Beyond Surrogate Modeling : Learning the Local Volatility via Shape Constraints

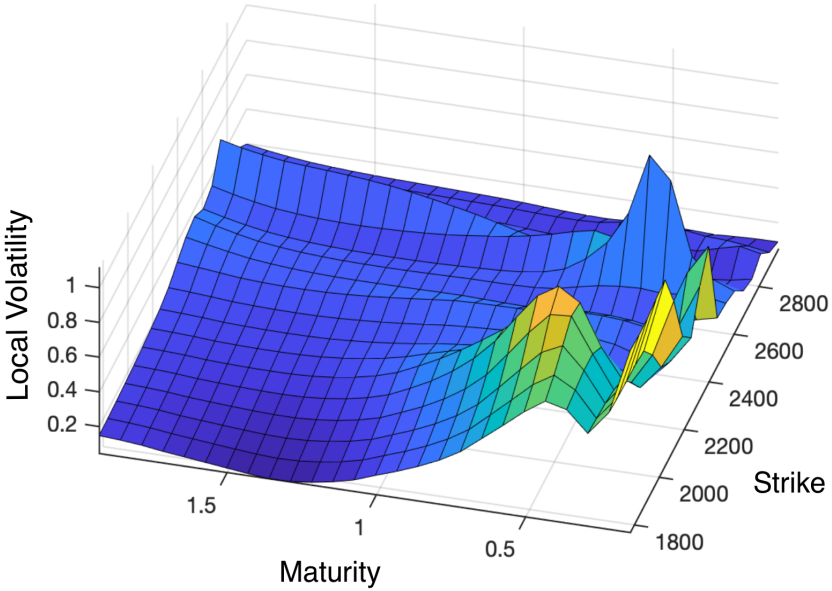
Areski Cousin
IRMA, Université de Strasbourg

Joint work with Marc Chataigner, Stéphane Crépey, Matthew Dixon and Djibril Gueye

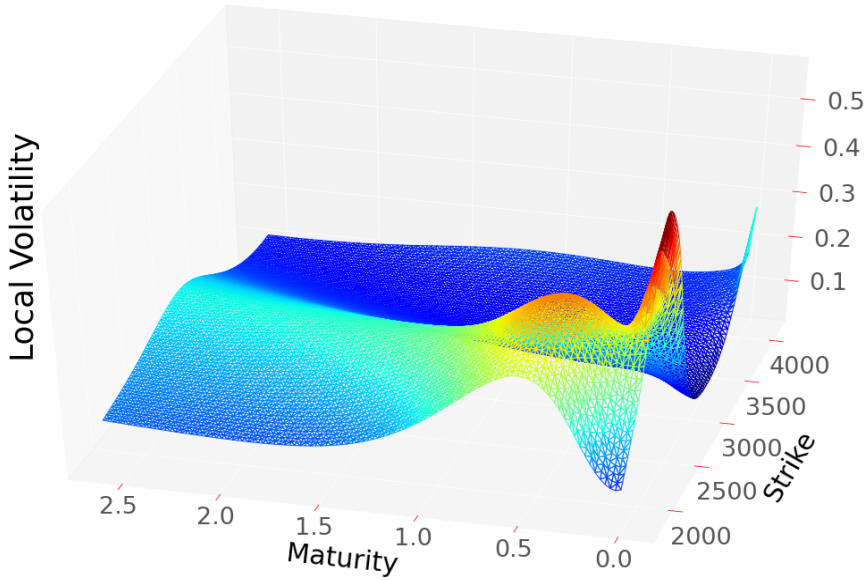
Séminaire équipe Calisto - INRIA, July 2, 2021

Motivation

- We explore 2 alternatives machine learning approaches for the construction of local volatility surfaces
- A no-arbitrage Gaussian process (GP) approach based on price and a neural net (NN) approach with penalization of arbitrages based on implied volatility
- Construction at a given cotation date, from observations of bid-ask prices for a set of liquidly traded European options



GP estimate



NN estimate

General Problem Statement [no-arbitrage GP approach]

In this presentation, we focus on the GP approach

- Construct a GP surface estimate $P : \mathbb{R}^d \to \mathbb{R}$, interpolating noisy option prices $\{\mathbf{x}_i, y_i\}_{i=1}^n$ so that $y_i = P(\mathbf{x}_i) + \epsilon_i$ for some additive noise ϵ_i, $\forall i$.
- For a given option, bid and ask prices considered as two (noisy) replicates of P
- Subject to shape constraints, i.e., non-decreasing in maturity and convex in strike (to avoid calendar spread and butterfly arbitrages).
- How to effectively enforce shape constraints in the GP regression ?
- Uncertainty quantification under shape constraints : how to derive no-arbitrage confidence bands ?

Connections between option price, IV and local volatility

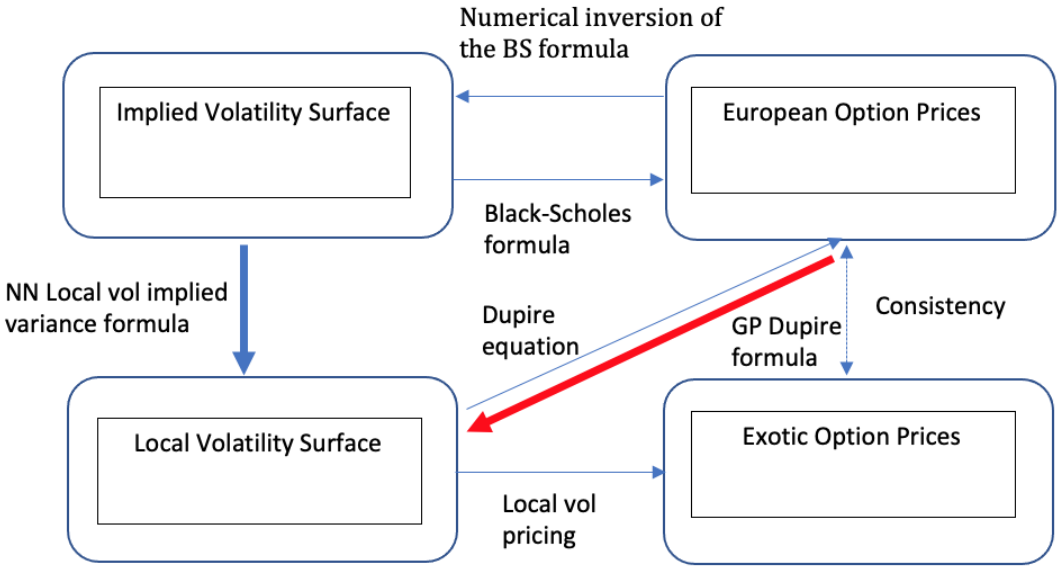


Figure: Mathematical connections between option prices, implied, and local volatility. The **bold red arrow** shows the route under the no-arbitrage GP approach based on price. The **bold blue arrow** shows the route under the NN approach based on implied volatility.

Absence of static arbitrage

The put price function $(T, K) \to P(T, K)$ is considered to be free of static arbitrage if there exists a measure $\mathbb{Q}$ such that the discounted asset $e^{-(r-q)t} S_t$ is a $\mathbb{Q}$–martingale and $P(T, K) = \mathbb{E}_{\mathbb{Q}} \left[e^{-rT} (K - S_T)^+ \right]$.

Absence of arbitrage [constant interest rate r and dividend yield q]

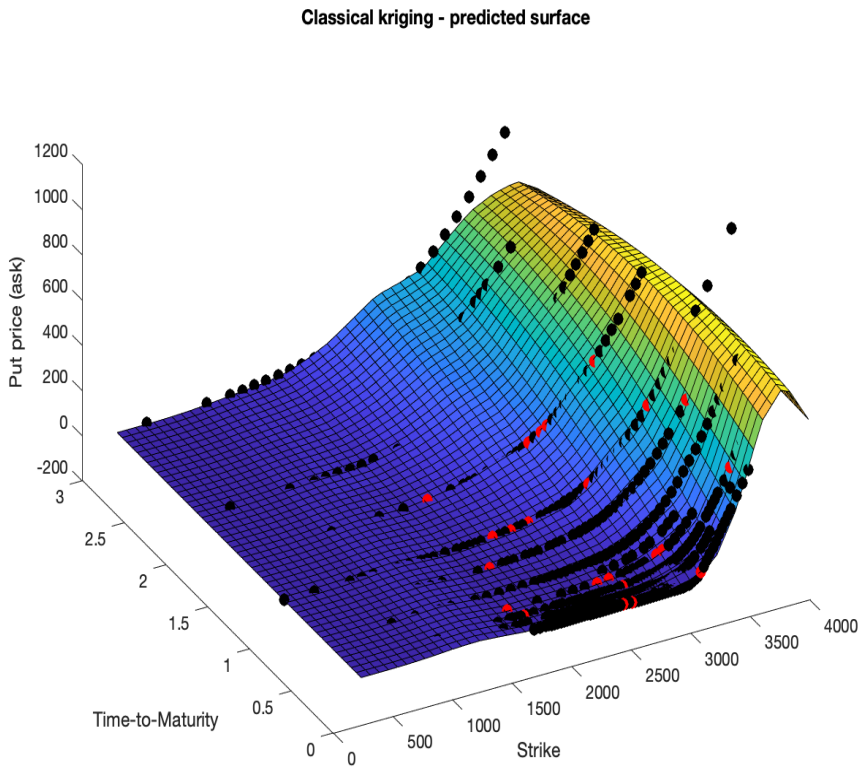
Let $p(T, k) := e^{qT} P(T, K)$ where $k = e^{-(r-q)T} K$ be the *reduced* put price. The put price surface $(T, K) \to P(T, K)$ is free of static arbitrage if and only if the reduced price function p is such that

