diag}(\tau^2)$$

$$m_n(\mathbf{x}) = \mathbf{k}(\mathbf{x})^\top \mathbf{K}_N^{-1} \mathbf{y},$$
$$s_n^2(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}(\mathbf{x})^\top \mathbf{K}_N^{-1} \mathbf{k}(\mathbf{x})$$

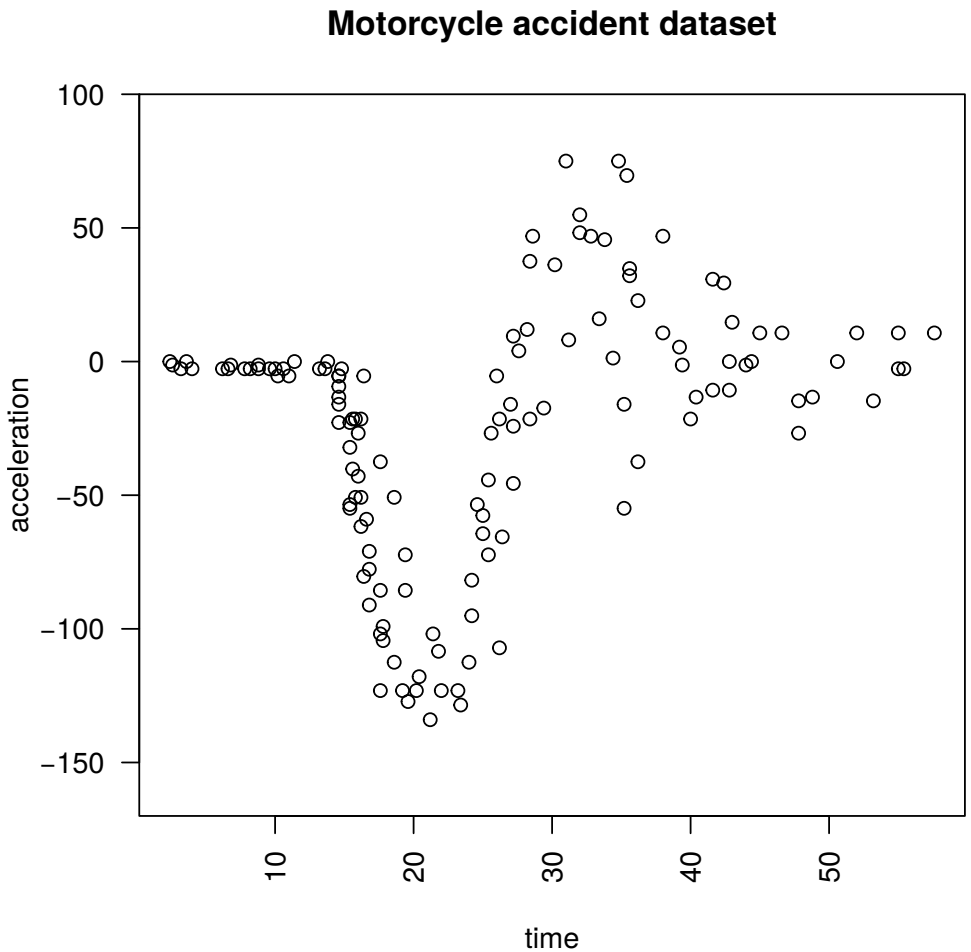
$$m_n(\mathbf{x}) = \mathbf{k}(\mathbf{x})^\top (\mathbf{K}_N + \boldsymbol{\Sigma}_N)^{-1} \mathbf{y},$$
$$s_n^2(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) + \tau^2 - \mathbf{k}(\mathbf{x})^\top (\mathbf{K}_N + \boldsymbol{\Sigma}_N)^{-1} \mathbf{k}(\mathbf{x})$$





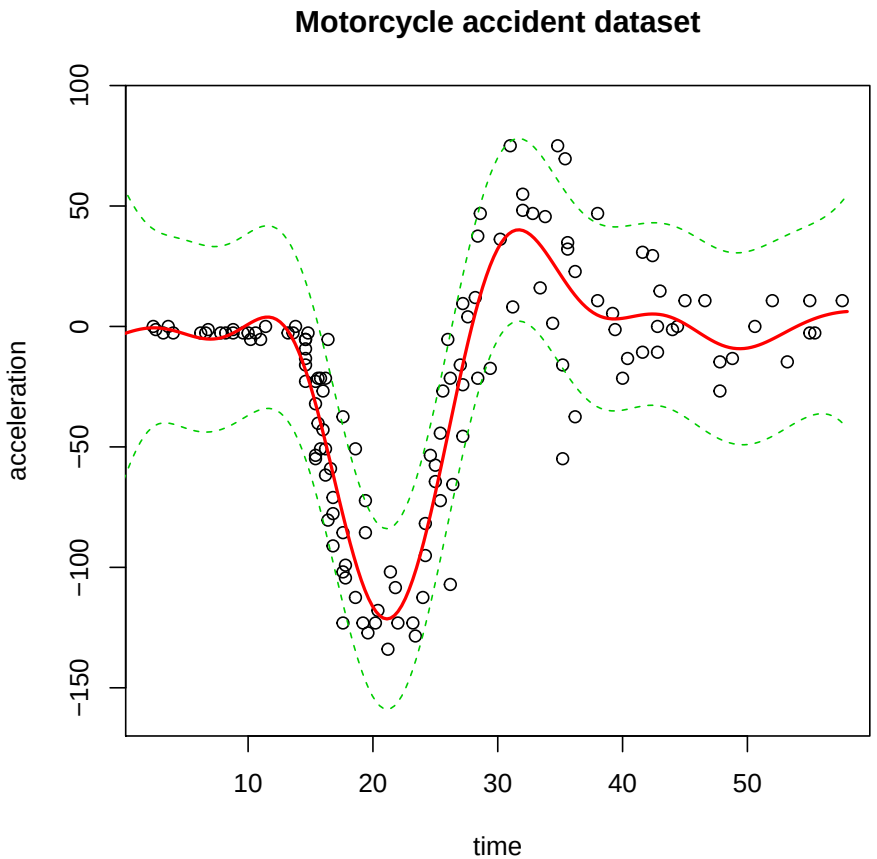
Motivating example for heteroskedasticity

Silverman (1985)'s motorcycle accident data



Motivating example for heteroskedasticity

Silverman (1985)'s motorcycle accident data

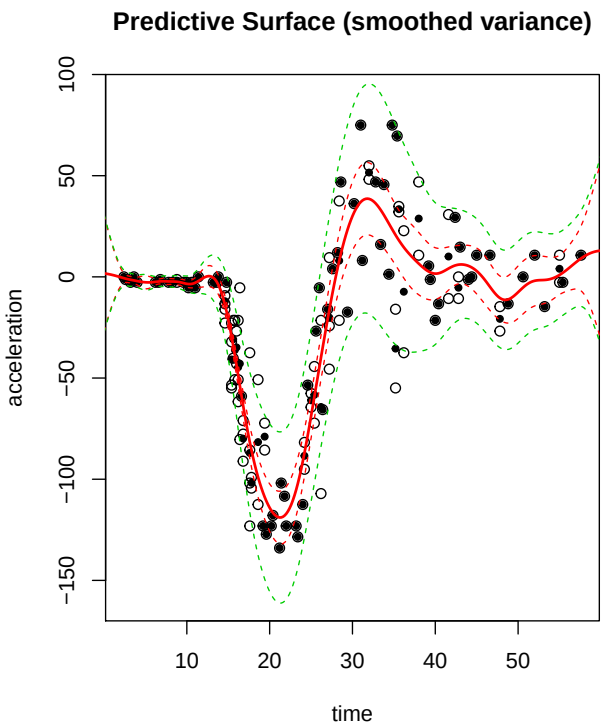


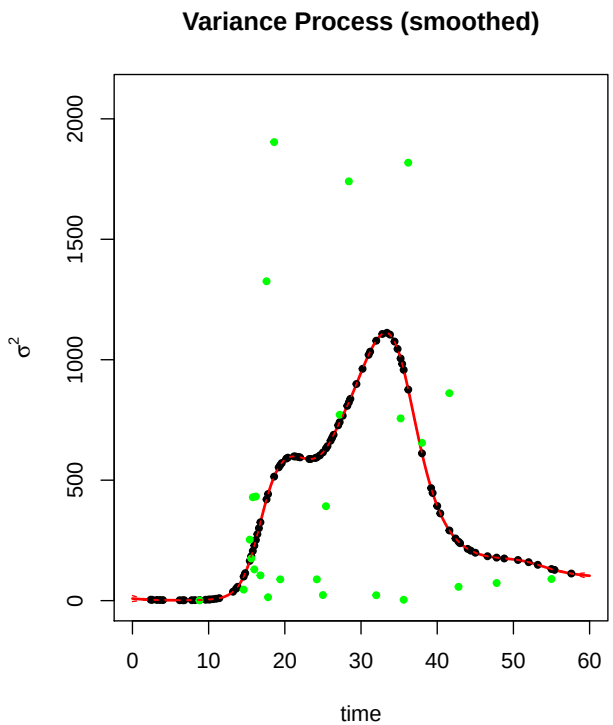
Gaussian process regression results with estimated constant noise:
$\rightarrow$ predictive mean is fine, but predictive variance is not.

Heteroskedastic GP modeling

To deal with input-dependent noise, one idea is to model jointly the (log-)variance by a second GP

- assumes smoothly varying noise across the input space,
- introduces latent variables (log-variances) or needs empirical $r(\mathbf{x}_i)$'s,
- full MCMC (Goldberg et al., 1998), hard-EM (Kersting et al., 2007), variational (Lazaro-Gredilla et al., 2011), MLE (Binois et al., 2018).



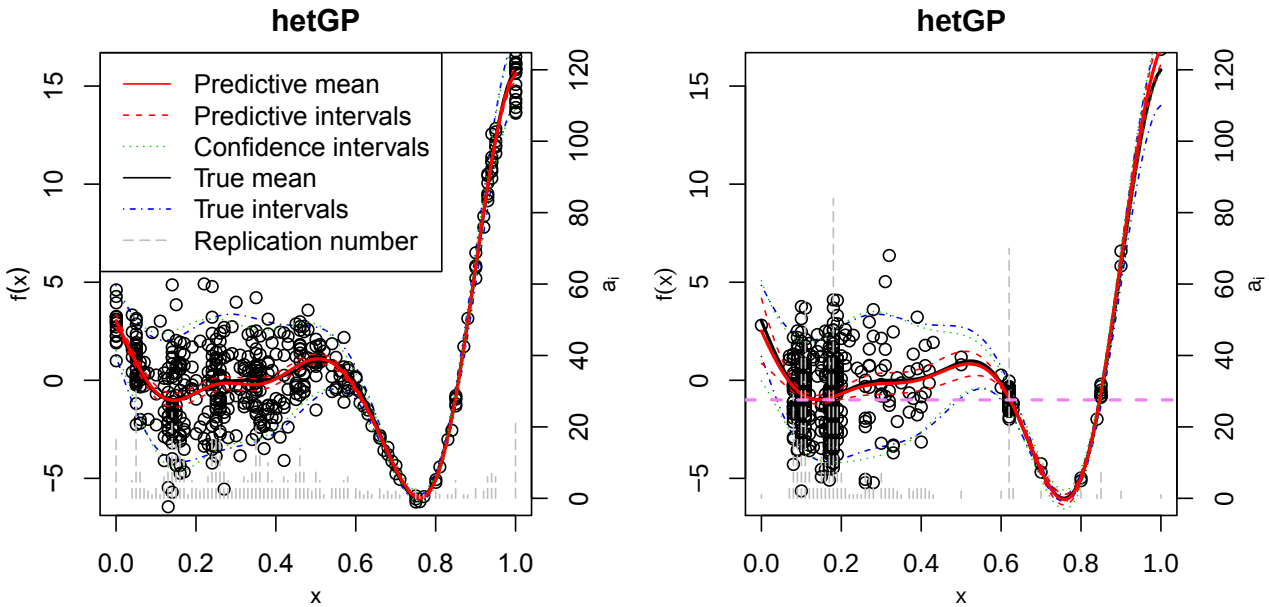


Heteroskedastic GP modeling (cont'd)

Joint modeling of input dependent noise $\varepsilon \sim \mathcal{N}(0, r(\mathbf{x}))$ allows:

- to add samples more efficiently,
- to benefit from – and plan – **replications** via looking-ahead

Examples



Left: targeting a globally accurate model
Right: critical area estimation $\{x \in D, f(x) \leq -1\}$

Example: Epidemic management (Hu et al., 2015)

Study disease outbreak dynamics based on stochastic compartmental modeling:

- Susceptible, Infected, Recovered (SIR) counts
- The continuous time state (S_t, I_t, R_t) is a Markov chain, with transition $S + I \to 2I$ and $I \to R$
- considered output is the total number of newly infected:

$$f(\mathbf{x}) := \mathbb{E}[S_0 - \lim_{T\to\infty} S_T | (S_0, I_0, R_0) = \mathbf{x}] = \gamma \mathbb{E}\Big[\int_0^\infty I_t \, dt | \mathbf{x}\Big]$$

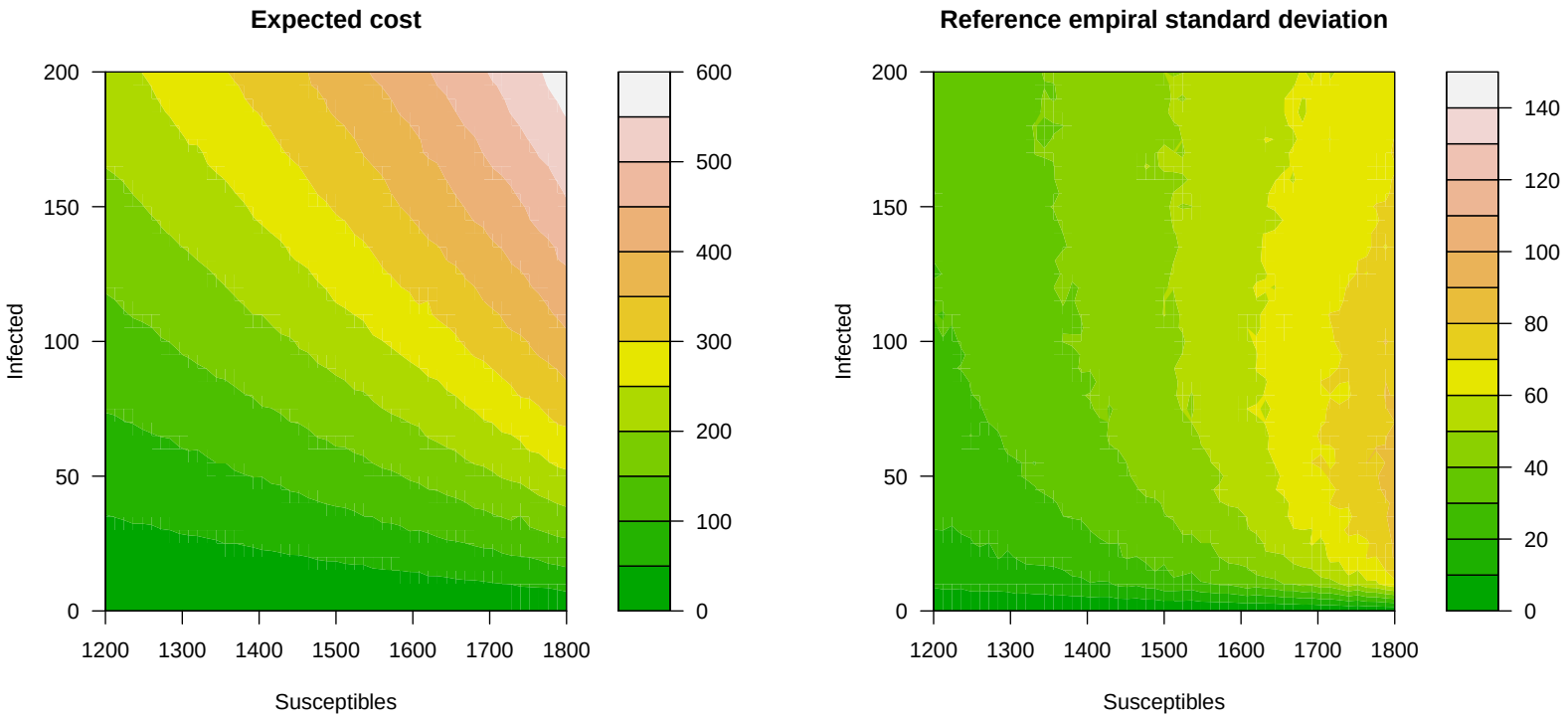
  estimated by Monte Carlo

Experiments:

- total population $M = 2000$
- testing set is 2000 designs on the grid, 100 replicates
- training set is 1000 designs, 500 with 5 replicates, 250 with 10, 150 with 50, 100 with 100