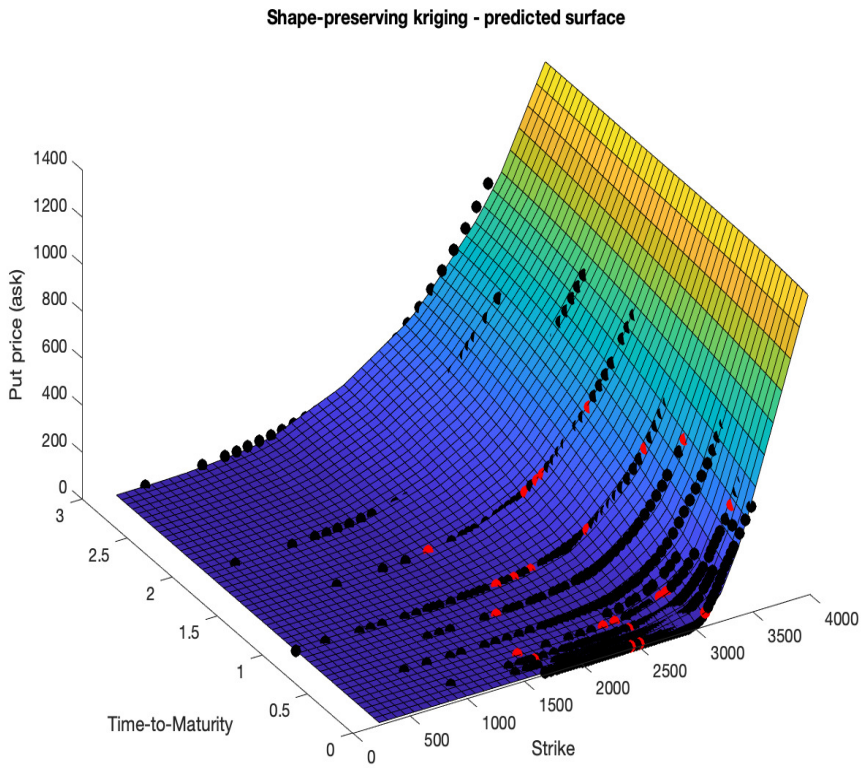
- $p(\cdot, k)$ is a non-decreasing function and $p(0, k) = (k - S_0)^+$, for any $k \geq 0$
- $p(T, \cdot)$ is a convex function, $p(T, 0) = 0$, $\frac{\partial p}{\partial k}(T, 0) = 0$ and $\lim_{k \to \infty} p(T, k) = k - S_0$, for any $T \geq 0$

GP regression with shape constraints

Illustrative example

- Data : Euro Stoxx 50 put prices, January 10, 2019
- 5% of the data used (red points), zero-mean Gaussian prior with a Gaussian kernel
- Classical GP (left) vs GP with no-arbitrage constraints (right)

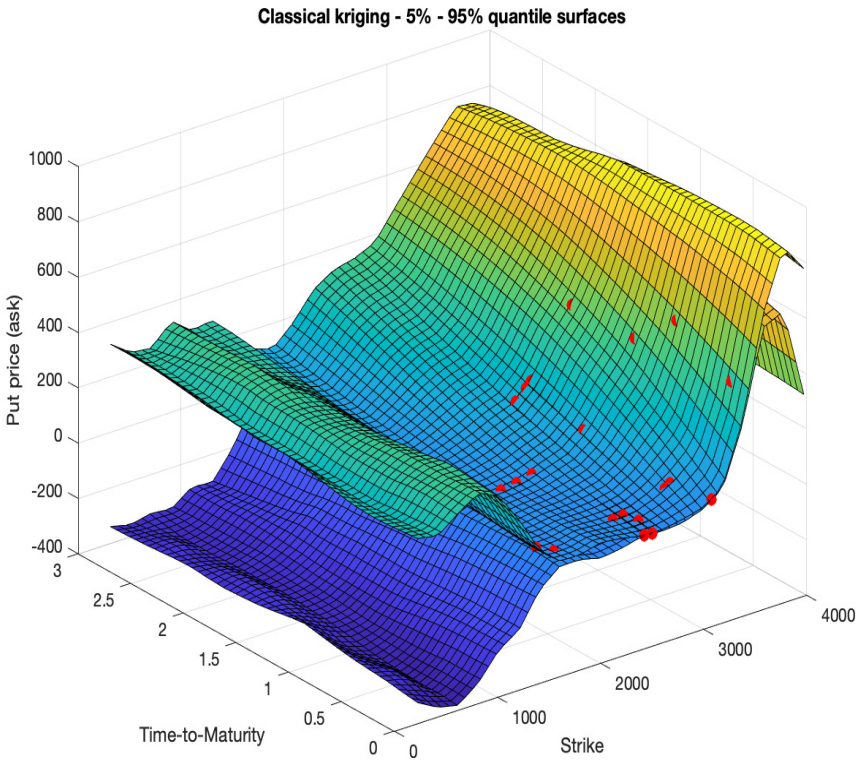


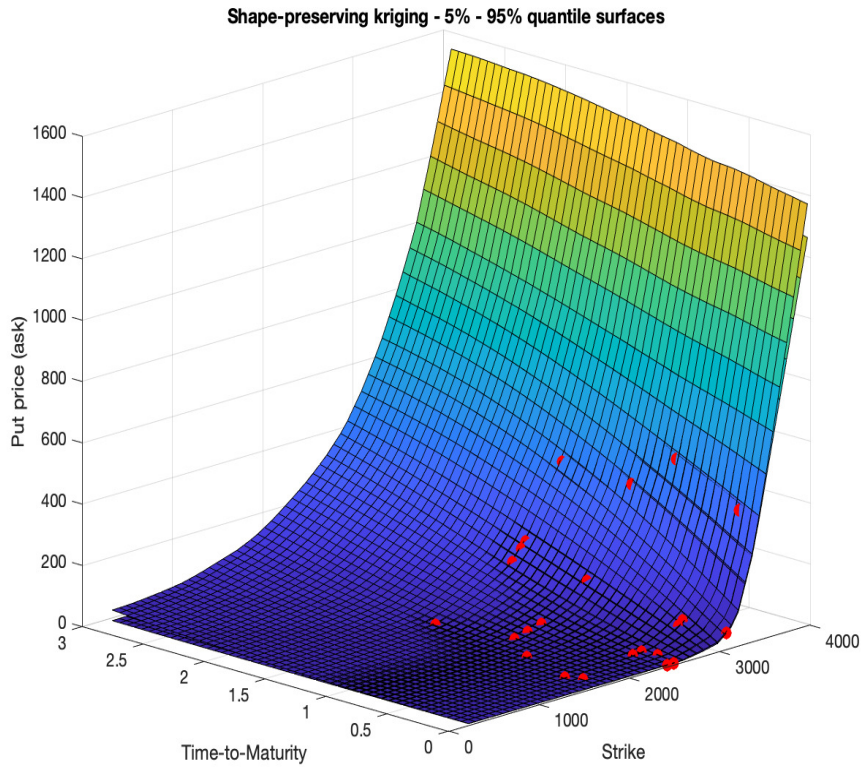


GP regression with shape constraints

Kriging of option price surface - quantification of uncertainty

- Pointwise 5% and 95% estimated quantiles of the fitted GP
- Classical kriging (left) vs kriging with no-arbitrage constraints (right)





GP regression with shape constraints

Assume that the a priori belief on the (reduced) price surface p is given as a GP prior process Y. The GP approach with shape constraints consists in estimating the conditional distribution of Y given

$$\begin{cases} \boldsymbol{y} = Y(X) + \varepsilon \\ Y \in \mathcal{M} \end{cases}$$

where $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$, $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$, ε is a zero-mean Gaussian noise in $\mathbb{R}^n$ (independent of Y) and $\mathcal{M}$ is a convex set of functions satisfying some shape properties.

For instance, $\mathcal{M}$ can be :

- $\mathcal{M}_0^d := \{f \in \mathcal{C}([0,1]^d, \mathbb{R}) \mid y_{\min} \leq f(x) \leq y_{\max},\ \forall x \in D\}$
- $\mathcal{M}_1^1 := \{f \in \mathcal{C}([0,1], \mathbb{R}) \mid f \text{ is non-decreasing}\}$
- $\mathcal{M}_2^1 := \{f \in \mathcal{C}([0,1], \mathbb{R}) \mid f \text{ is convex}\}$
- $\mathcal{M}_{12}^2 := \{f \in \mathcal{C}([0,1]^2, \mathbb{R}) \mid f \text{ is non-decreasing in } t \text{ and convex in } x\}$

GP regression with shape constraints

Main issues :

- The posterior process is not Gaussian
- The shape condition is usually infinite-dimensional