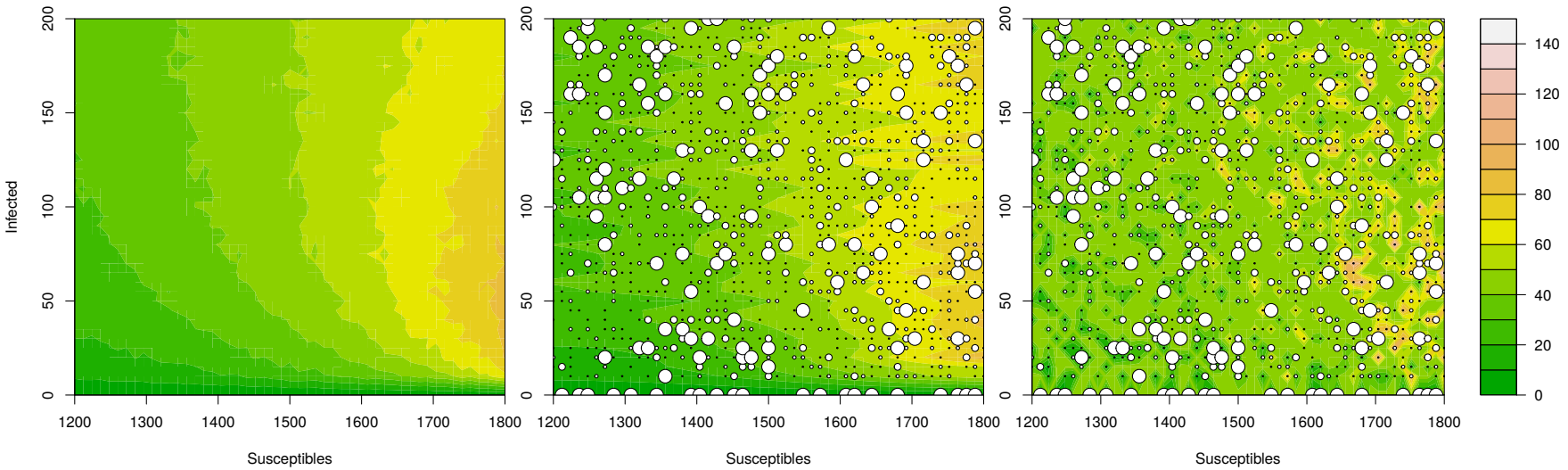
Example: Epidemic management (Hu et al., 2015)

Reference mean and noise surfaces



Example: Epidemic management (Hu et al., 2015)

Comparison of standard deviation estimations

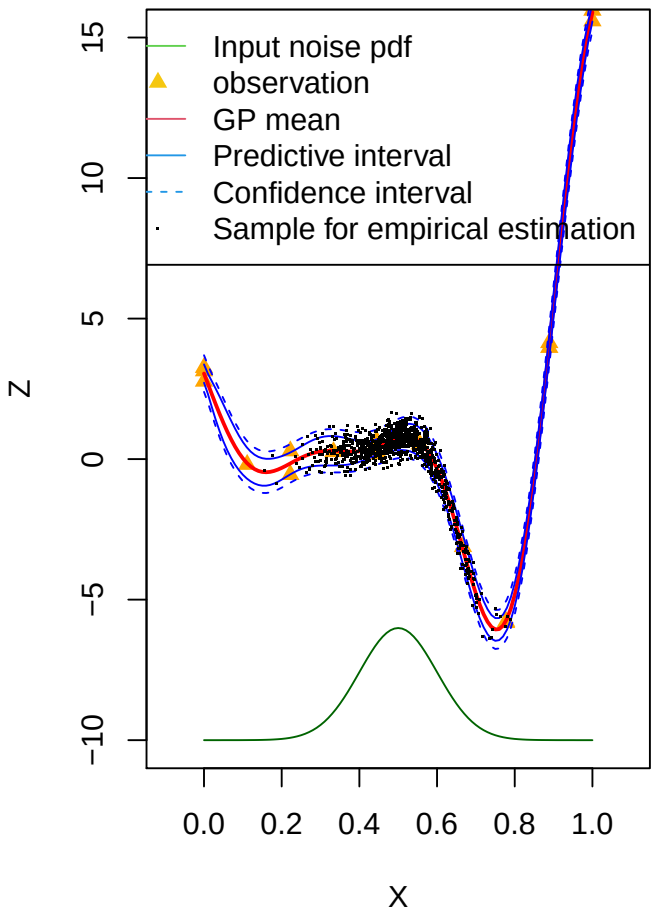


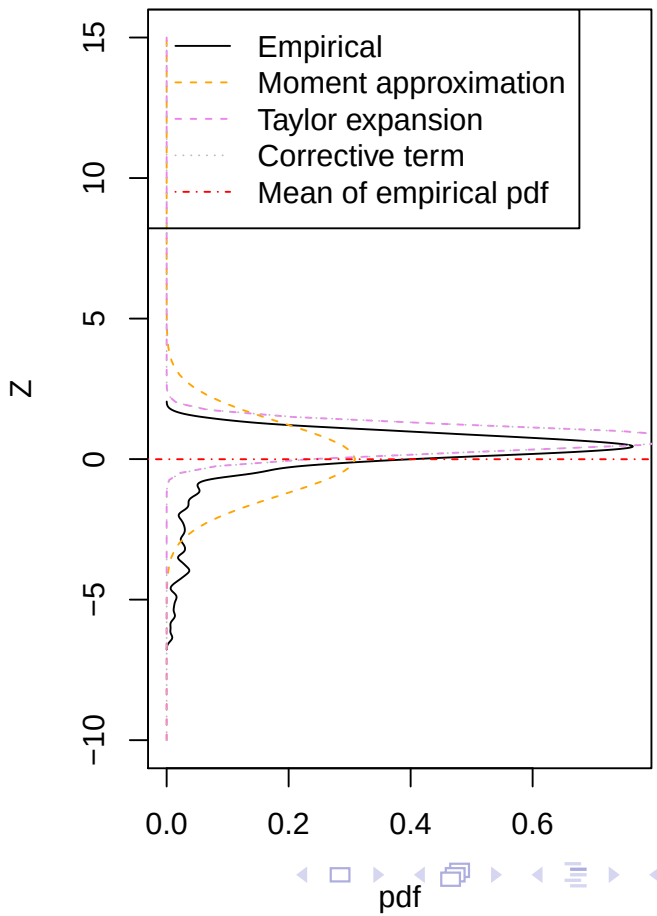
(a) Reference set (b) Joint estimation (c) Empirical

Dot size indicates number of replicates

Input Noise

In some applications (e.g., robustness), it may be necessary to predict at uncertain locations. The corresponding approximation can be derived, see e.g., [Girard, 2004].





Plan

Concepts in Multi-objective Optimization (MOO)

A solution minimizing every objective at once usually does not exist.

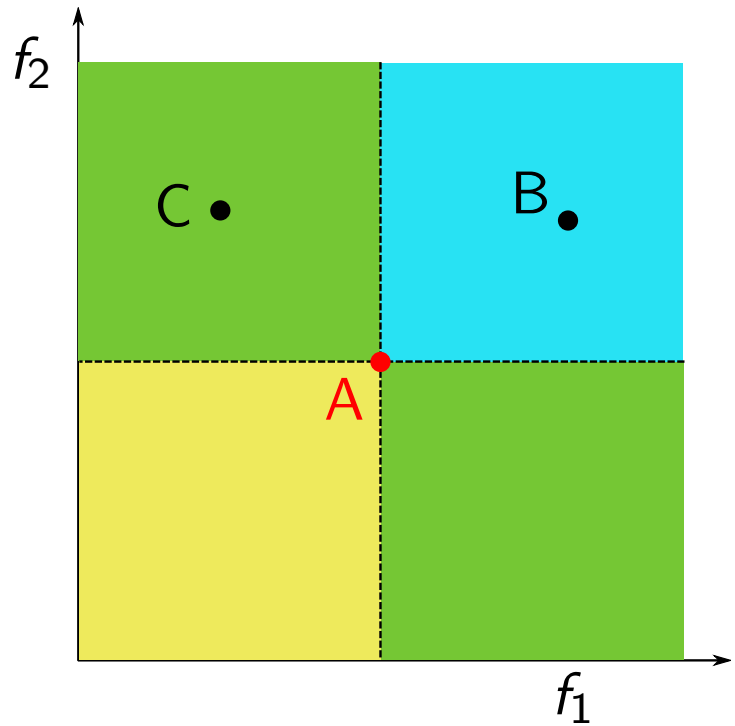
Pareto dominance

Vector A dominates vector B if:

- $\forall i \in \{1, \ldots, n\},\ a_i \leq b_i$
- $\exists j \in \{1, \ldots, n\}, a_j < b_j$

Pareto optimality

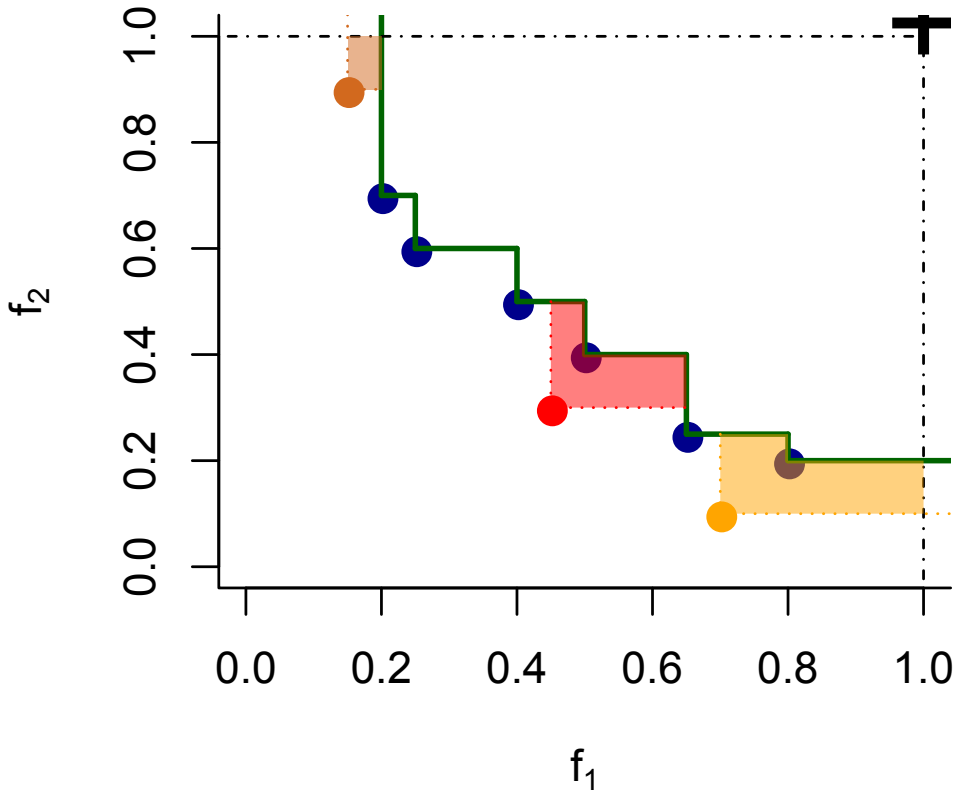
A is Pareto optimal if it is non-dominated.



- **Pareto set** (PS): set of all optimal points in the variable space
- **Pareto front** (PF): image of the Pareto set in the objective space
- Noisy case: PF defined on expected values of $f_1, \ldots, f_m$

2bis) MO infill criterion - Hypervolume Improvement

One possible MO improvement is the Hypervolume Improvement $I_{\mathcal{H}}$, i.e., the volume added to the current Pareto front by a new observation.



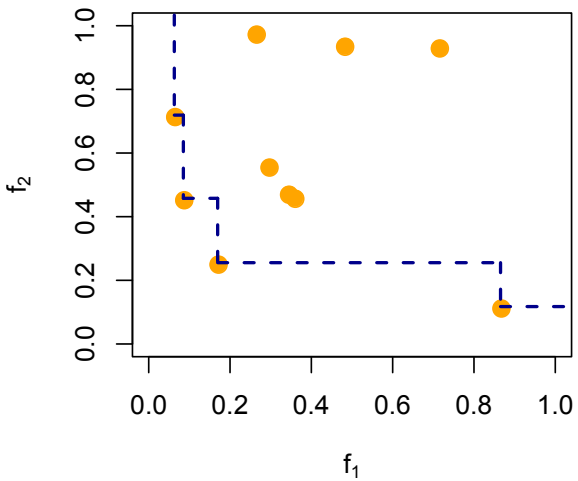
The corresponding generalization of EI is the Expected Hypervolume Improvement [Emmerich et al., 2011]:
$EHI(\mathbf{x}) = \mathbb{E}(I_{\mathcal{H}}(Y_1(\mathbf{x}), \ldots, Y_m(\mathbf{x}))|\mathbf{Y})$.

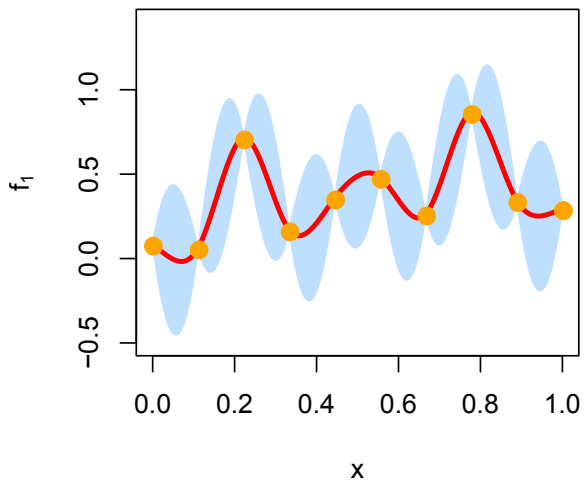
Estimating the Pareto front

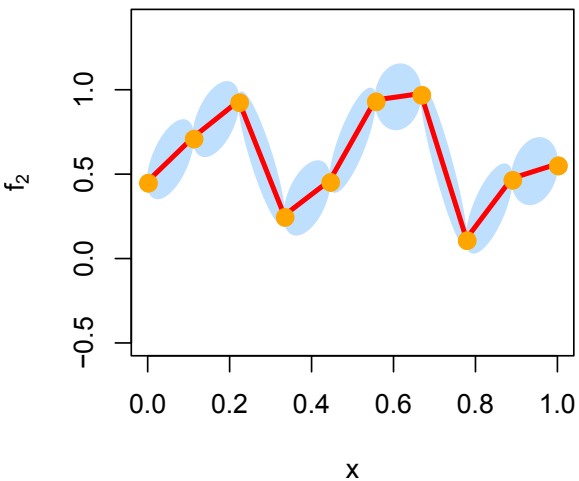
Here, we consider n observations of m functions $f_i, 1 \leq i, \leq m$.

We build m independent GP models $Y_i \sim GP(m_n, k_n)$.

Aim: giving an estimation of the whole Pareto front





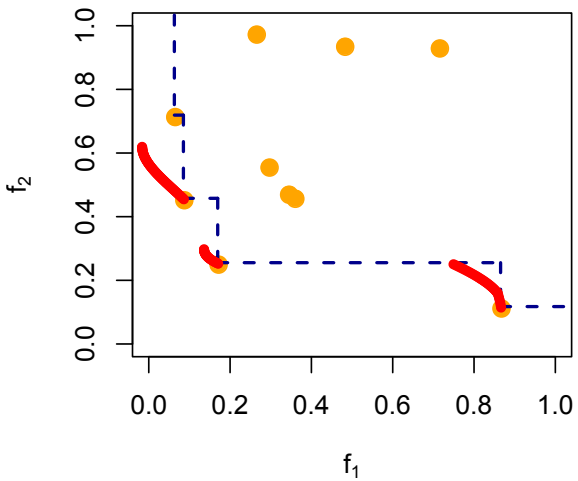


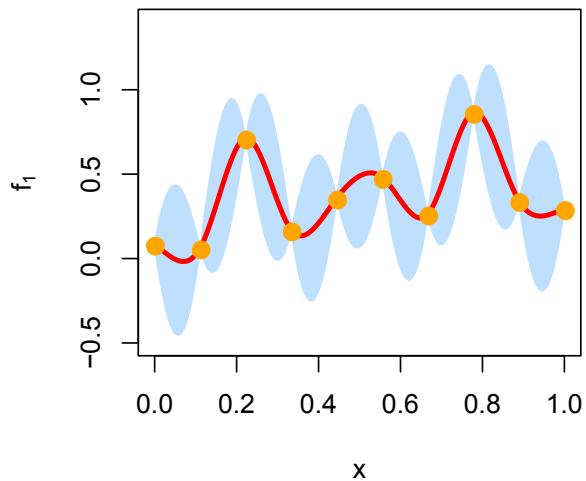
Estimating the Pareto front

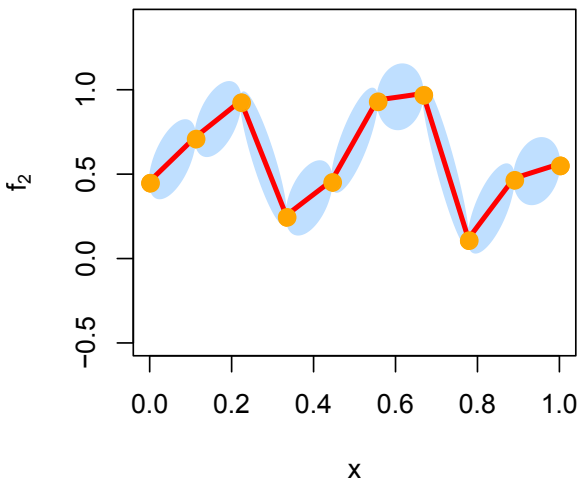
Here, we consider n observations of m functions $f_i, 1 \leq i, \leq m$.

We build m independent GP models $Y_i \sim GP(m_n, k_n)$.

Aim: giving an estimation of the whole Pareto front







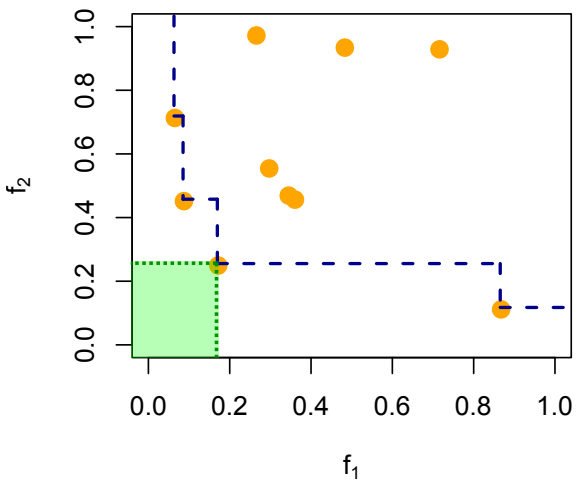
Naive solution: taking the Pareto front of the GP models' mean
$\rightarrow$ does not propagate model uncertainty.

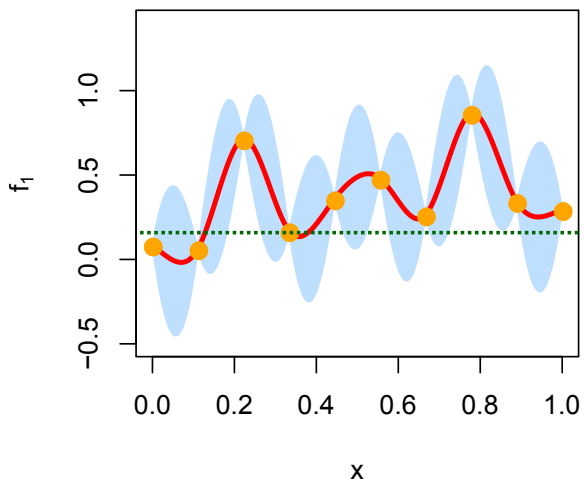
Attainment probability

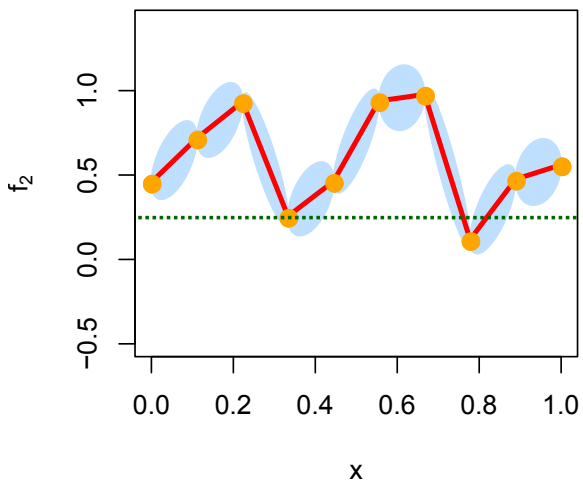
We consider the random attained set:
$\mathcal{Y} = \{\mathbf{y} \in \mathbb{R}^m \mid \exists \mathbf{x} \in \mathcal{X} \text{ s.t. } Y_1(\mathbf{x}) \leq y_1 \cap \cdots \cap Y_m(\mathbf{x}) \leq y_m\}$

In the rest, we make use of the following attainment probability:

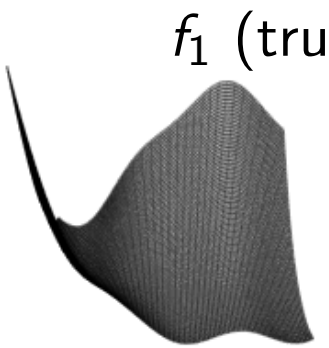
$$p_{\mathcal{Y}} : \mathbb{R}^m \to [0,1], \mathbf{y} \to \mathbb{P}[\mathbf{y} \in \mathcal{Y}]$$

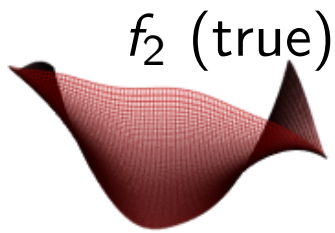


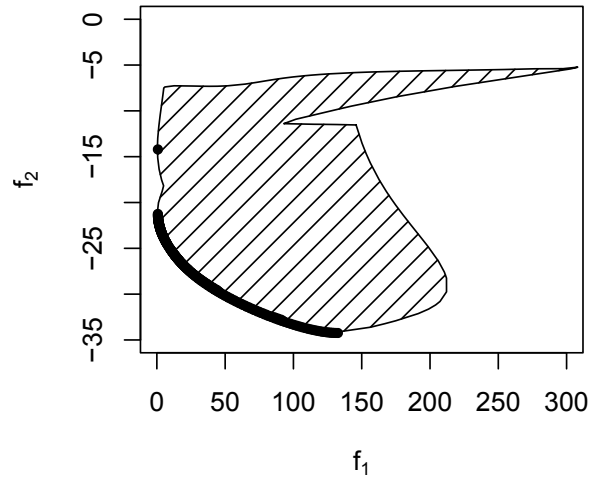




Conditional Pareto Front (CPF) simulations

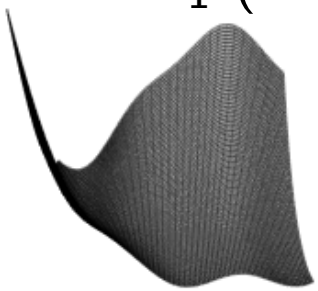




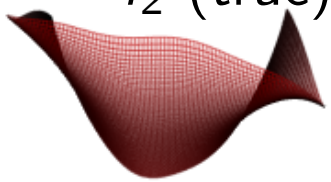


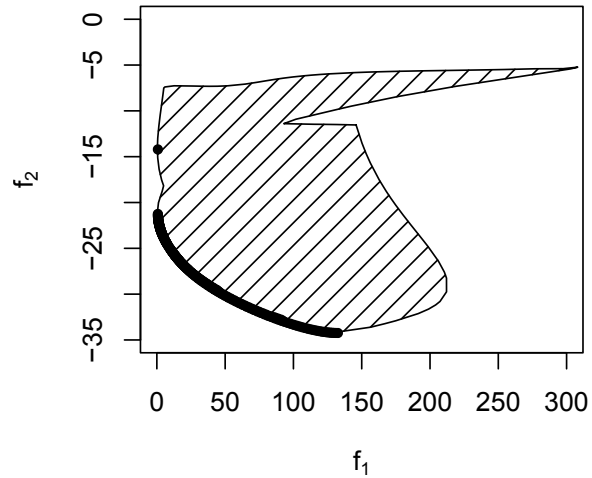
Conditional Pareto Front (CPF) simulations

f_1 (true)

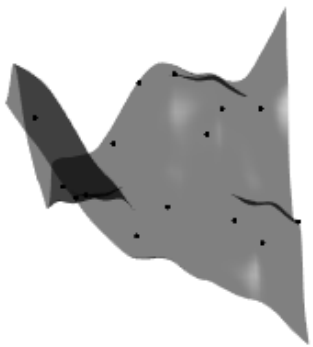


f_2 (true)





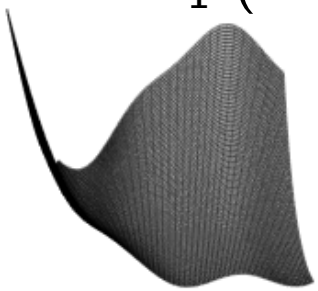
Y_1 conditional simulation



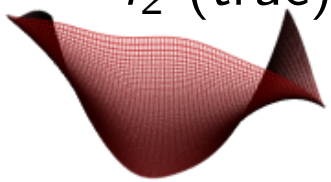
Y_2 conditional simulation

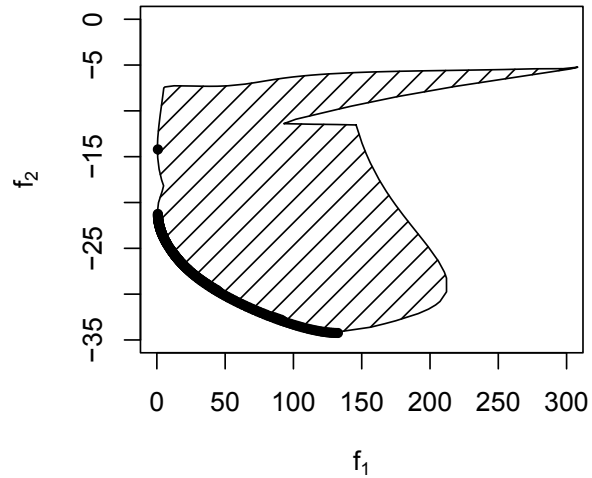
Conditional Pareto Front (CPF) simulations

f_1 (true)

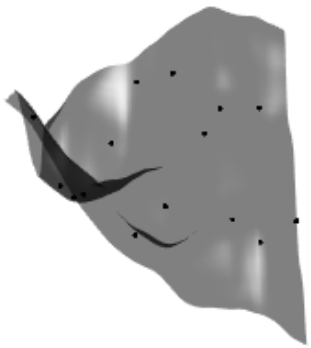


f_2 (true)





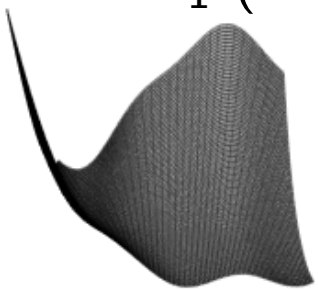
Y_1 conditional simulation



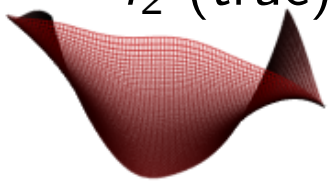
Y_2 conditional simulation

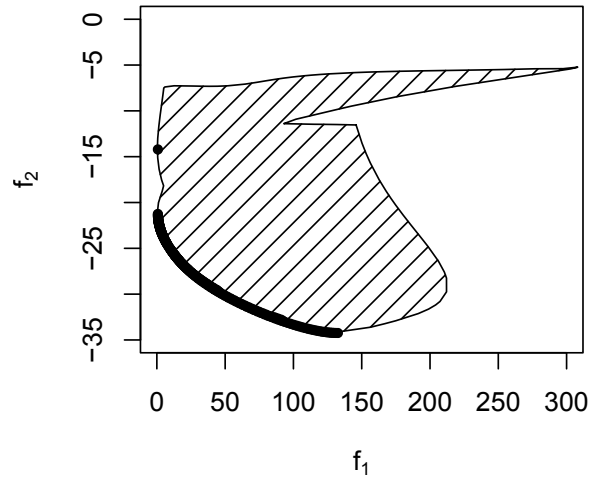
Conditional Pareto Front (CPF) simulations

f_1 (true)

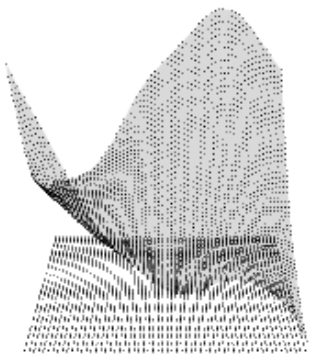


f_2 (true)

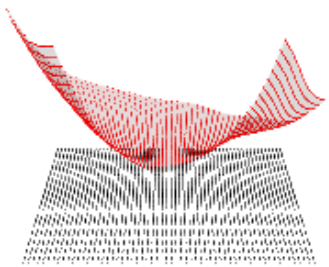




Y_1 conditional simulation

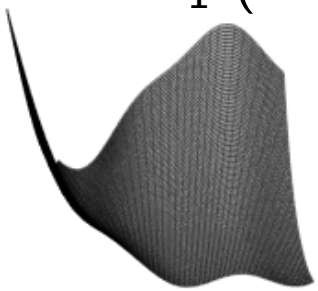


Y_2 conditional simulation

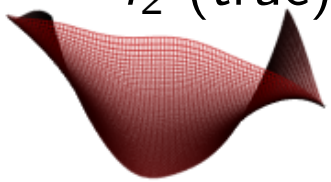


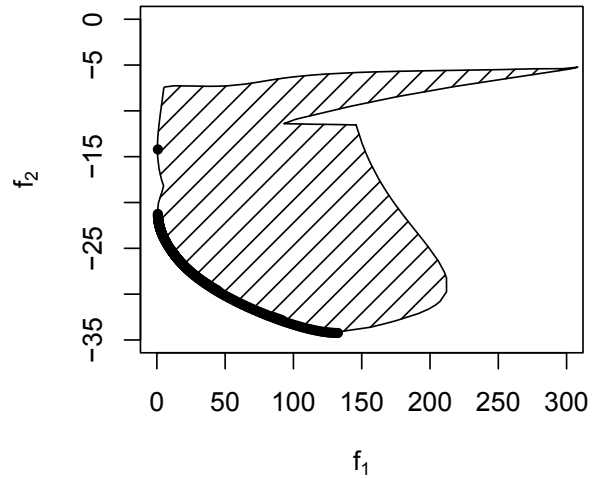
Conditional Pareto Front (CPF) simulations

f_1 (true)

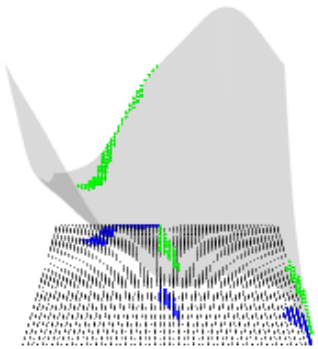


f_2 (true)

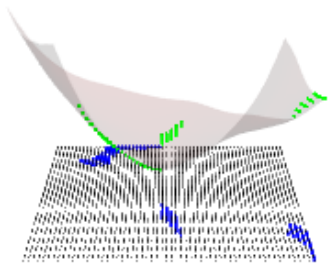




Y_1 conditional simulation



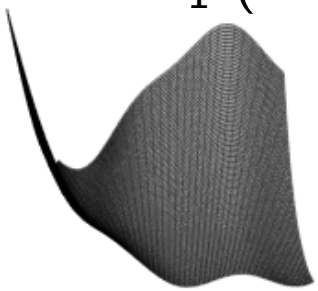
Y_2 conditional simulation



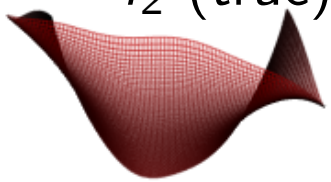
CPF: non-dominated points of a pair of conditional simulations

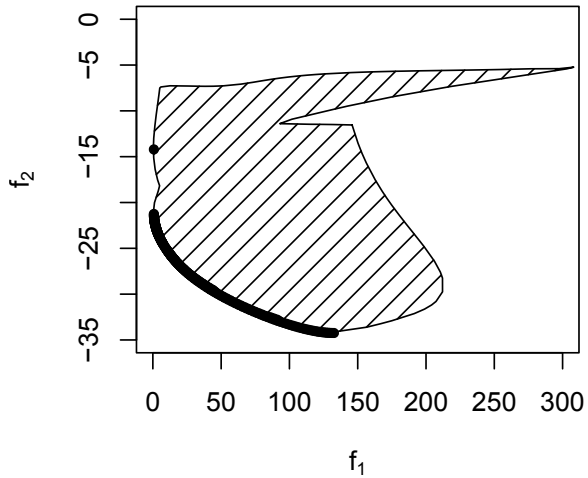
Conditional Pareto Front (CPF) simulations

f_1 (true)

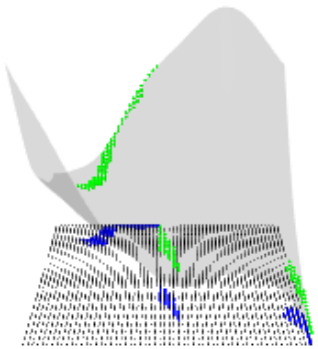


f_2 (true)

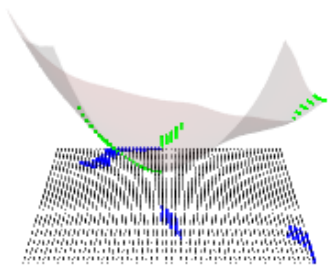


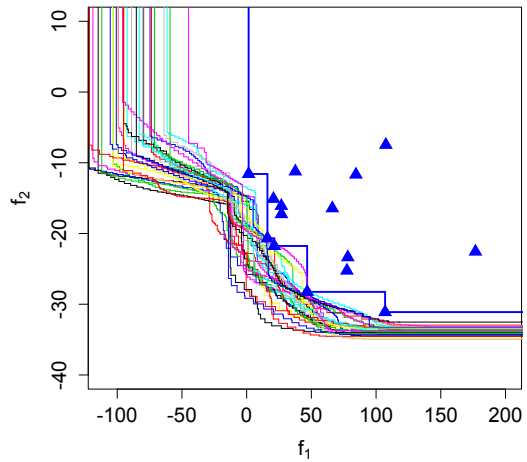


Y_1 conditional simulation



Y_2 conditional simulation

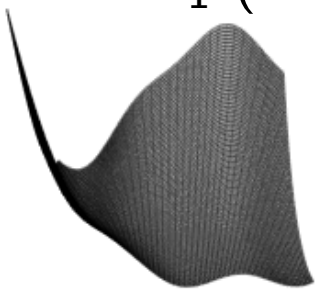




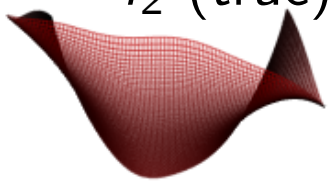
CPF: non-dominated points of a pair of conditional simulations