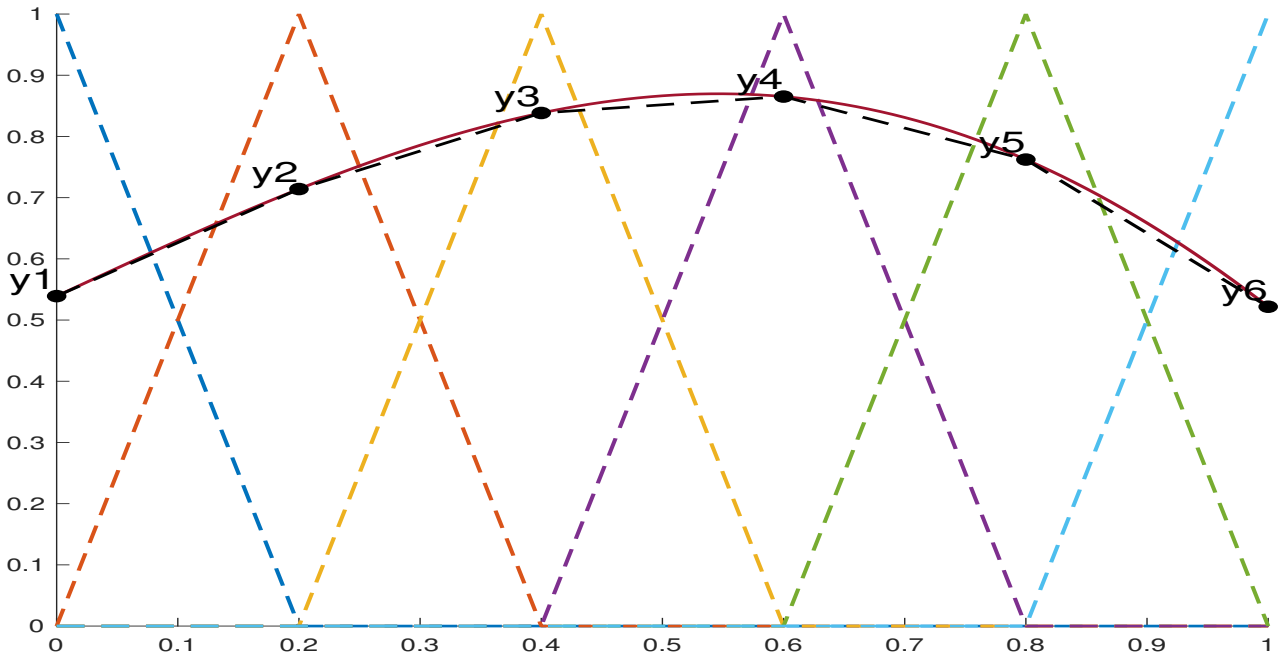
Proposed solutions :

- We construct a finite-dimensional approximation of Y for which the shape condition is easy to check.
- We consider the mode of the posterior distribution (as opposed to the posterior mean) as the response surface predictor
- Hyper-parameters are estimated using (unconstrained) MLE
- Sampling of the posterior (no-arbitrage) distribution by Hamiltonian Monte Carlo starting from the mode

Finite-dimensional approximation of GP (1d case)

As in Maatouk and Bay (2014), Cousin et al. (2016), López et al. (2018), we rely on basis function approximation.

- Input domain D is normalized on $[0,1]$ and discretized on a regular subdivision $u_0 < \ldots < u_N$ with a constant mesh δ.
- For each u_i, we consider hat functions $\phi_i(x) := \max\left(1 - \frac{|x-u_i|}{\delta}, 0\right)$
- Y is approximated on D by $Y^N(x) = \sum_{i=0}^{N} Y(u_i)\phi_i(x)$



Finite-dimensional approximation of GP (2d case)

- $D = [0,1]^2$ is discretized on a $(N_t + 1) \times (N_x + 1)$ regular grid with knots (u_i, v_j), $i = 0, \ldots, N_t$, $j = 0, \ldots, N_x$.
- For each knot (u_i, v_j), we consider tensor product basis functions

$$\phi_{i,j}(t,x) := \max\left(1 - \frac{|t - u_i|}{\delta_t}, 0\right) \max\left(1 - \frac{|x - v_j|}{\delta_x}, 0\right)$$

- Y is approximated on D by

$$Y^N(t,x) = \sum_{i=0}^{N_t} \sum_{j=0}^{N_x} Y(u_i, v_j)\phi_{i,j}(t,x)$$

- $N = (N_t + 1)(N_x + 1)$ is the number of knots

Finite-dimensional approximation of GP (2d case)

Proposition

Let Y be a zero-mean GP with covariance function K and with almost surely continuous paths.

- The finite-dimensional process Y^N uniformly converges to Y on D as $N_t \to \infty$ and $N_x \to \infty$, almost surely.
- $Y^N(t,x) = \mathbf{\Phi}(t,x)\boldsymbol{\xi}$ where $\boldsymbol{\xi} := (Y(u_0, v_0), Y(u_0, v_1), \ldots, Y(u_{N_t}, v_{N_x}))^\top$ is a zero-mean Gaussian vector with $N \times N$ covariance matrix Γ^N such that $\Gamma^N = K((u_{i_1}, v_{j_1}), (u_{i_2}, v_{j_2}))$.

Shape-preserving conditions :

- Y^N is bounded on $[y_{\min}, y_{\max}]$ if and only if $y_{\min} \leq \xi_{i,j} \leq y_{\max}$
- $Y^N(t,x)$ is a non-decreasing function of t if and only if $\xi_{i+1,j} \geq \xi_{i,j}$
- $Y^N(t,x)$ is a convex function of x if and only if $\xi_{i,j+2} - \xi_{i,j+1} \geq \xi_{i,j+1} - \xi_{i,j}$
- $\ldots$

GP regression with shape constraints

New formulation of the problem : Consider a zero-mean GP prior Y with covariance function K. The N-dimensional approximation Y^N of Y on D is such that, for any $\boldsymbol{x} \in D$, $Y^N(\boldsymbol{x}) = \boldsymbol{\Phi}(\boldsymbol{x})\boldsymbol{\xi}$, where $\boldsymbol{\xi}$ is a zero-mean Gaussian vector with covariance matrix Γ^N.

GP regression with shape constraints consists in finding the conditional distribution of $Y^N = \boldsymbol{\Phi}(\cdot)\boldsymbol{\xi}$ given that

$$\begin{cases} \boldsymbol{y} = \boldsymbol{\Phi}(X) \cdot \boldsymbol{\xi} + \varepsilon \\ \boldsymbol{\xi} \in \mathcal{C}_{ineq} \end{cases}$$

where $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$, $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$, ε is a zero-mean Gaussian noise in $\mathbb{R}^n$ (independent of $\boldsymbol{\xi}$) and $\mathcal{C}_{ineq}$ is a set of linear inequality constraints.

This is equivalent to finding the distribution of a truncated Gaussian vector $Z := [\boldsymbol{\xi} \mid \boldsymbol{y} = \boldsymbol{\Phi}(X) \cdot \boldsymbol{\xi} + \varepsilon]$ given $\boldsymbol{\xi} \in \mathcal{C}_{ineq}$

Estimation of hyper-parameters

- We consider 2-dimensional anisotropic stationary kernels :

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 K_t(T - T'; \theta_t) K_x(k - k'; \theta_x)$$

  where K_t, K_x are stationary kernels. ex : Gaussian, Matérn 5/2, Matérn 3/2, Exponential.

- Homoscedastic noise : $\varepsilon \sim \mathcal{N}(0, \Sigma_{noise})$ where $\Sigma_{noise} = \sigma^2_{noise} \mathbb{I}_n$
- **Hyper-parameters** : $\lambda = (\sigma, \theta_1, \ldots, \theta_d, \sigma_{noise})$

Estimation of hyper-parameters

Following López-Lopera et al (2017), two MLE approaches can be considered

- **Unconditional likelihood** : Find λ that maximizes the Gaussian likelihood $\mathbb{P}\left(\boldsymbol{\Phi}(X)\cdot\boldsymbol{\xi}+\varepsilon=\boldsymbol{y}\mid\lambda\right)$ or log-likelihood

$$\mathcal{L}_N(\lambda):=-\frac{n}{2}\log(2\pi)-\frac{1}{2}\log|C|-\frac{1}{2}\boldsymbol{y}^\top C^{-1}\boldsymbol{y}$$

  where $C:=\boldsymbol{\Phi}(X)\Gamma^N(\lambda)\boldsymbol{\Phi}(X)+\Sigma_{noise}(\lambda)$

- **Conditional likelihood** : Find λ that maximizes the conditional probability $\mathbb{P}\left(\boldsymbol{\Phi}(X)\cdot\boldsymbol{\xi}+\varepsilon=\boldsymbol{y}\mid\boldsymbol{\xi}\in\mathcal{C}_{ineq},\lambda\right)$ or the log-likelihood

$$\mathcal{L}_{N,cond}(\lambda):=\mathcal{L}_N(\lambda)+\log\mathbb{P}(\boldsymbol{\xi}\in\mathcal{C}_{ineq}\mid\boldsymbol{\Phi}(X)\cdot\boldsymbol{\xi}+\varepsilon=\boldsymbol{y})-\log\mathbb{P}(\boldsymbol{\xi}\in\mathcal{C}_{ineq})$$

Mode estimator

We define the (a posteriori) most probable response surface and measurement noises as

$$\begin{cases} M_K^N(\boldsymbol{x}) := \boldsymbol{\Phi}(\boldsymbol{x}) \cdot (c_1^*, \dots, c_N^*)^\top, \ \boldsymbol{x} \in D \\ \boldsymbol{e}^* := (e_1^*, \dots, e_n^*)^\top \end{cases}$$

where $(\boldsymbol{c}^*, \boldsymbol{e}^*)$ is the mode of the truncated Gaussian vector $(\boldsymbol{\xi}, \varepsilon)$ given the constraints, defined as solution of