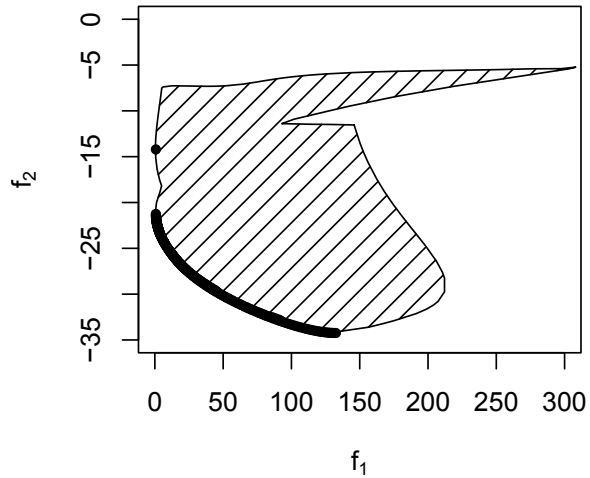
Conditional Pareto Front (CPF) simulations

f_1 (true)

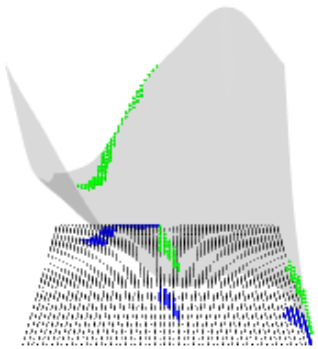


f_2 (true)

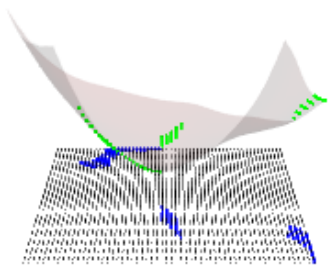


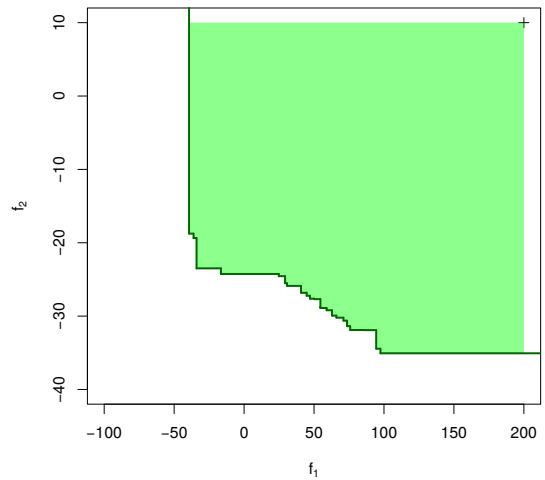


Y_1 conditional simulation



Y_2 conditional simulation

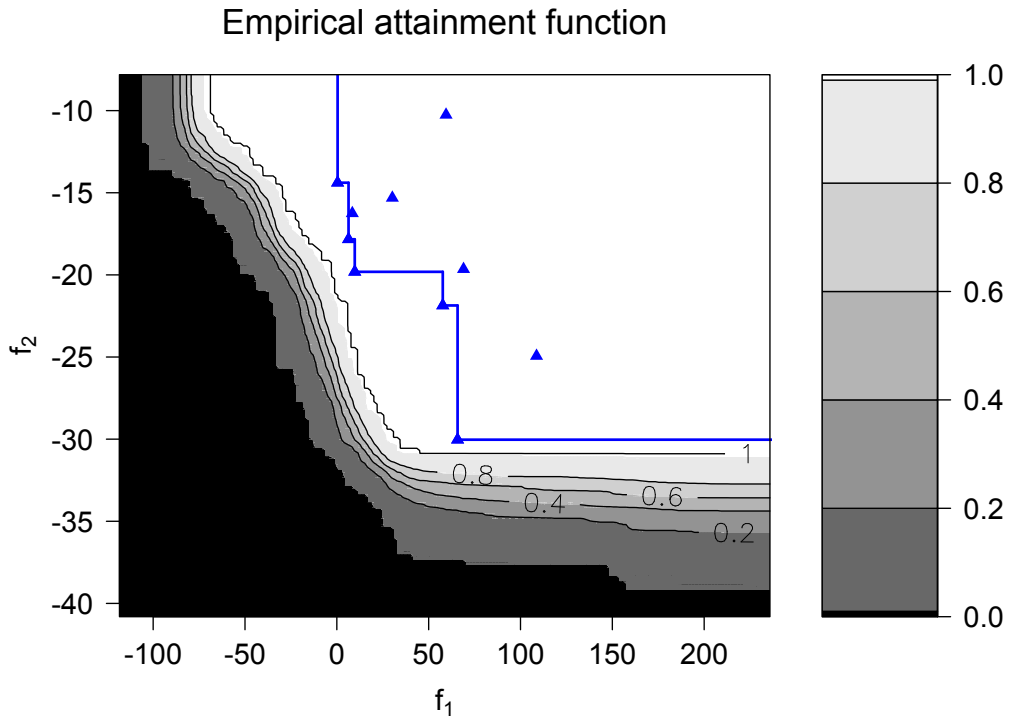




CPF: non-dominated points of a pair of conditional simulations

Empirical attainment function

$p_{\mathcal{Y}}$ is estimated as follows : $\hat{\alpha}_N(z) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{I}\{CPF_i \text{ dominates } z\}$



β-quantiles are level sets of the attainment/coverage function of $\mathcal{Y}$, $p_{\mathcal{Y}}$:

$$\mathcal{Q}_\beta = \{z \in \mathbb{R}^m, p_{\mathcal{Y}}(z) \geq \beta\}$$

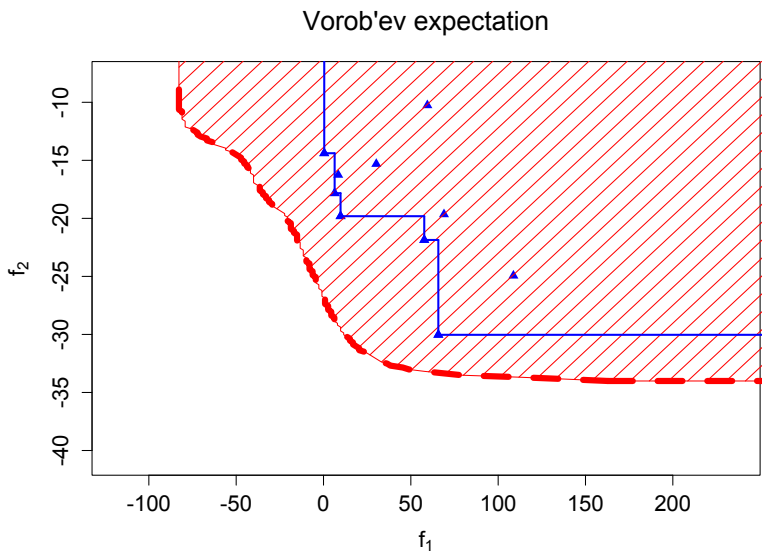
Vorob'ev expectation and deviation

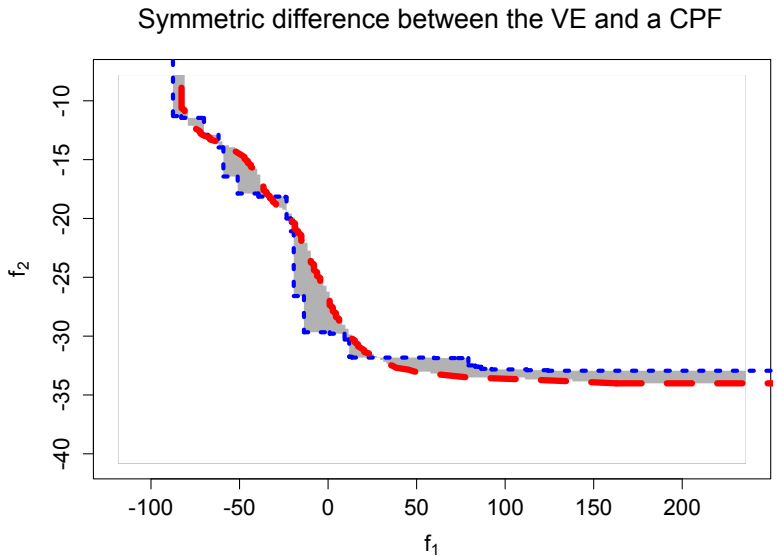
Vorob'ev expectation (VE) [Molchanov, 2005, Chevalier et al., 2013]

Assuming that $\mathbb{E}(\mu(\mathcal{Y})) < \infty$, it is defined as the smallest β^*-quantile such that $\mathbb{E}(\mu(\mathcal{Y})) = \mu(\mathcal{Q}_{\beta^*})$ where μ is the Lebesgue measure.

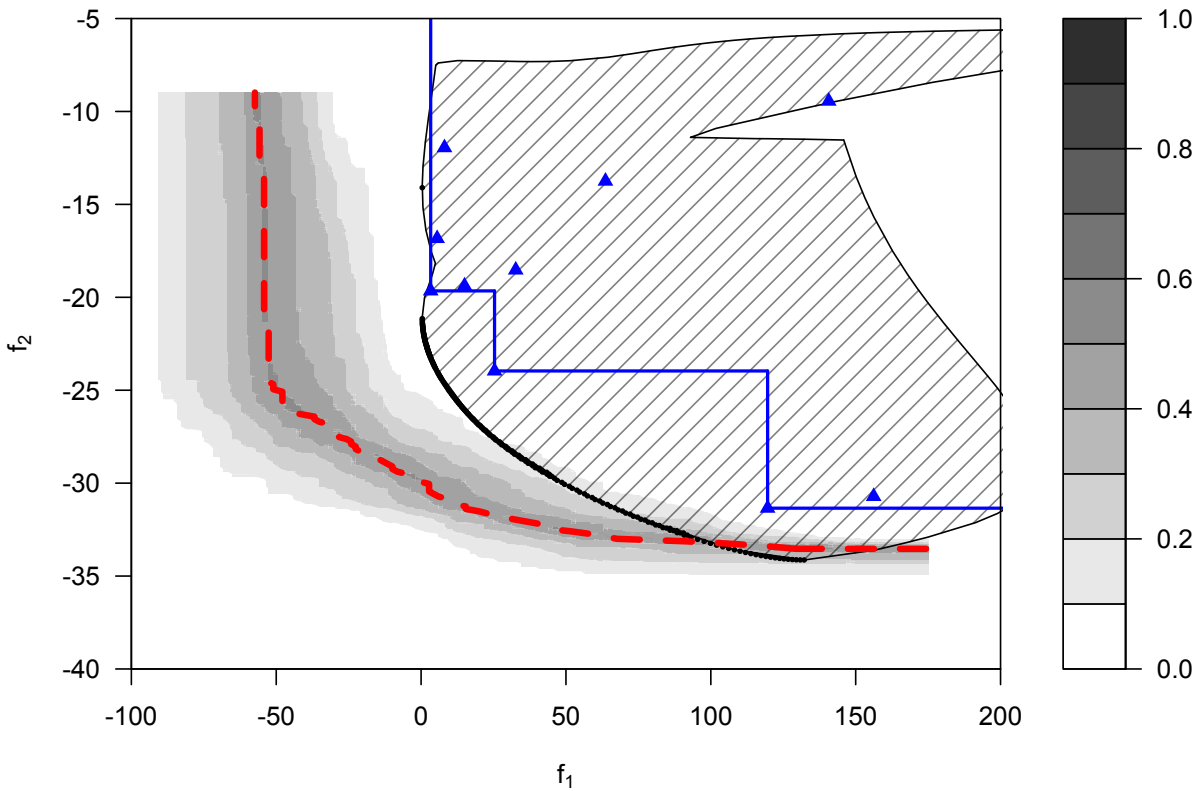
Associated variance: Vorob'ev deviation $\mathbb{E}(\mu(\mathcal{Q}_{\beta^*} \Delta \mathcal{Y}))$ (Δ: symmetric difference between sets)

Example: Application to CPFs, using a reference point to bound volumes



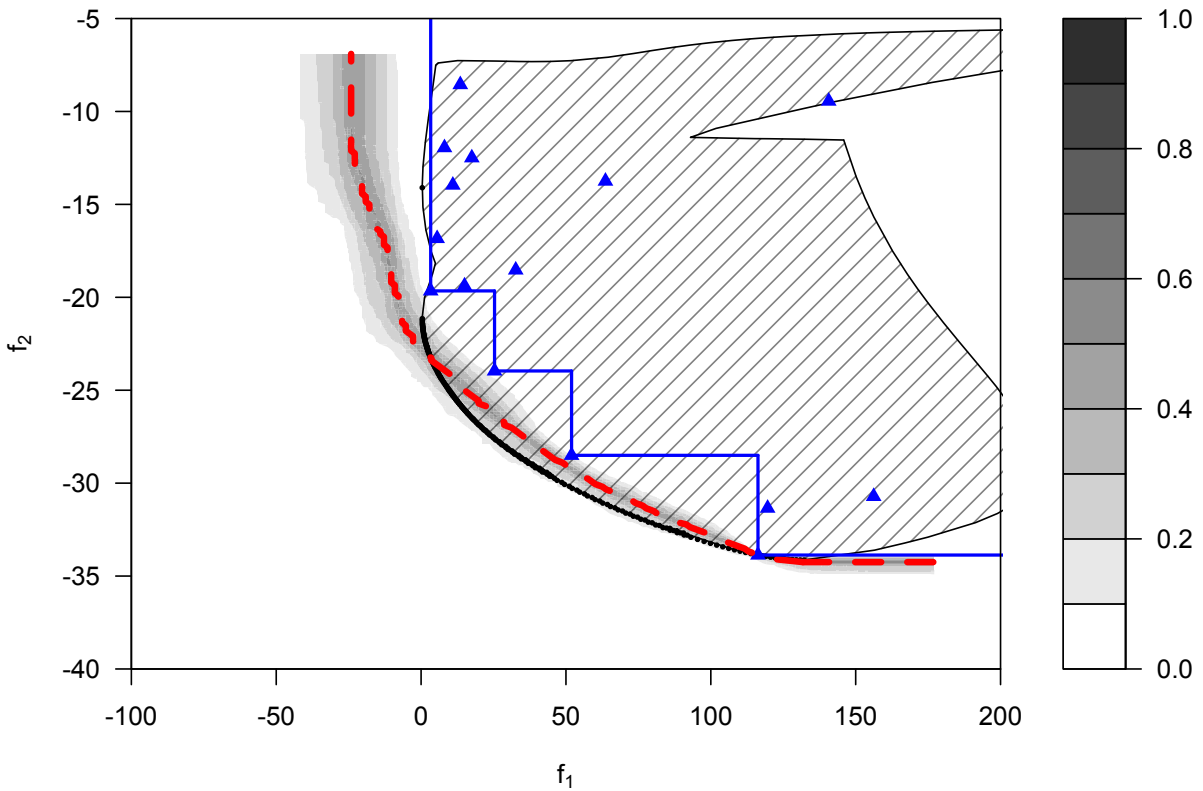


Example 1



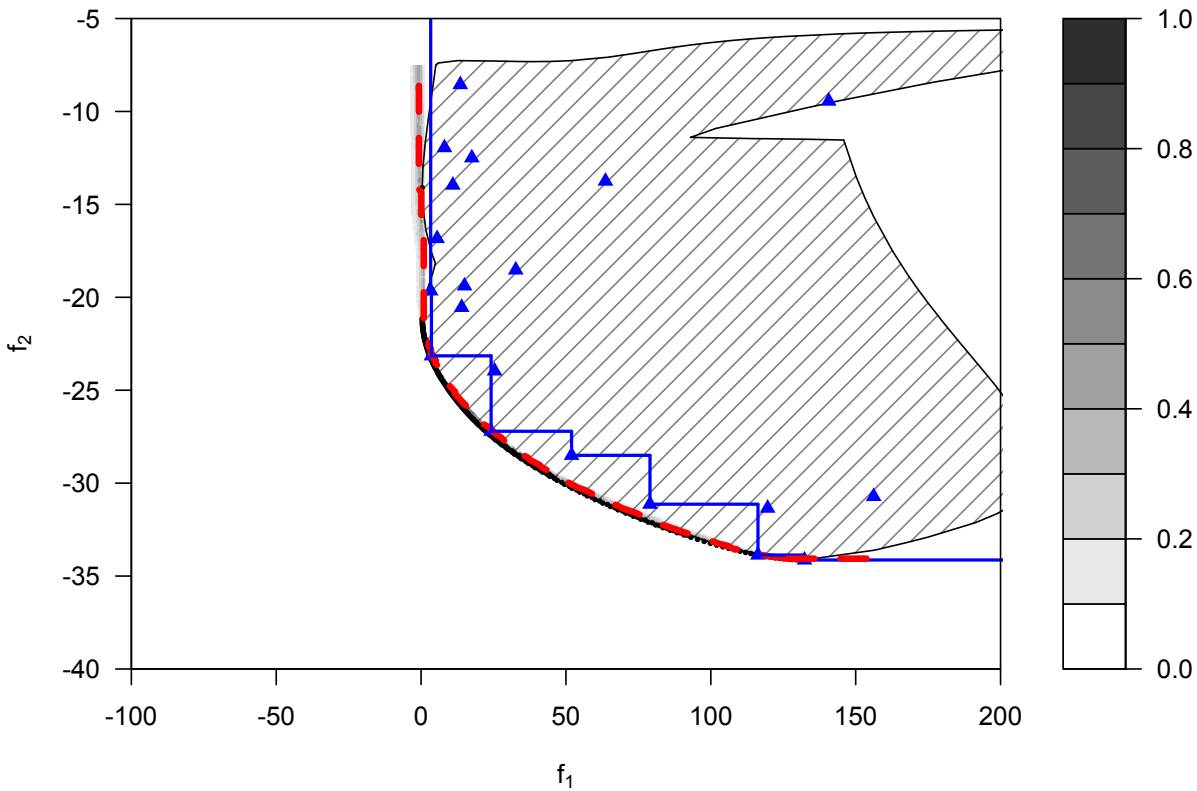
$n = 10$

Example 1



$n = 15$

Example 1



$n = 20$

Stepwise Uncertainty Reduction: definition of the criterion

The Vorob'ev deviation is a measure of the uncertainty on $\mathcal{P}$.
Then it is a possible infill criterion to minimize for a new candidate observation $\mathbf{x}_{n+1}$: $J(\mathbf{x}_{n+1}) = \mathbb{E}\left(\mu(\mathcal{Q}_{\beta^*_{n+1}}\Delta\mathcal{Y})|\mathbf{Y}\right)$.