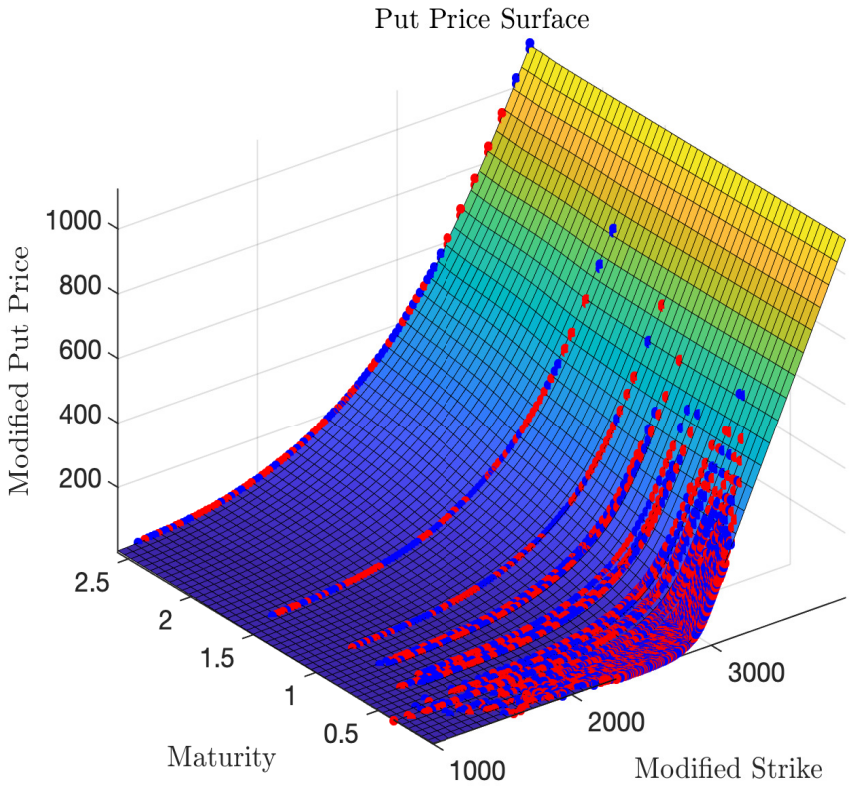
$$\max_{\boldsymbol{c},\boldsymbol{e}} \mathbb{P}\left(\boldsymbol{\xi} \in [\boldsymbol{c}, \boldsymbol{c} + d\boldsymbol{c}], \varepsilon \in [\boldsymbol{e}, \boldsymbol{e} + d\boldsymbol{e}] \mid \boldsymbol{\Phi}(X) \cdot \boldsymbol{\xi} + \varepsilon = \boldsymbol{y}, \ \boldsymbol{\xi} \in \mathcal{C}_{ineq}\right).$$

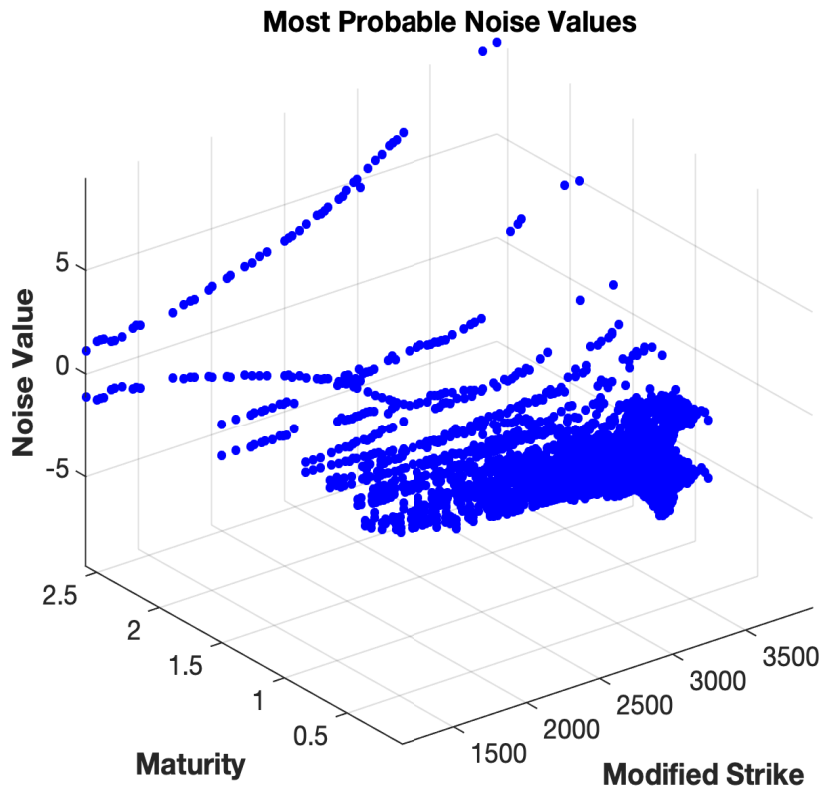
The mode $(\boldsymbol{c}^*, \boldsymbol{e}^*)$ is solution of a quadratic problem

$$\min_{\boldsymbol{\Phi}(X)\cdot\boldsymbol{c}+\boldsymbol{e}=\boldsymbol{y}, \ \boldsymbol{c}\in\mathcal{C}_{ineq}} \left(\boldsymbol{c}^\top (\Gamma^N)^{-1} \boldsymbol{c} + \boldsymbol{e}^\top \Sigma_{noise}^{-1} \boldsymbol{e}\right)$$

Mode estimator

- Data : S&P 500 bid ask put prices as of of May 18, 2019
- Fitted Matérn 5/2 kernel using uncond. MLE, 3340 training data in blue, 1755 testing data in red, $N_t = 25$, $N_x = 100$.
- Most probable surface (left) vs most probable noise values (right)





Sampling finite-dimensional GP with shape constraints

First remark that the distribution of $\boldsymbol{\xi}$ given $\boldsymbol{\Phi}(X) \cdot \boldsymbol{\xi} + \varepsilon = \boldsymbol{y}$ is multinormal $\mathcal{N}(\boldsymbol{\mu}_{cond}, \Sigma_{cond})$ where

$$\begin{cases} \boldsymbol{\mu}_{cond} = \Gamma^N \boldsymbol{\Phi}(X)^\top \left(\boldsymbol{\Phi}(X)\Gamma^N \boldsymbol{\Phi}(X)^\top + \Sigma_{noise}\right)^{-1} \boldsymbol{b} \\ \Sigma_{cond} = \Gamma^N - \Gamma^N \boldsymbol{\Phi}(X)^\top \left(\boldsymbol{\Phi}(X)\Gamma^N \boldsymbol{\Phi}(X)^\top + \Sigma_{noise}\right)^{-1} \boldsymbol{\Phi}(X)\Gamma^N \end{cases}$$

Following López-Lopera et al (2017), we consider the Hamiltonian Monte Carlo method introduced by Pakman and Paninski (2013) for sampling truncated multivariate Gaussian :

$$\mathcal{TN}\left(\boldsymbol{\mu}_{cond}, \Sigma_{cond}, \mathcal{C}_{ineq}\right)$$

MCMC initialized using the mode estimator since it satisfies the inequality constraints.

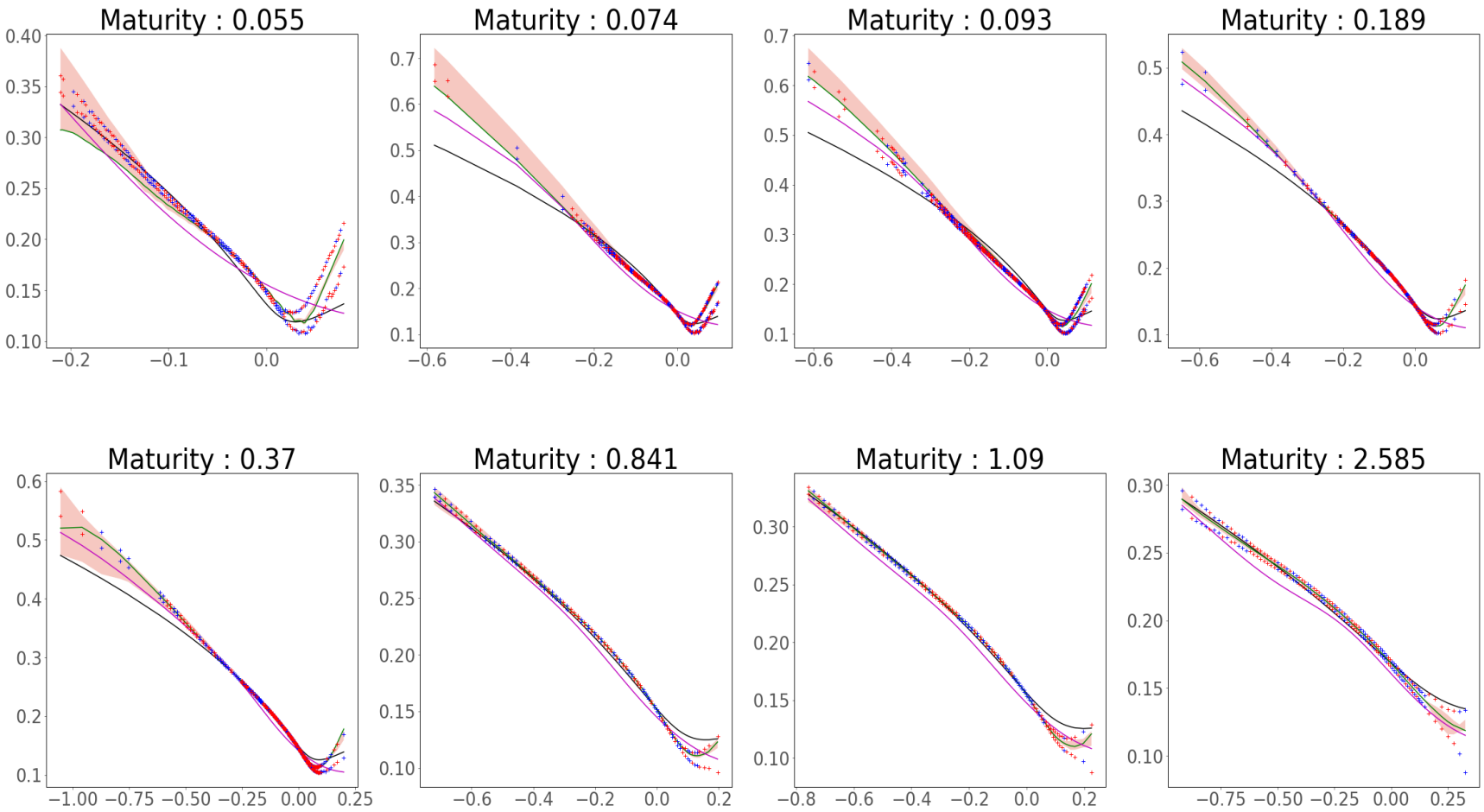
Comparison of RMSE and backtest results for the different approaches

SSVI, GP, NN based on IV, NN based on price, and unconstrained versions

IV RMSE (Price RMSE)	SSVI	GP	IV based NN	Price based NN	SSVI Unconstr.	GP Unconstr.	IV based NN Unconstr.	Price based NN Unconstr.
Calibr. fit on the training set	1.37% (2.574)	0.58% (0.338)	1.23% (2.897)	13.70% (9.851)	1.04% (2.691)	0.60% (0.321)	0.84% (2.163)	5.65 % (2.456)
Calibr. fit on the testing set	1.52% (2.892)	0.57% (0.355)	1.29% (2.966)	14.27% (10.347)	1.09% (2.791)	0.57% (0.477)	0.86% (2.045)	6.14% (2.888)
MC backtest	8.69% (22.826)	19.76% (74.017)	2.95% (4.989)	6.37% (11.764)	N/A	N/A	N/A	N/A
FD backtest	6.88% (33.545)	7.86% (35.270)	3.43% (11.976)	5.56% (26.785)	N/A	N/A	N/A	N/A
Comput. time (seconds)	33	856	191	185	1	16	76	229

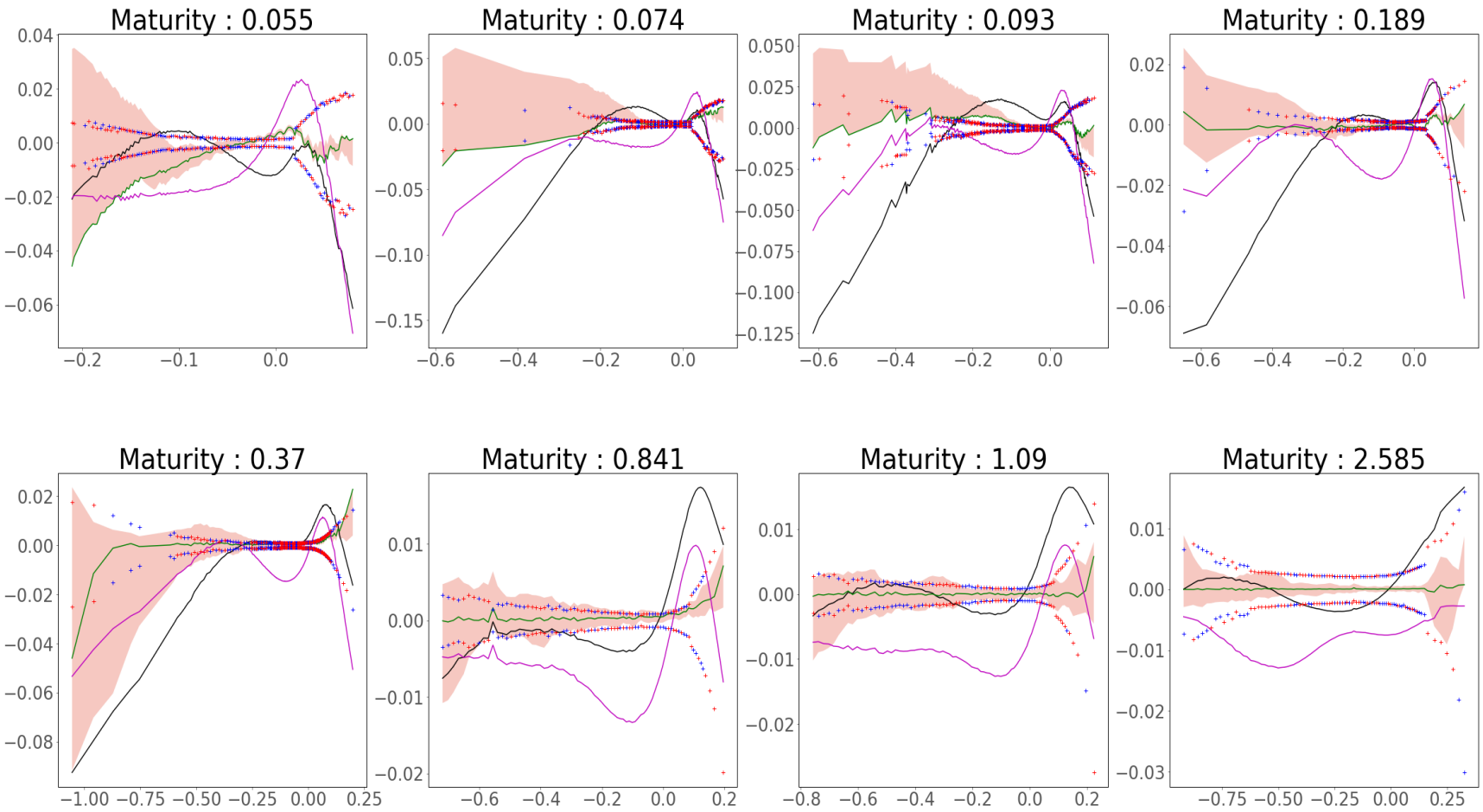
Table: The IV and price RMSEs of the SSVI, GP and NN approaches. Last line : computation times.

Slices of the IV Surface



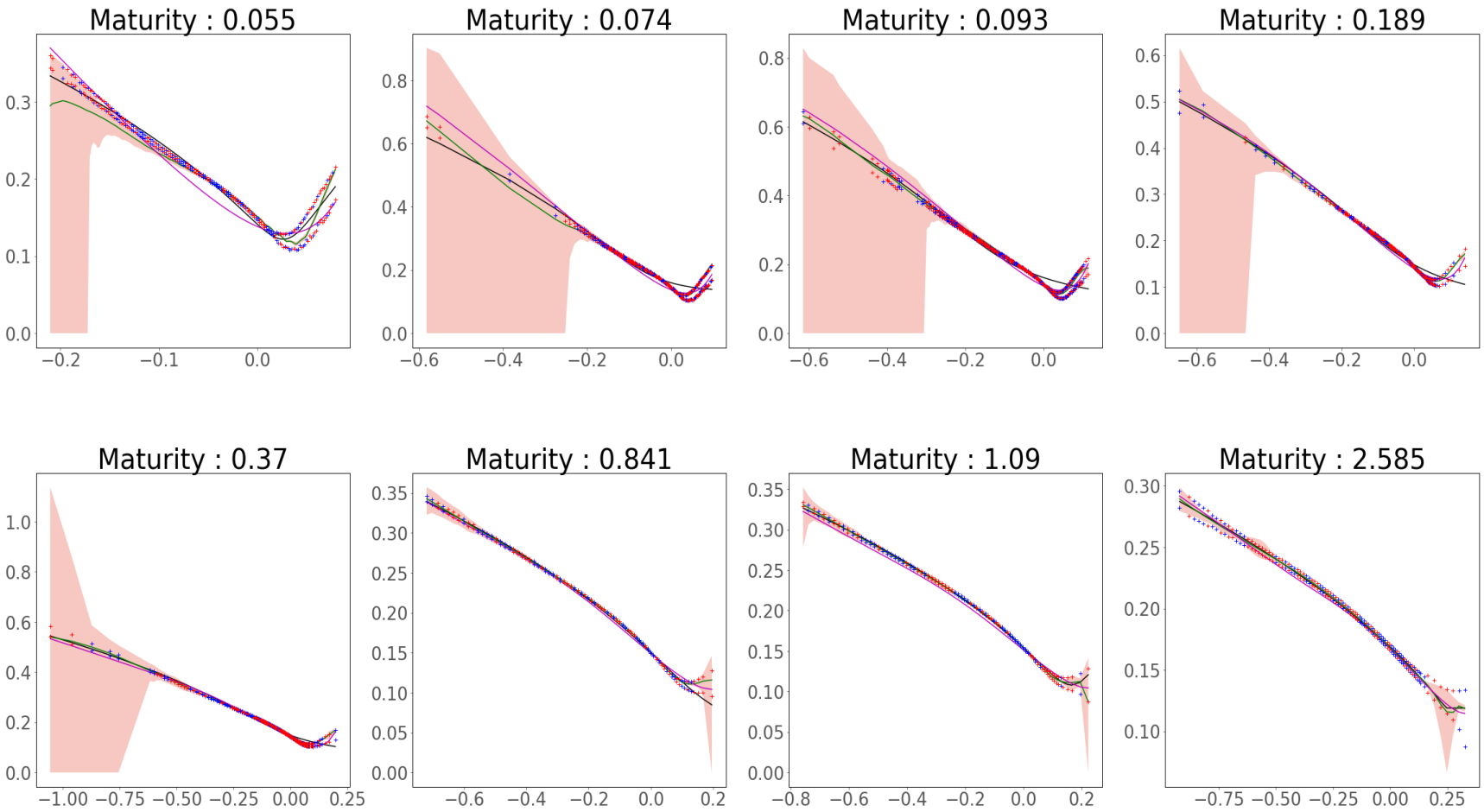
Slices of constrained GP (green), NN (purple), and SSVI (black) models of SPX puts with training bid-asks IVs (+) and testing bid-asks IVs (+) (the bid-ask IVs are reconstructed numerically from the corresponding bid-ask market prices). The shaded envelopes show 100 paths of the constrained GP's posterior.