Stepwise Uncertainty Reduction: definition of the criterion

The Vorob'ev deviation is a measure of the uncertainty on $\mathcal{P}$.
Then it is a possible infill criterion to minimize for a new candidate observation $\mathbf{x}_{n+1}$: $J(\mathbf{x}_{n+1}) = \mathbb{E}\left(\mu(\mathcal{Q}_{\beta^*_{n+1}}\Delta\mathcal{Y})|\mathbf{Y}\right)$.
This implies updating conditional simulations [Chevalier et al., 2014].

Stepwise Uncertainty Reduction: definition of the criterion

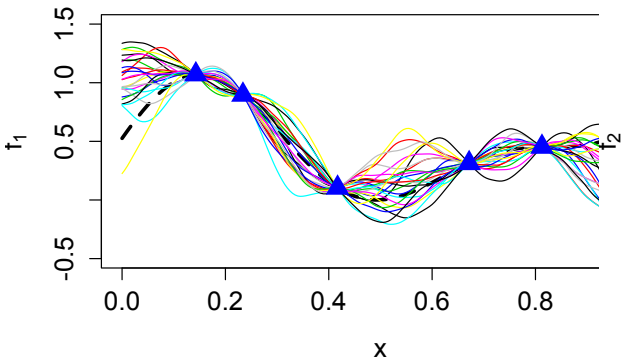
The Vorob'ev deviation is a measure of the uncertainty on $\mathcal{P}$.
Then it is a possible infill criterion to minimize for a new candidate observation $\mathbf{x}_{n+1}$: $J(\mathbf{x}_{n+1}) = \mathbb{E}\left(\mu(\mathcal{Q}_{\beta^*_{n+1}}\Delta\mathcal{Y})|\mathbf{Y}\right)$.
This implies updating conditional simulations [Chevalier et al., 2014].

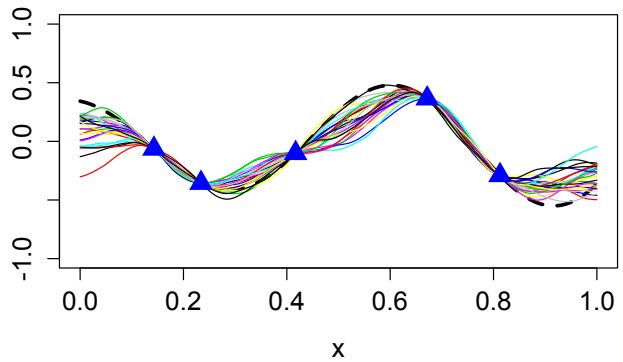
Algorithm Estimation of $J(\mathbf{x}_{n+1})$

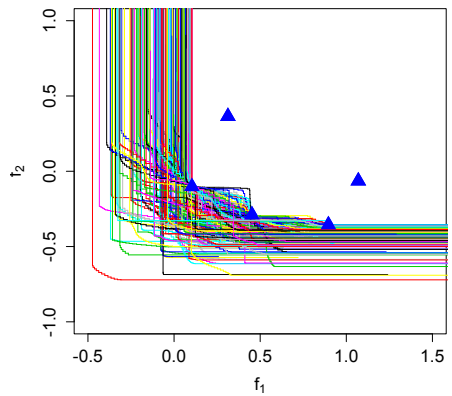
1: **Require** : p conditional simulations knowing $\mathbf{Y}$
2: **for** $i \in (1, \ldots, q)$ **do**
3: Sample $\mathbf{z} \sim \mathcal{N}(m_n(\mathbf{x}_{n+1}), s_n(\mathbf{x}_{n+1}))$
4: $Y^{(i)} \leftarrow$ Update the p conditional simulations with $\mathbf{z}$
5: **end for**
6: Estimate the uncertainty based on the q ensembles of p conditional simulations, $Y^{(i)}, 1 \leq i \leq q$

Update of simulations - illustration on CPFs

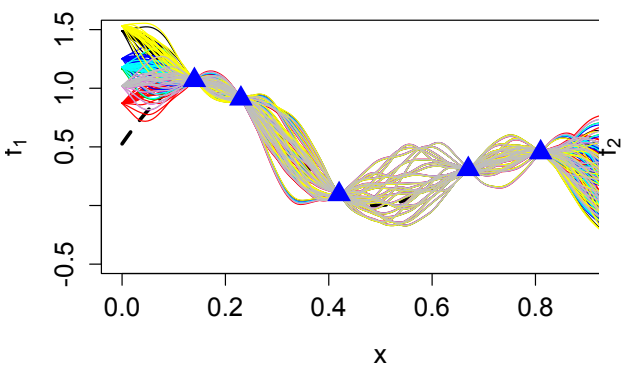
Simulations with n observations:

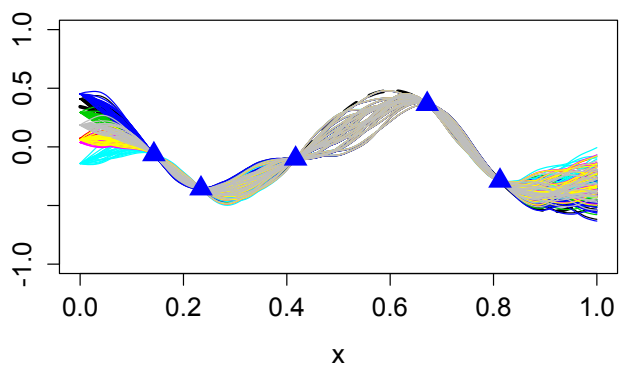


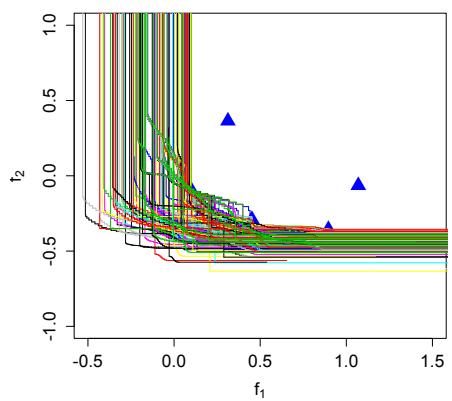




Update at $x_{n+1} = 0$

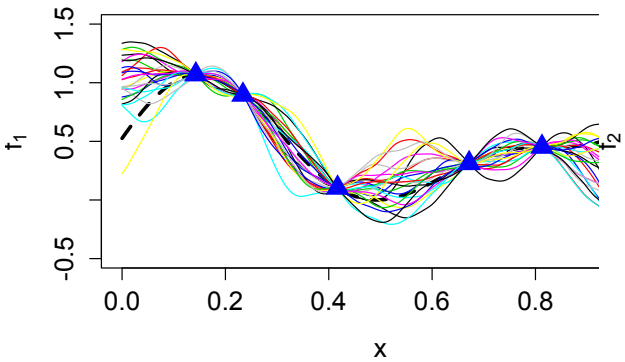


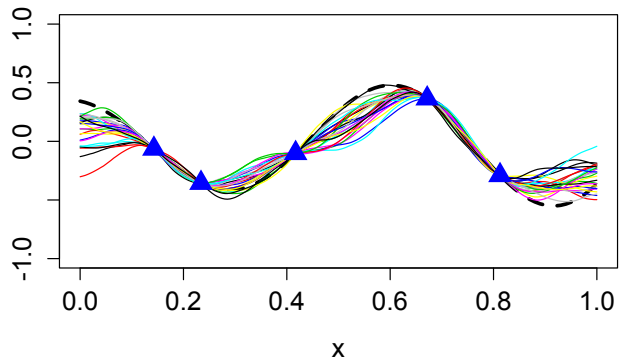


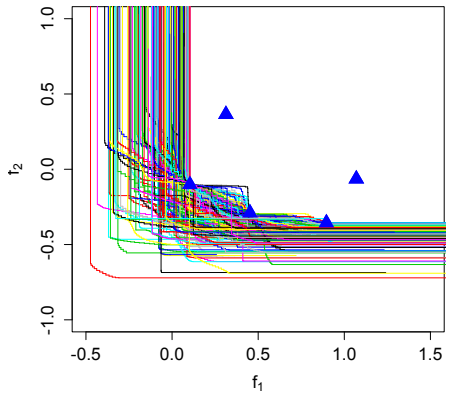


Update of simulations - illustration on CPFs

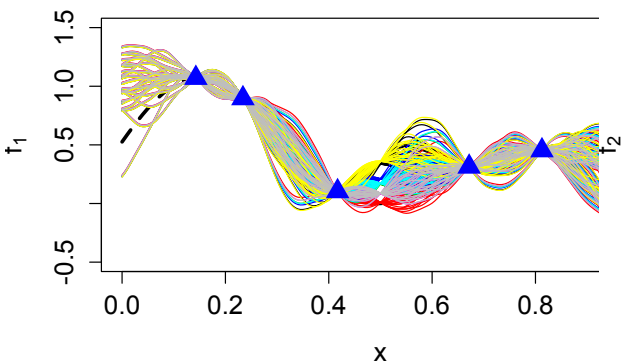
Simulations with n observations:

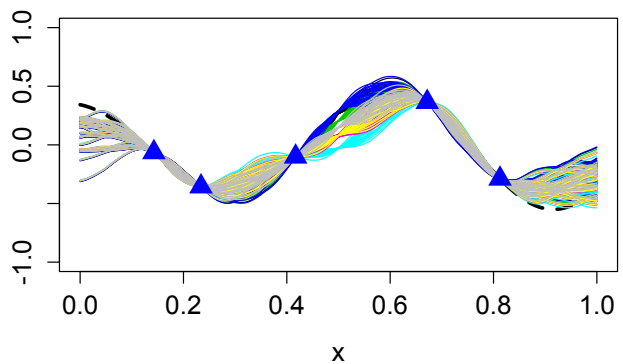


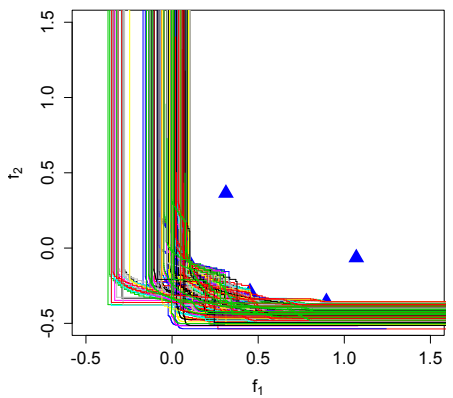




Update at $x_{n+1} = 0.5$

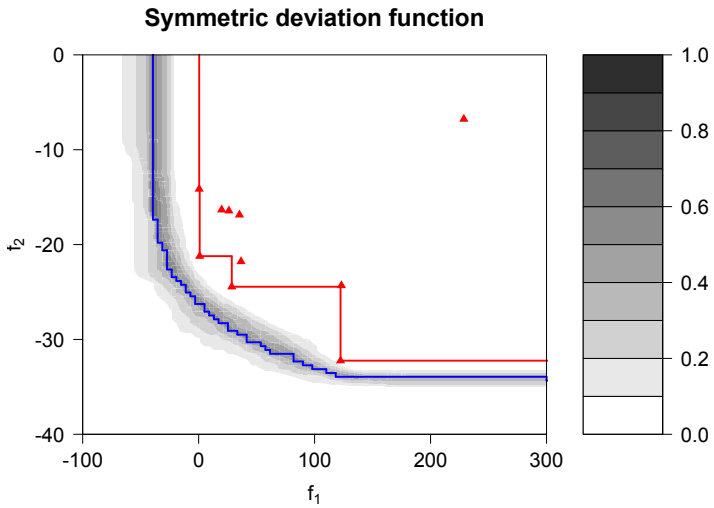




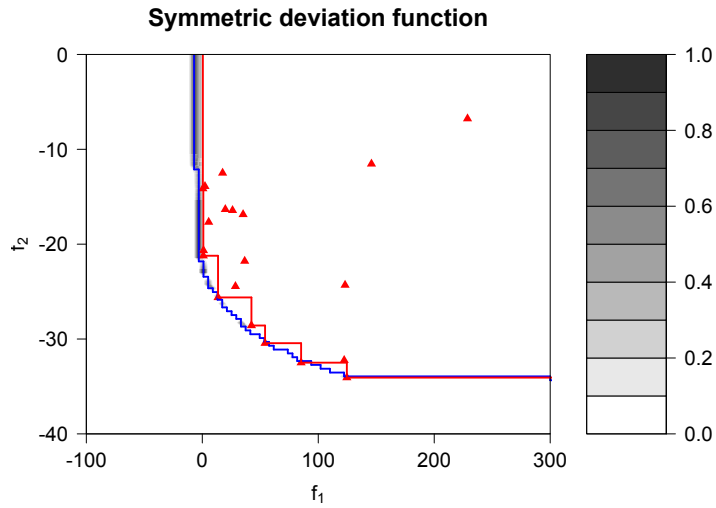


Performance of this SUR criterion

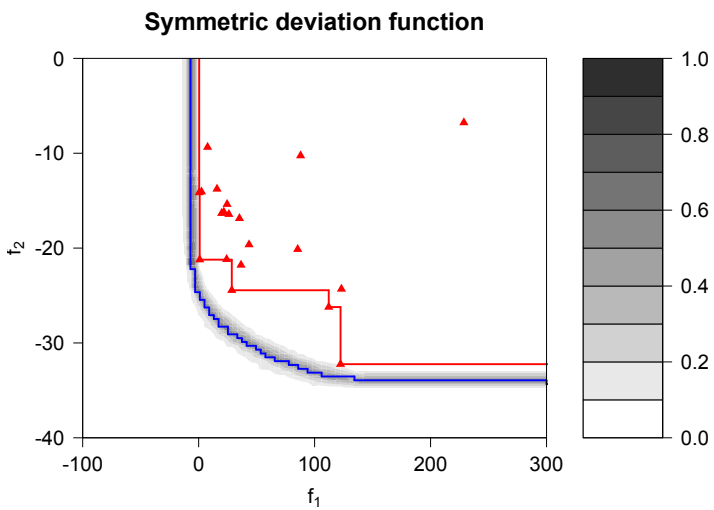
Initial state ($n = 10$)



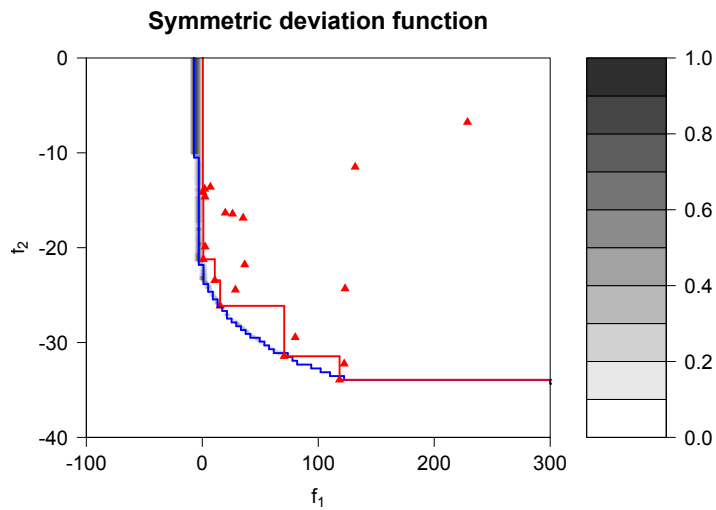
EHI ($n = 20$)



Random ($n = 20$)

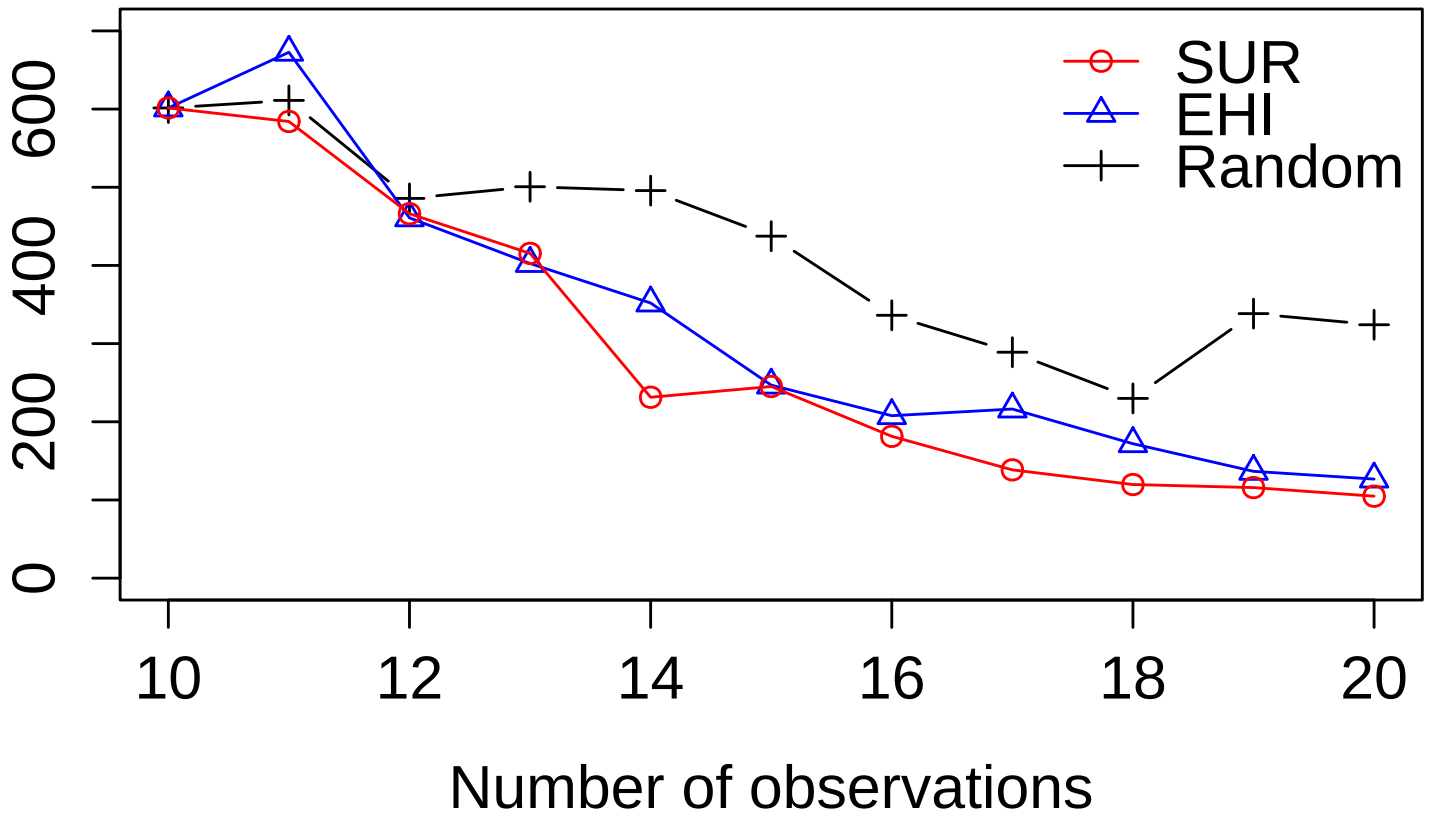


SUR ($n = 20$)



Performance of this SUR criterion

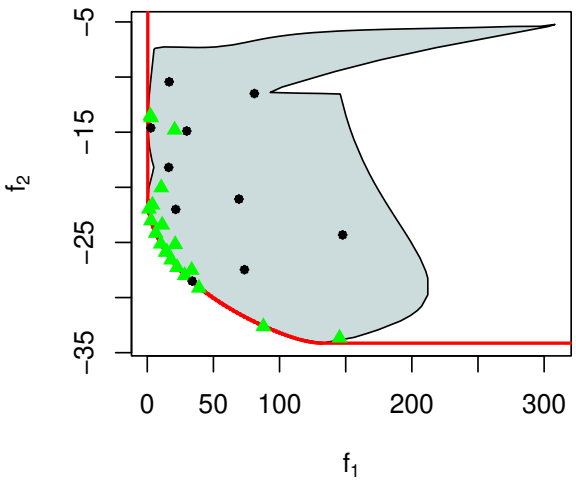
Monitoring of Vorob'ev deviation value:



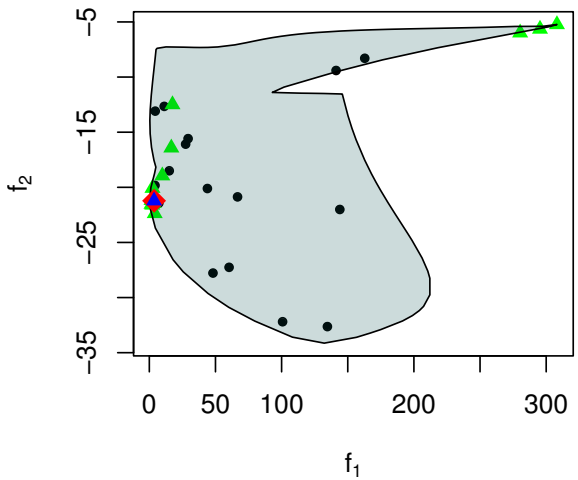
Complementing multi-objective optimization with game theory

Goal: optimize more outputs (m) simultaneously

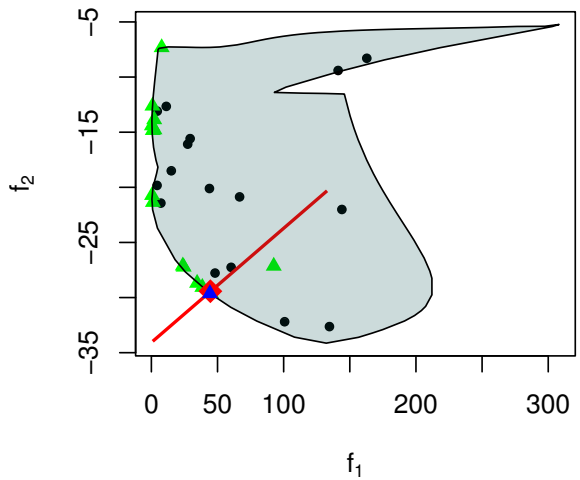
Examples :



Pareto front



Nash equilibrium



Kalai-Smorodinsky solution

[Binois and Picheny, 2019, Picheny et al., 2018, Binois et al., 2019]

Plan

Calibration problem

Suppose that we have observations of a physical phenomenon depending on some variables $\mathbf{x}$, with an observable y (corrupted by noise, $y(\mathbf{x}) = y_R(\mathbf{x}) + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$:
That is, we have $(\mathbf{x}_i, y_i)$, $1 \leq i \leq n$;

In many cases, there also exist a mathematical/computer model y_M of the phenomenon:

- still, it may take some time to evaluate;
- there may exist a bias with reality (imperfect model);
- they involve additional tuning parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$ that can be controlled. That is, we have can evaluate $y_M(\mathbf{x}, \boldsymbol{\theta})$.

Calibration problem (cont'd)

The goal is to reconcile the model with the data, i.e., find the *best* $\boldsymbol{\theta}$ corresponding to the data.
A natural way is to define *best* as follows:

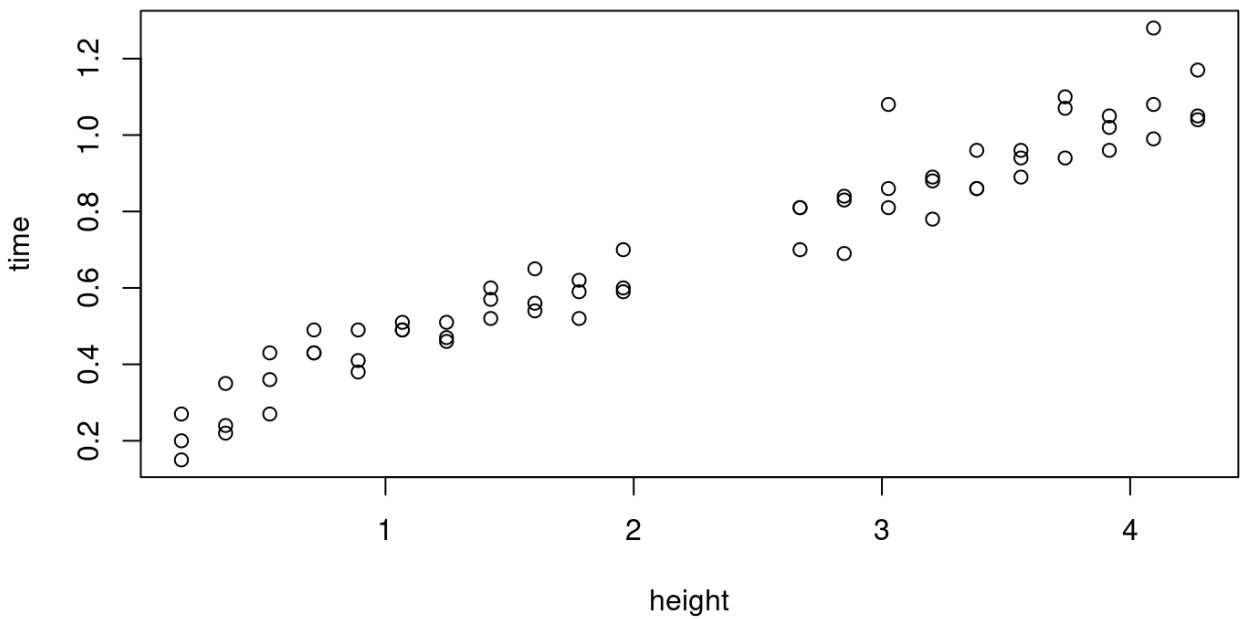
$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \left(\sum_{i=1}^{n} (y_i - y_M(x_i, \boldsymbol{\theta}))^2 \right)$$

Maybe we can do more:

- quantify uncertainty on the optimal $\boldsymbol{\theta}$
  $\rightarrow$ Bayesian calibration : find $\mathbb{P}(\boldsymbol{\theta}|\mathbf{y})$, given a prior on $\boldsymbol{\theta}$: $\mathbb{P}(\boldsymbol{\theta})$ (estimated by Markov Chain Monte Carlo - MCMC)
- estimate the bias (and potentially correct it)
  $\rightarrow$ Kennedy and O'Hagan framework: use a GP to model the bias and use its likelihood to drive the MCMC.

Calibration example (from [Gramacy, 2020])

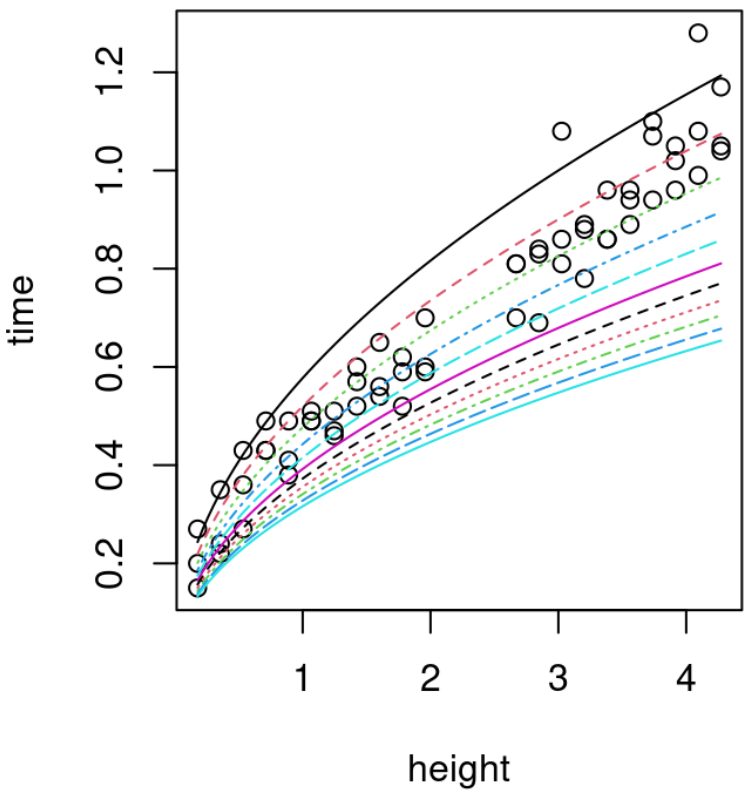
The aim is to predict the amount of time it takes for a ball to hit the ground depending on the height it is dropped from.

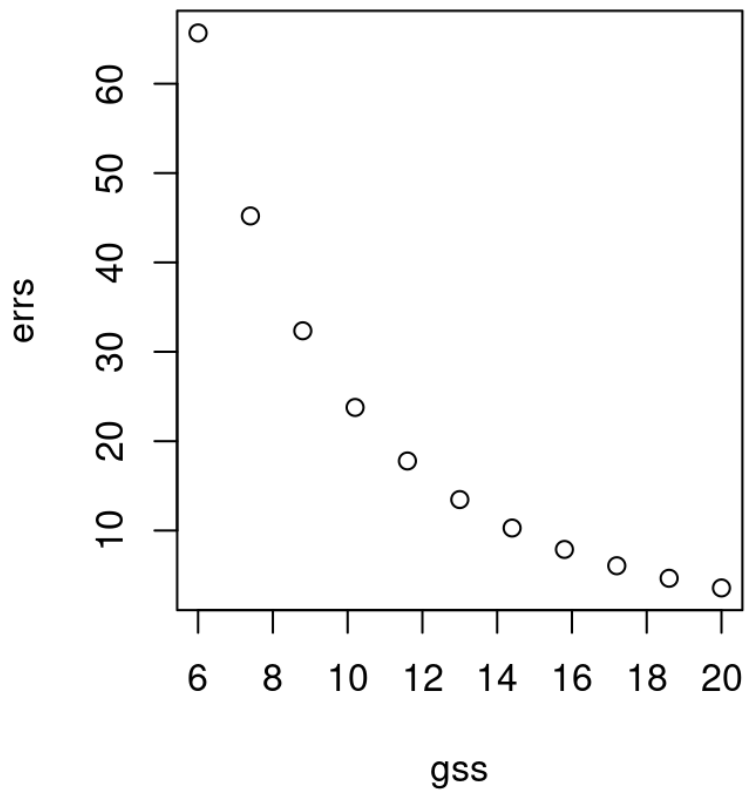


The mathematical model (coming from physics) says that the time of the fall of a height h is $t = \sqrt{\frac{2h}{g}}$.

Calibration example (from [Gramacy, 2020]) (cont'd)

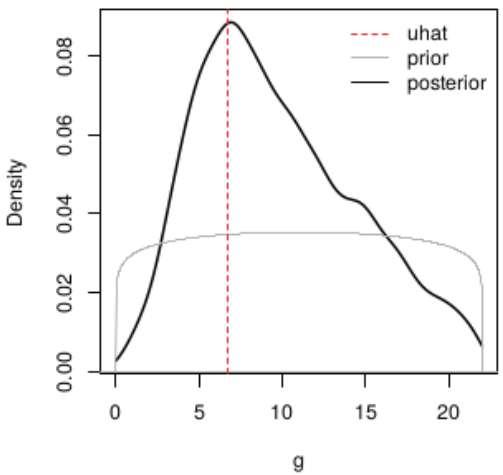
Suppose we don't recall the value of g and want to recover it.