Slices of fitted IV errors with respect to mid-price IVs



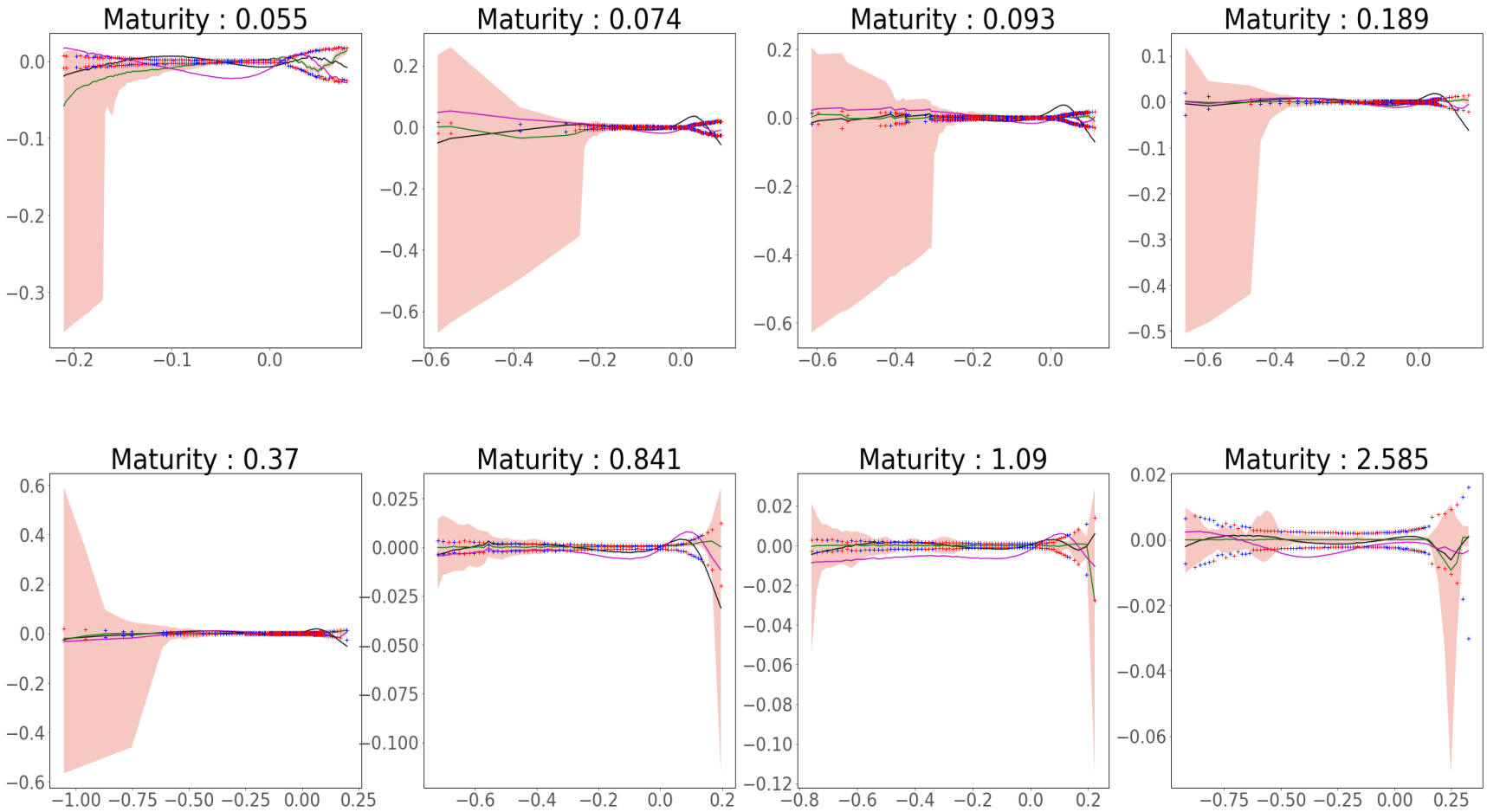
Slices of constrained GP (green), NN (purple), and SSVI (black) models of SPX puts with training bid-asks IVs (+) and testing bid-asks IVs (+) (the bid-ask IVs are reconstructed numerically from the corresponding bid-ask market prices). The shaded envelopes show 100 paths of the constrained GP's posterior.

Slices of the IV Surface - unconstrained case



Slices of constrained GP (green), NN (purple), and SSVI (black) models of SPX puts with training bid-asks IVs (+) and testing bid-asks IVs (+) (the bid-ask IVs are reconstructed numerically from the corresponding bid-ask market prices). The shaded envelopes show 100 paths of the constrained GP's posterior.

Slices of fitted IV errors with respect to mid-price IVs - unconstrained case



Slices of constrained GP (green), NN (purple), and SSVI (black) models of SPX puts with training bid-asks IVs (+) and testing bid-asks IVs (+) (the bid-ask IVs are reconstructed numerically from the corresponding bid-ask market prices). The shaded envelopes show 100 paths of the constrained GP's posterior.

Thanks for your attention.

References I


Asgharian, H., Hess, W., and Liu, L. (2013).
A spatial analysis of international stock market linkages.
Journal of Banking & Finance, 37(12) :4738–4754.


Bay, X., Grammont, L., and Maatouk, H. (2016).
Generalization of the Kimeldorf-Wahba correspondence for constrained interpolation.


Cousin, A., Maatouk, H., and Rullière, D. (2016).
Kriging of financial term-structures.
European J. Oper. Res., 255(2) :631–648.


Cressie, N. (1990).
The origins of kriging.
Math. Geol., 22(3) :239–252.


da Barrosa, M. R., Salles, A. V., and Ribeiro, C. d. O. (2016).
Portfolio optimization through kriging methods.
Applied Economics, 48(50) :4894–4905.


De Spiegeleer, J., Madan, D. B., Reyners, S., and Schoutens, W. (2018).
Machine learning for quantitative finance : fast derivative pricing, hedging and fitting.
Quantitative Finance, 18(10) :1635–1643.


Dixon, M. F. and Crépey, S. (2018).
Multivariate gaussian process regression for derivative portfolio modeling : Application to cva.

References II


Liu, M. and Staum, J. (2010).
Stochastic kriging for efficient nested simulation of expected shortfall.
Journal of Risk, 12(3) :3.


López-Lopera, A. F., Bachoc, F., Durrande, N., and Roustant, O. (2018).
Finite-dimensional gaussian approximation with linear inequality constraints.
SIAM/ASA Journal on Uncertainty Quantification, 6(3) :1224–1255.


Ludkovski, M. (2018).
Kriging metamodels and experimental design for bermudan option pricing.
Journal of Computational Finance, 22(1) :37–77.


Ludkovski, M. and Risk, J. (2018).
Sequential design and spatial modeling for portfolio tail risk measurement.
SIAM Journal of Financial Mathematics.


Ludkovski, M., Risk, J., and Zail, H. (2018).
Gaussian process models for mortality rates and improvement factors.
ASTIN Bulletin : The Journal of the IAA, 48(3) :1307–1347.


Maatouk, H. and Bay, X. (2014).
Gaussian Process Emulators for Computer Experiments with Inequality Constraints.
in revision SIAM/ASA J. Uncertainty Quantification.


Pakman, A. and Paninski, L. (2014).
Exact hamiltonian monte carlo for truncated multivariate gaussians.
Journal of Computational and Graphical Statistics, 23(2) :518–542.

References III


Roberts, S., Osborne, M., Ebden, M., Reece, S., Gibson, N., and Aigrain, S. (2013).
Gaussian processes for time-series modelling.
Philosophical Transactions of the Royal Society A : Mathematical, Physical and Engineering Sciences, 371(1984) :20110550.

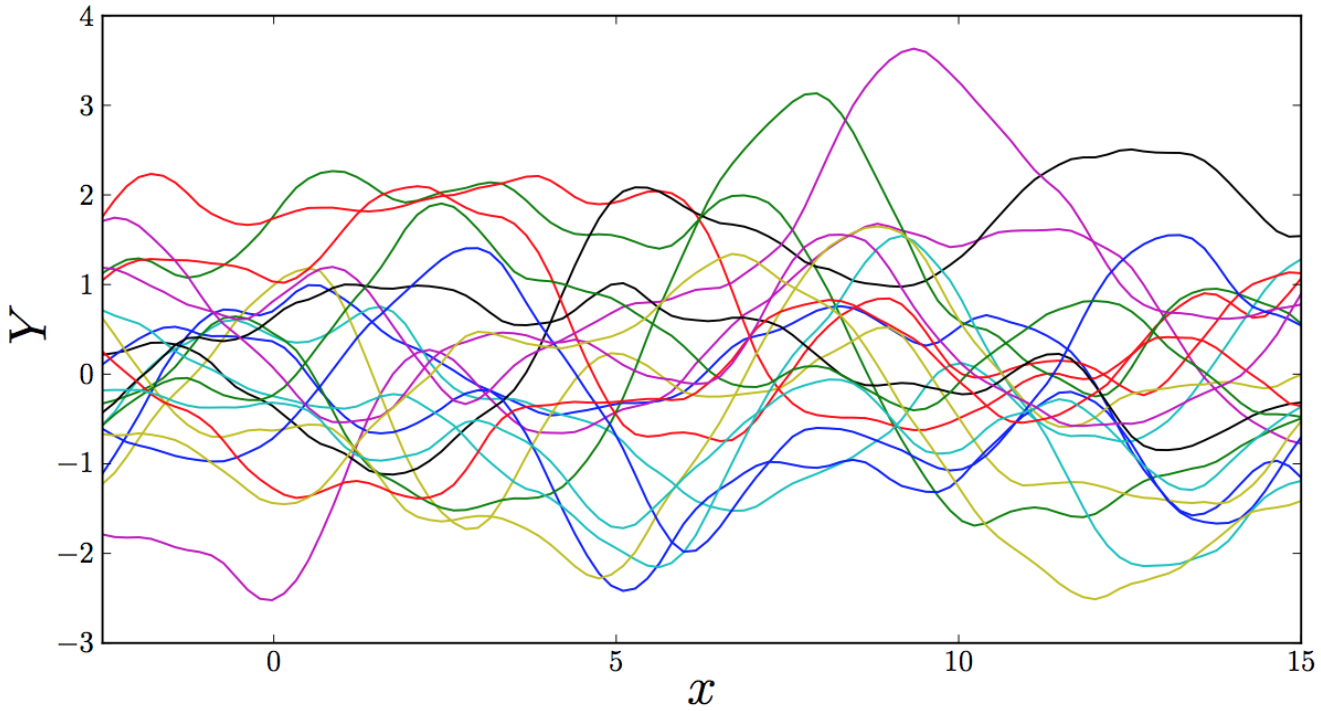

Wahba, G. (1990).
Spline models for observational data, volume 59.
Siam.


Williams, C. K. and Rasmussen, C. E. (2006).
Gaussian processes for machine learning.
the MIT Press, 2(3) :4.

Classical kriging

Estimation of the unknown function f using Bayesian statistics

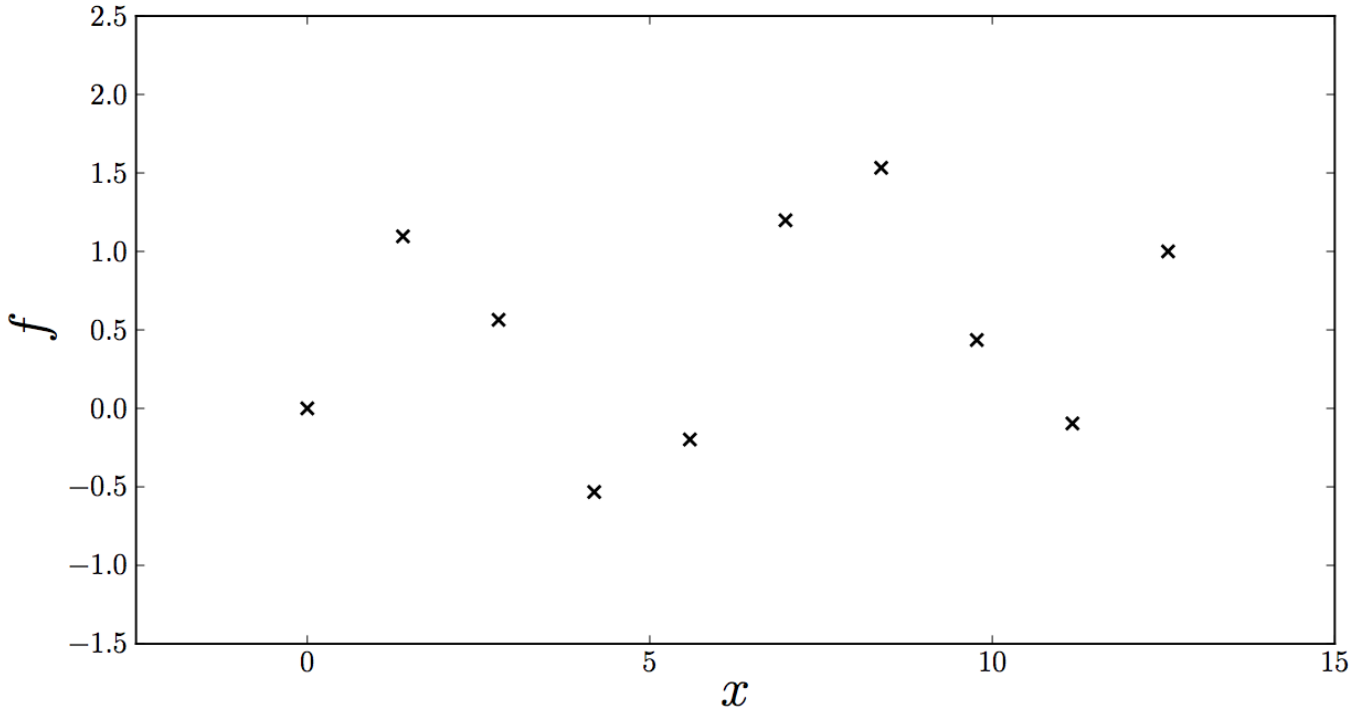
Our first belief in f is given as a Gaussian process prior Y



Classical kriging

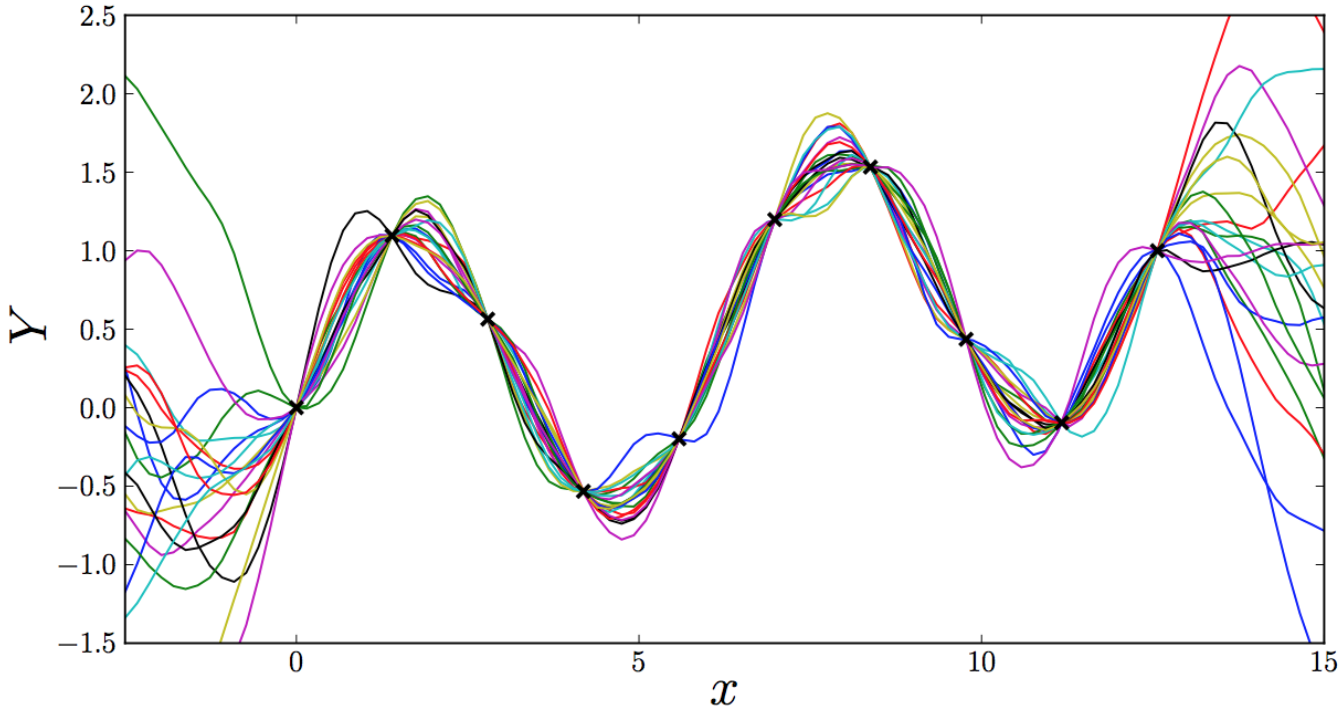
The function f is known at some input points $x^1, \ldots, x^n$:

$$f\left(x^1\right) = y^1, \ldots, f\left(x^n\right) = y^n.$$



Classical kriging

This belief is updated given that $Y(x_1) = y_1, \ldots, Y(x_n) = y_n$



Source : presentation of N. Durrande

Classical kriging

Definition : Gaussian process (GP) or Gaussian random field

A Gaussian process is a collection of random variables, any finite number of which have (consistent) joint Gaussian distributions.

A Gaussian process $\big(Y(x), x \in \mathbb{R}^d\big)$ is characterized by its mean function

$$\mu : x \in \mathbb{R}^d \longrightarrow \mathsf{E}(Y(x)) \in \mathbb{R}.$$

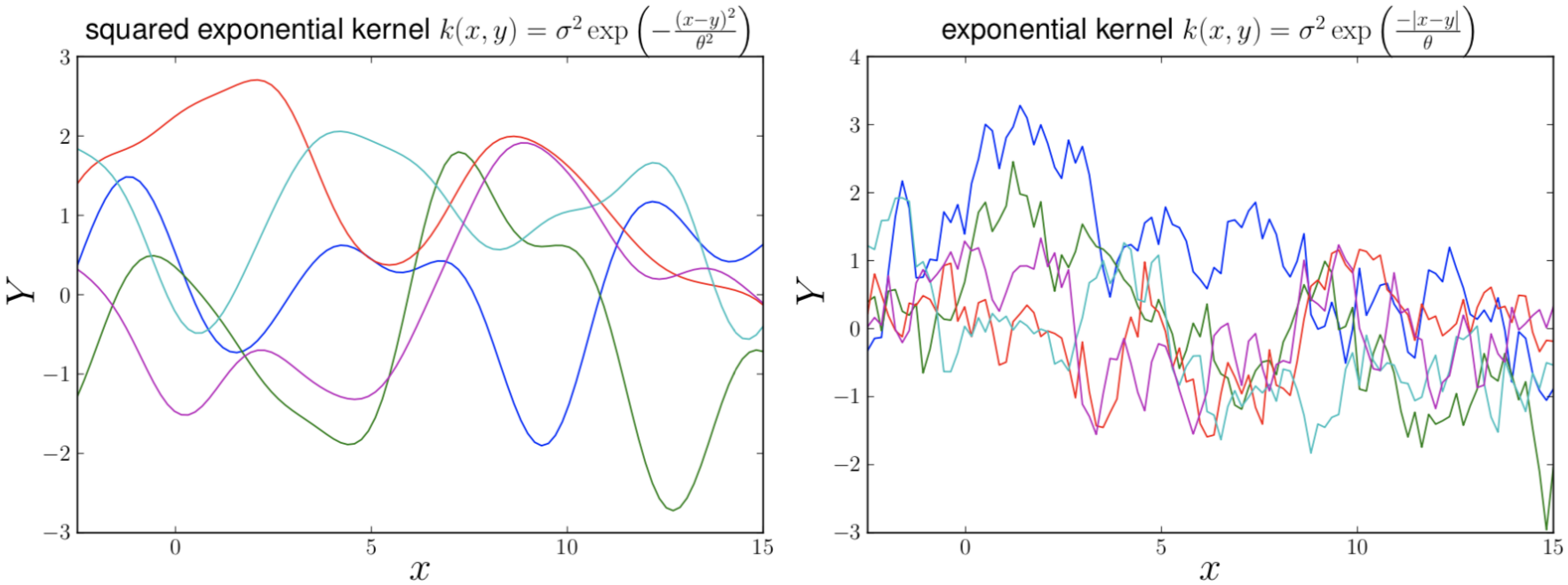
and its covariance function

$$K : (x, x') \in \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathrm{Cov}(Y(x), Y(x')) \in \mathbb{R}.$$