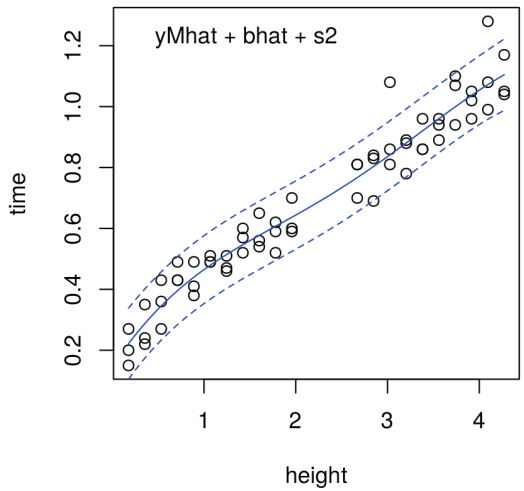


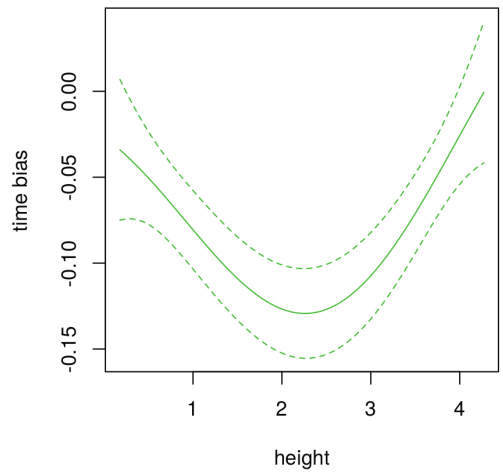


Calibration example (from [Gramacy, 2020]) (cont'd)

Results from [Gramacy, 2020]:

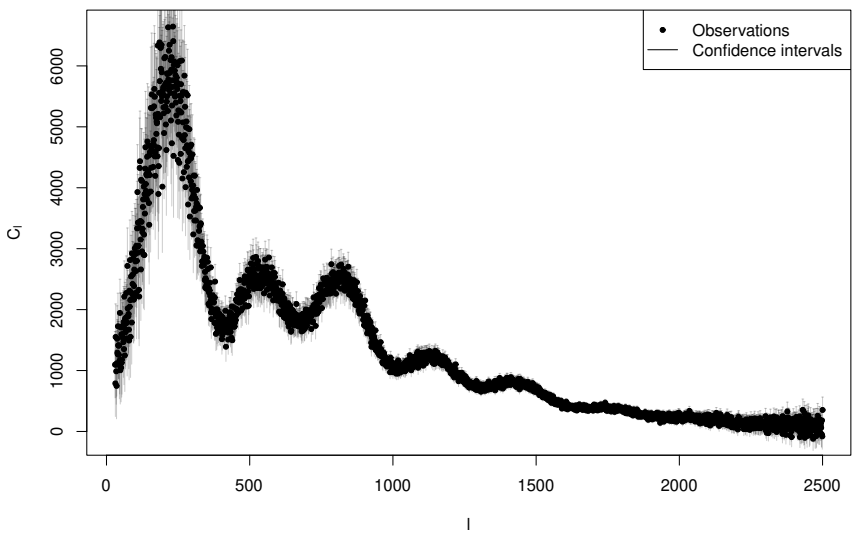




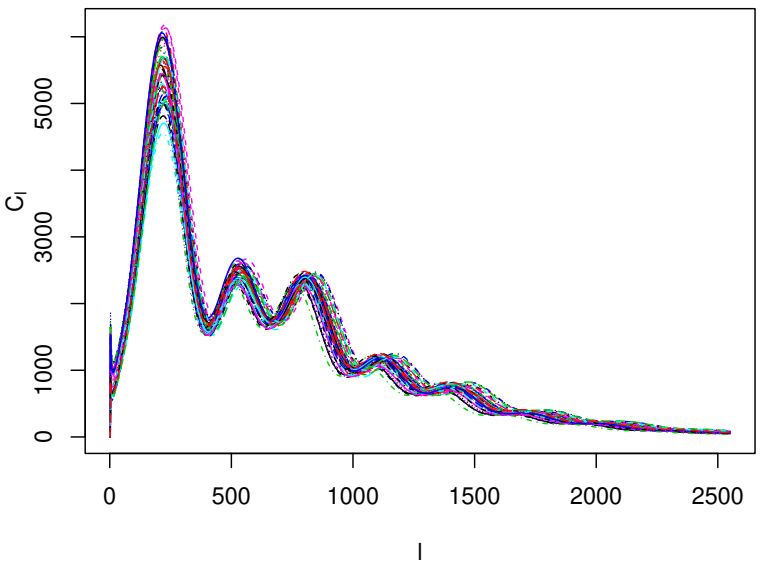


Calibration example (2): Code for Anisotropies in the Microwave Background

Data: Planck survey
CAMB: code for calculating cosmological observables, here angular power spectra for temperature, with parameters $\boldsymbol{\theta}$



Planck data



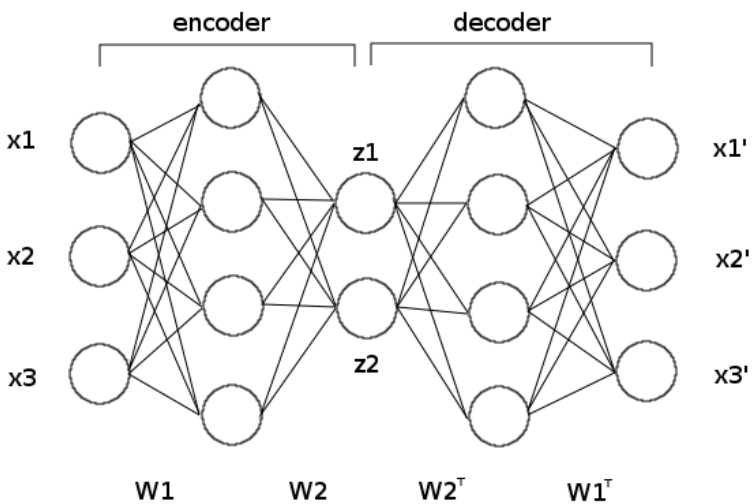
Model output examples

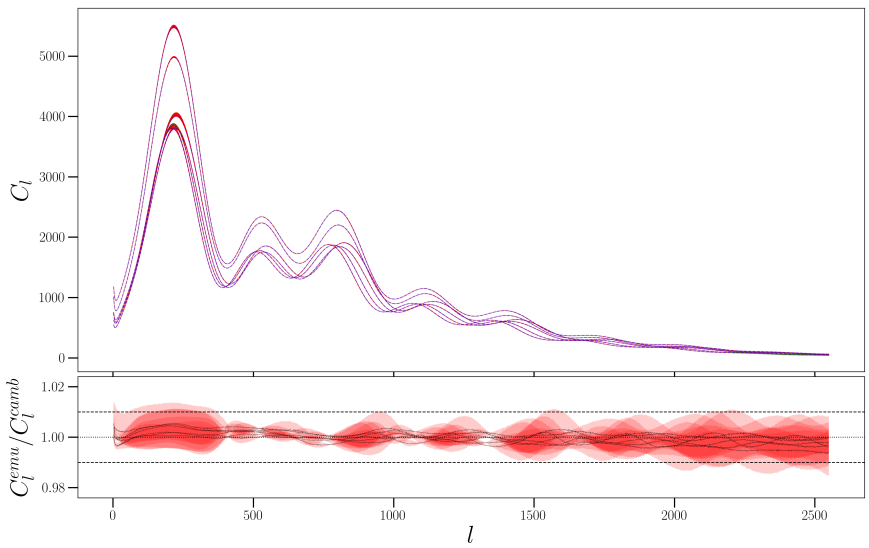
Calibration example (2): Code for Anisotropies in the Microwave Background

Approach: emulate CAMB outputs via decomposition on a functional basis, and model the coefficients:

$$Y(\boldsymbol{\theta}, l) = \sum_{i=1}^{p} w_i(\boldsymbol{\theta})\phi_i(l)$$

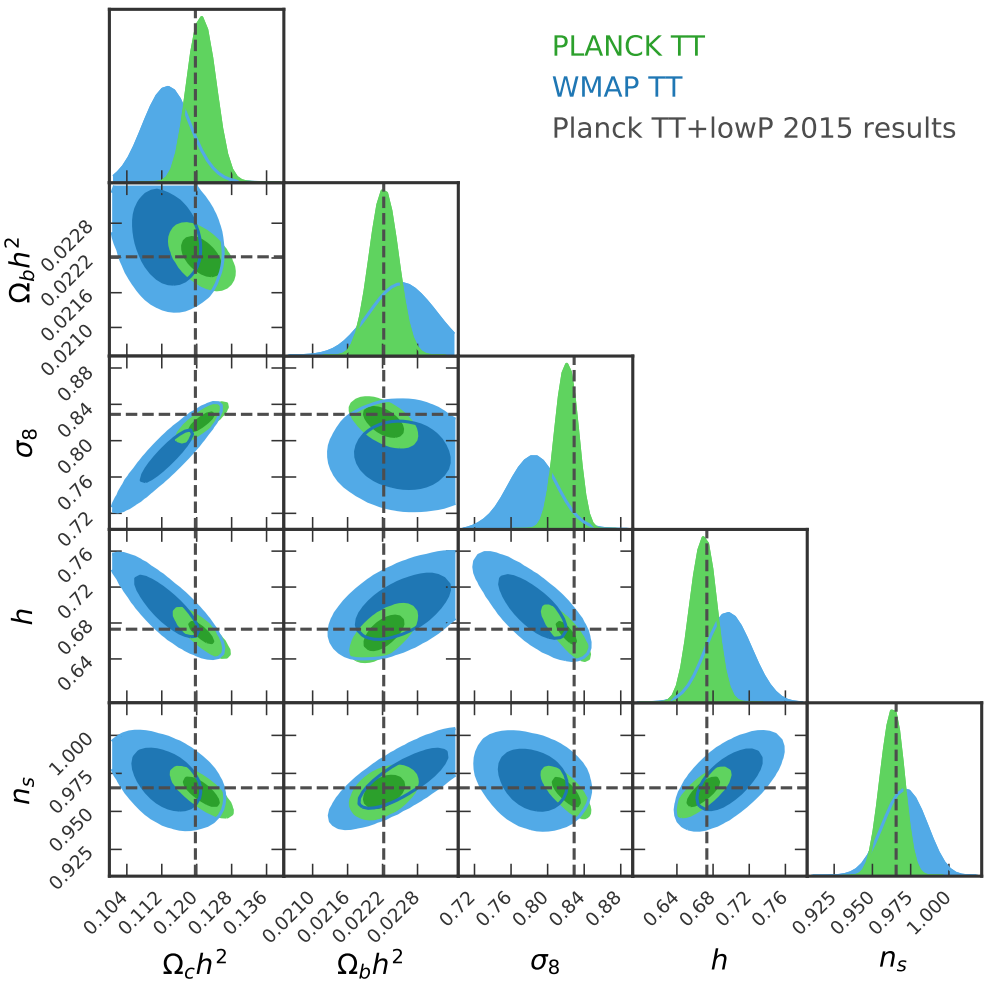
Reference method: PCA for ϕ_i and GP for w_i
Alternative method: variational autoencoder for ϕ and GP for w





Calibration example (2): Code for Anisotropies in the Microwave Background

Results:

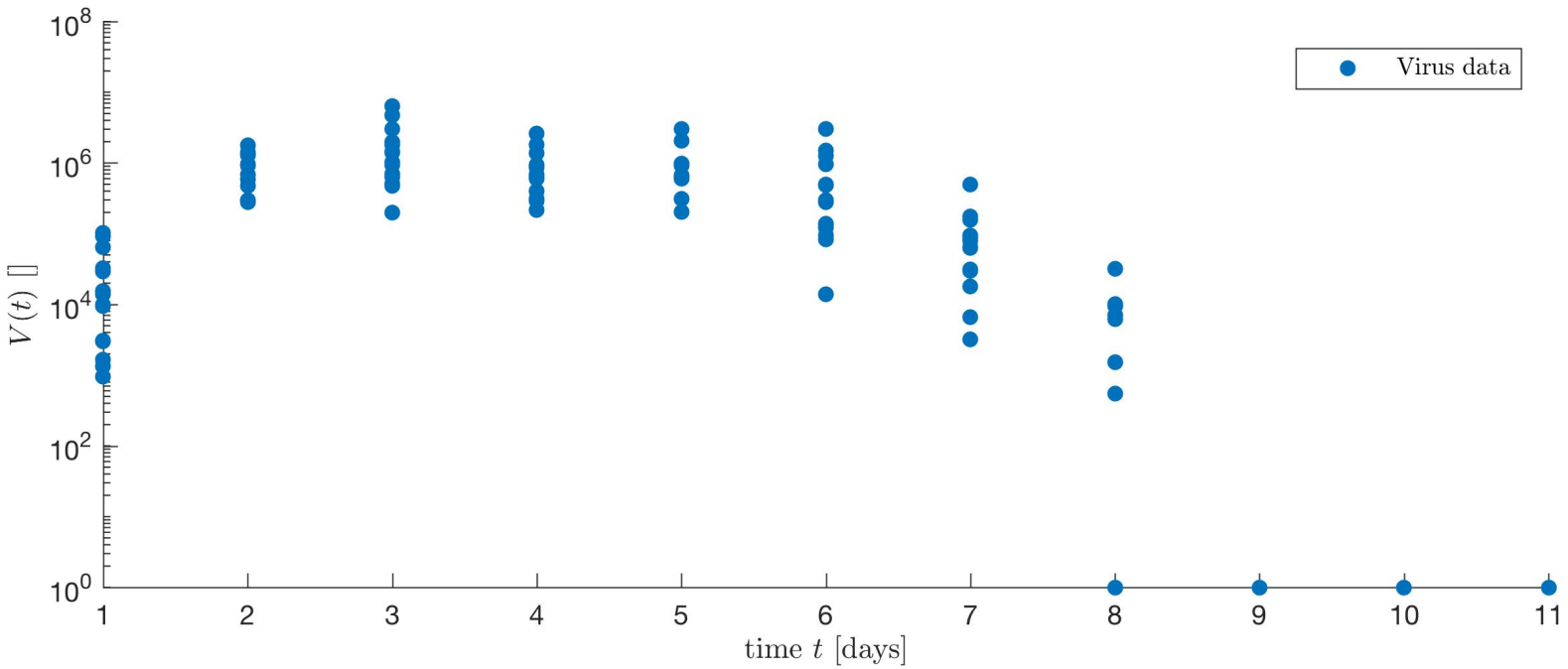


Calibration example (3): Influenza A

Influenza A virus is a frequent cause of lower respiratory tract infections, causing over 15 million diagnoses resulting in 200 000 hospitalizations each year.

Scope: get deeper understanding of infection mechanisms with mathematical models, paired with experimental data

Data: Viral lung titers from individual mice infected with 75 TCID 50 influenza H1N1



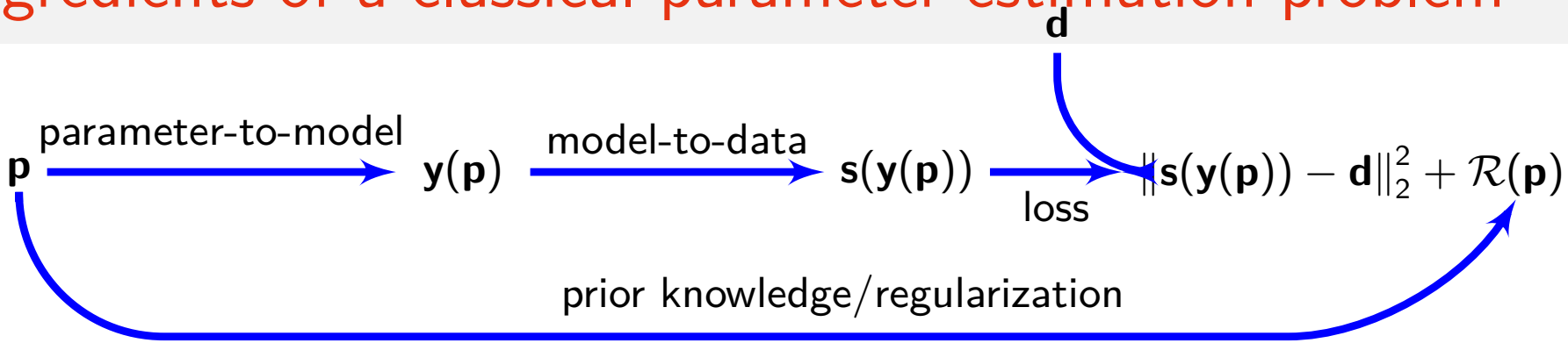
Mathematical model [Myers2017]

Tracks susceptible ("target") cells T and classes of infected cells I_1 and I_2, and virus V.
Target cells become infected with virus at rate βV per cell.
Once infected, they enter an eclipse phase I_1 at rate κ per cell before transitioning to produce virus at rate ρ per cell I_2.
Virus is cleared at rate c and virus-producing infected cells I_2 are cleared in a density dependent manner with max rate δ and half-saturation constant K_d.
The following system of differential equations describes these dynamics:

$$\begin{aligned} T' &= -\beta TV, \\ I_1' &= \beta TV - \kappa I_1, \\ I_2' &= \kappa I_1 - \frac{\delta I_2}{K_d + I_2}, \\ V' &= \rho I_2 - cV. \end{aligned}$$

The system is of the form $\mathbf{y}' = \mathbf{f}(t, \mathbf{y}, \mathbf{p})$. The state variables are given by $\mathbf{y}(t) = [T(t), I_1(t), I_2(t), V(t)]\top$. The parameters $\beta, \kappa, \delta, K_d, \rho, c$ and initial conditions $T(0), I_1(0), I_2(0), V(0)$ uniquely determine the initial value problem.

Ingredients of a classical parameter estimation problem



Parameter $\mathbf{p}$ defines the specific model $\mathbf{y}$, which then gets mapped by $\mathbf{s}$ onto the data $\mathbf{d}$, and finally measured with a quality measure $\mathcal{J}$, that may include prior knowledge $\mathcal{R}$ about $\mathbf{p}$.

For the inverse problem we want to find a $\widehat{\mathbf{p}}$ with best quality measure $\mathcal{J}$, given $\mathbf{y}$, $\mathbf{s}$, regularization $\mathcal{R}$, and data $\mathbf{d}$.

But:

- this problem is very often ill-posed
- and prior knowledge introduction can be tricky

UQ is generally performed by local sensitivity analysis. Alternatives include Bayesian methods, via MCMC.

Proposed alternative [Chung et al., 2019]

Here, we propose to perform the regularization implicitly, via a surrogate stochastic process of the data.

Then posterior samples are used to obtain a distribution for the parameters, using the classical framework.

Algorithm 1 Parameter Estimation

input: data $\mathbf{d}$ and ODE model

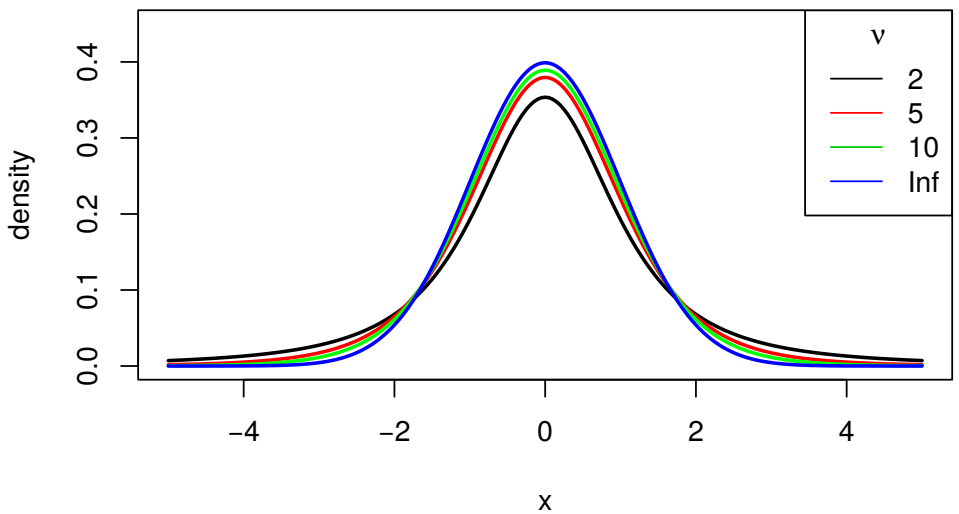
1: use $\mathbf{d}$ to generate stochastic process $\mathcal{G}$
2: **parallel for** $j = 1$ to J **do**
3: get sample $\mathbf{g}_j$ from $\mathcal{G}$
4: compute $\widehat{\mathbf{p}}_j$ using $\mathbf{g}_j$
5: **end parallel for**

output: $\{\widehat{\mathbf{p}}_j\}_{j=1}^{J}$

Much easier to include prior information such as regularity.

Handling noise with larger tails than Gaussian

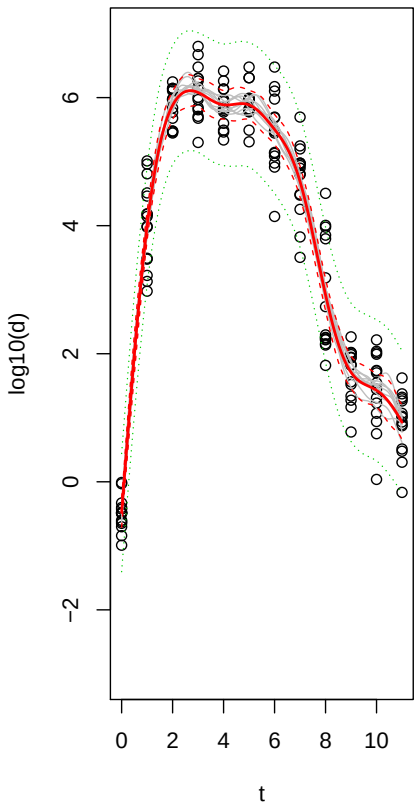
The Gaussian distribution is a special case of the Student distribution (with $\nu = \infty$)

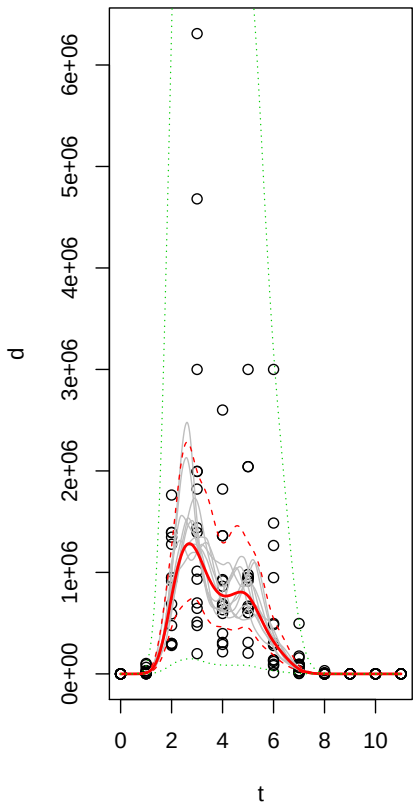


[Shah2014] generalized GPs to Student-t processes, with homoskedastic noise.
Turns out that it can be extended further as we showed for GPs.

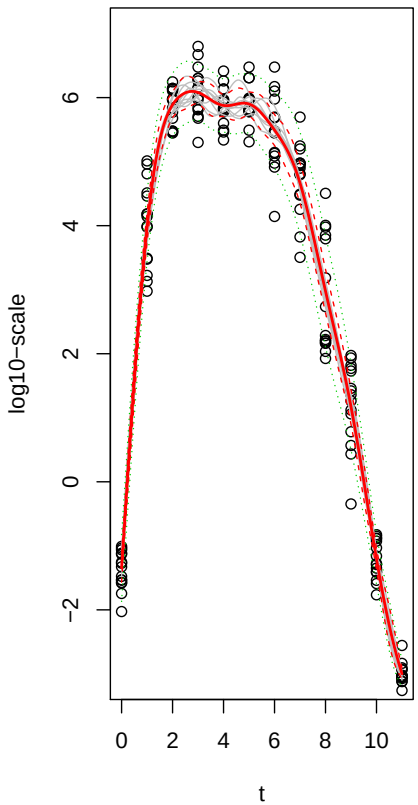
Comparison of heteroskedastic GPs and TPs

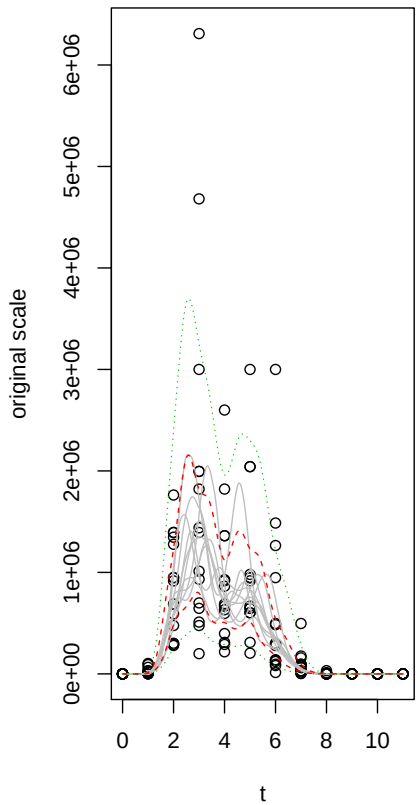
GP



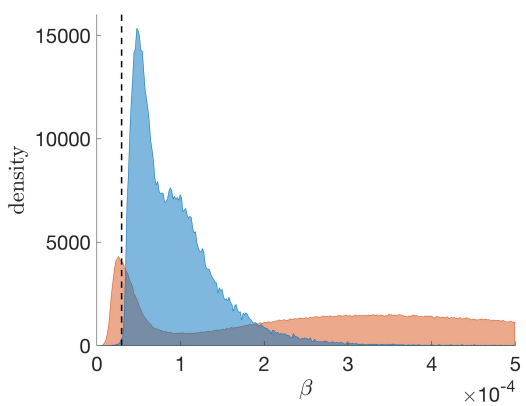


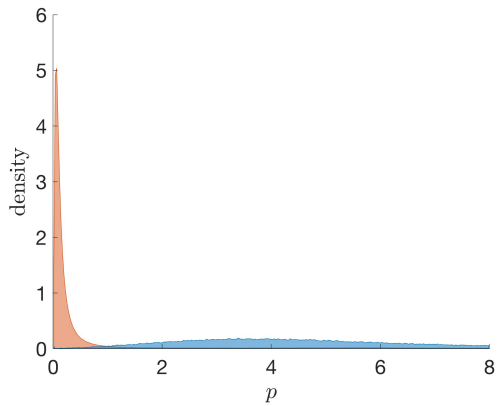
TP

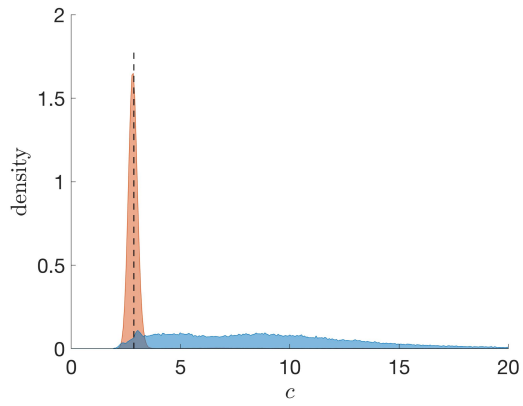


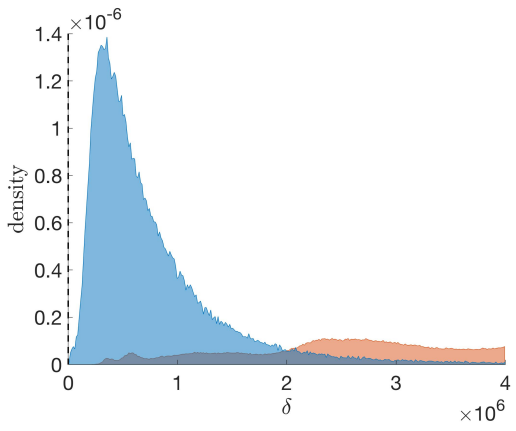


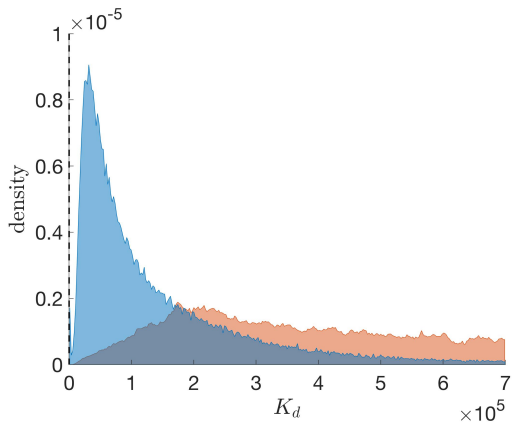
Results

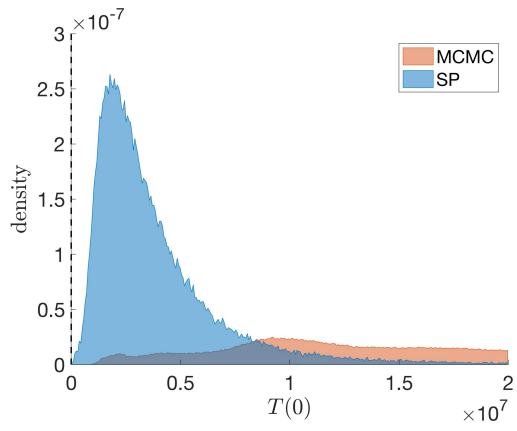








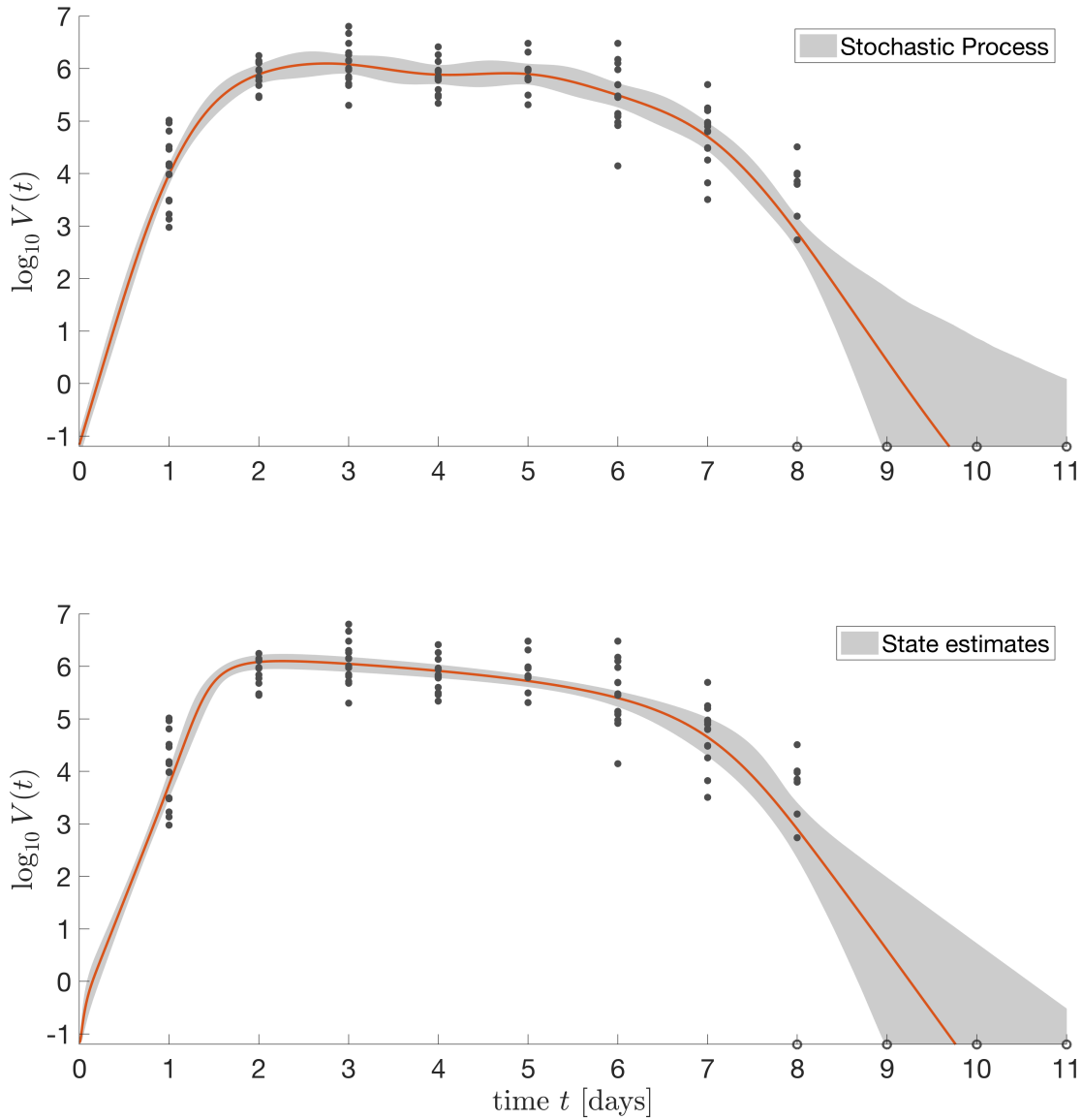




MCMC results

proposed methodology

Results


Stochastic Process
State estimates
$\log_{10} V(t)$
time t [days]
7
6
5
4
3
2
1
0
-1
0
1
2
3
4
5
6
7
8
9
10
11

Concluding remarks

Gaussian processes are the power horse in many sequential design problems.

They provide sensible uncertainty quantification, amenable to uncertainty propagation.

Many difficulties can be handled (non-stationarity, multi-objective, etc.) but no unified framework.

References I


Binois, M. and Picheny, V. (2019).
Gpareto: An r package for gaussian-process-based multi-objective optimization and analysis.
Journal of Statistical Software, 89(1):1–30.


Binois, M., Picheny, V., Taillandier, P., and Habbal, A. (2019).
The Kalai-Smorodinski solution for many-objective bayesian optimization.
arXiv preprint arXiv:1902.06565.


Chevalier, C., Emery, X., Ginsbourger, D., et al. (2014).
Fast update of conditional simulation ensembles.
Mathematical Geosciences.

References II


Chevalier, C., Ginsbourger, D., Bect, J., and Molchanov, I. (2013).
Estimating and quantifying uncertainties on level sets using the vorob'ev expectation and deviation with gaussian process models.
In Ucinski, D., Atkinson, A. C., and Patan, M., editors, *mODa 10 – Advances in Model-Oriented Design and Analysis*, Contributions to Statistics, pages 35–43. Springer International Publishing.


Chung, M., Binois, M., Gramacy, R. B., Bardsley, J. M., Moquin, D. J., Smith, A. P., and Smith, A. M. (2019).
Parameter and uncertainty estimation for dynamical systems using surrogate stochastic processes.
SIAM Journal on Scientific Computing, 41(4):A2212–A2238.


Contal, E. and Vayatis, N. (2013).
Gaussian process optimization with mutual information.
arXiv preprint arXiv:1311.4825.

References III


Emmerich, M. T., Deutz, A. H., and Klinkenberg, J. W. (2011).
Hypervolume-based expected improvement: Monotonicity properties and exact computation.
In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pages 2147–2154. IEEE.


Erickson, C. B., Ankenman, B. E., and Sanchez, S. M. (2017).
Comparison of gaussian process modeling software.
European Journal of Operational Research.


Girard, A. (2004).
Approximate methods for propagation of uncertainty with gaussian process models.
PhD thesis, Citeseer.

References IV


Gramacy, R. B. (2020).
Surrogates: Gaussian Process Modeling, Design, and Optimization for the Applied Sciences.
CRC Press.


Hernández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. (2014).
Predictive entropy search for efficient global optimization of black-box functions.
In *Advances in Neural Information Processing Systems*, pages 918–926.


Mockus, J. (1989).
Bayesian approach to global optimization.
Springer.

References V


Mockus, J., Tiesis, V., and Zilinskas, A. (1978).
The application of Bayesian methods for seeking the extremum.
Towards Global Optimization, 2(117-129):2.


Molchanov, I. (2005).
Random closed sets.
In *Space, Structure and Randomness*, pages 135–149. Springer.


Picheny, V., Binois, M., and Habbal, A. (2018).
A bayesian optimization approach to find nash equilibria.
Journal of Global Optimization.


Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. (2009).
Gaussian process optimization in the bandit setting: No regret and experimental design.
arXiv preprint arXiv:0912.3995.

References VI


Villemonteix, J., Vazquez, E., and Walter, E. (2009).
An informational approach to the global optimization of expensive-to-evaluate functions.
Journal of Global Optimization, 44(4):509–534.


Wang, Z. and Jegelka, S. (2017).
Max-value entropy search for efficient bayesian optimization.
arXiv preprint arXiv:1703.01968.

Thanks

Questions?