1D kriging kernel	$K(x, x')$	Class
Gaussian	$\sigma^2 \exp\left(-\frac{(x-x')^2}{2\theta^2}\right)$	$\mathcal{C}^\infty$
Matérn 5/2	$\sigma^2 \left(1 + \frac{\sqrt{5}\lvert x-x'\rvert}{\theta} + \frac{5(x-x')^2}{3\theta^2}\right) \exp\left(-\frac{\sqrt{5}\lvert x-x'\rvert}{\theta}\right)$	$\mathcal{C}^2$
Matérn 3/2	$\sigma^2 \left(1 + \frac{\sqrt{3}\lvert x-x'\rvert}{\theta}\right) \exp\left(-\frac{\sqrt{3}\lvert x-x'\rvert}{\theta}\right)$	$\mathcal{C}^1$
Exponential	$\sigma^2 \exp\left(-\frac{\lvert x-x'\rvert}{\theta}\right)$	$\mathcal{C}^0$

Classical kriging

Changing the kernel K means changing the initial belief on f (i.e., the prior).

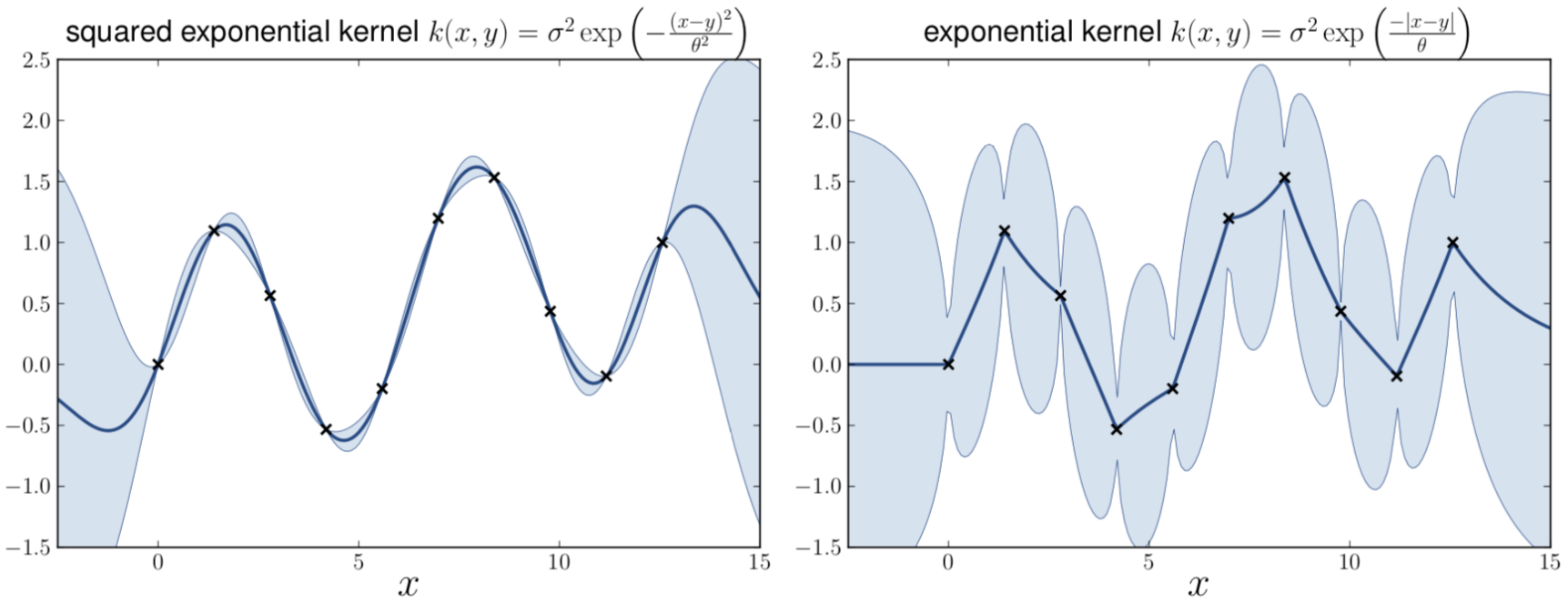


Source : presentation of N. Durrande

Given the observations, the model is entirely defined by the kernel.

Classical kriging

Changing the kernel K has a huge impact on the model



Source : presentation of N. Durrande

Classical kriging

- $\boldsymbol{X} = (x_1, \ldots, x_n)^\top \in \mathbb{R}^{n \times d}$: some design points
- $\boldsymbol{y} = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$: observed values of f at these points
- $Y(\boldsymbol{X}) = (Y(x_1), \ldots, Y(x_n))^\top$: vector composed of Y at point $\boldsymbol{X}$

The conditional process is still a Gaussian Process

Let Y be a GP with mean μ and covariance function K. The conditional process $Y \mid Y(\boldsymbol{X}) = \boldsymbol{y}$ is a GP with mean function

$$\eta(x) = \mu(x) + \boldsymbol{k}(x)^\top \mathbb{K}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}), \quad x \in \mathbb{R}^d$$

and covariance function $\tilde{K}$ given by

$$\tilde{K}(x, x') = K(x, x') - \boldsymbol{k}(x)^\top \mathbb{K}^{-1} \boldsymbol{k}(x'), \quad x, x' \in \mathbb{R}^d$$

where $\boldsymbol{\mu} = \mu(X) = (\mu(x_1), \ldots, \mu(x_n))^\top$, $\mathbb{K}$ is the covariance matrix of $Y(\boldsymbol{X})$ and $\boldsymbol{k}(x) = (K(x, x_1), \ldots, K(x, x_n))^\top$

Note that computational complexity of $\mathbb{K}^{-1}$ is $O(n^3)$.

Classical kriging - indirect observations

In some applications, the unknown function f is not directly observed at X. But, if $f(X)$ is known up to solving a linear equality system, kriging can still be applied without loss of efficiency :

$$A \cdot f(X) = \boldsymbol{b}, \tag{1}$$

where

- A is a given matrix of dimension $n \times m$
- $\boldsymbol{b} = (b_1, \ldots, b_n)^\top \in \mathbb{R}^n$
- $X = (x_1, \ldots, x_m)^\top \in \mathbb{R}^{m \times d}$: some design points
- $f(X) = (f(x_1), \ldots, f(x_m))^\top \in \mathbb{R}^m$

Classical kriging - indirect observations

The GP prior Y is updated given $AY(X) = \boldsymbol{b}$ where

- $Y(X) = (Y(x_1), \ldots, Y(x_m))$: vector composed of Y at point $\boldsymbol{X}$

The conditional process is still a Gaussian Process

Let Y be a GP with mean μ and covariance function K. The conditional process $Y \mid AY(X) = \boldsymbol{b}$ is a GP with mean function

$$\eta(x) = \mu(x) + (A\boldsymbol{k}(x))^\top \left(A\mathbb{K}A^\top\right)^{-1} (\boldsymbol{b} - A\boldsymbol{\mu}), \quad x \in \mathbb{R}^d$$

and covariance function $\tilde{K}$ given by

$$\tilde{K}(x, x') = K(x, x') - (A\boldsymbol{k}(x))^\top \left(A\mathbb{K}A^\top\right)^{-1} A\boldsymbol{k}(x'), \quad x, x' \in \mathbb{R}^d$$

where $\boldsymbol{\mu} = \mu(X) = (\mu(x_1), \ldots, \mu(x_m))^\top$, $\mathbb{K}$ is the covariance matrix of $Y(X)$, $\boldsymbol{k}(x) = (K(x, x_1), \ldots, K(x, x_m))^\top$

Classical kriging - indirect observations with noise

Assume that f is known up to solving a linear equality system with measurement errors :

$$A \cdot f(X) + \varepsilon = \boldsymbol{b}. \tag{2}$$

where

- A is a given matrix of dimension $n \times m$
- $\boldsymbol{b} = (b_1, \ldots, b_n)^\top \in \mathbb{R}^n$
- $X = (x_1, \ldots, x_m)^\top \in \mathbb{R}^{m \times d}$: some design points
- $f(X) = (f(x_1), \ldots, f(x_m))^\top \in \mathbb{R}^m$
- ε is zero-mean Gaussian noise in $\mathbb{R}^n$ with covariance matrix Σ_{noise}

Note that A is not necessarily a full-rank matrix in the presence of noise

Classical kriging - indirect observations with noise

The GP prior Y is updated given $AY(X) + \varepsilon = \boldsymbol{b}$ where

- $Y(X) = (Y(x_1), \ldots, Y(x_m))$: vector composed of Y at point $\boldsymbol{X}$
- ε is assumed to be independent of Y

The conditional process is still a Gaussian Process

Let Y be a GP with mean μ and covariance function K. The conditional process $Y \mid AY(X) + \varepsilon = \boldsymbol{b}$ is a GP with mean function

$$\eta(x) = \mu(x) + (A\boldsymbol{k}(x))^\top \left(A\mathbb{K}A^\top + \Sigma_{noise}\right)^{-1} (\boldsymbol{b} - A\boldsymbol{\mu}), \quad x \in \mathbb{R}^d$$

and covariance function $\tilde{K}$ given by

$$\tilde{K}(x, x') = K(x, x') - (A\boldsymbol{k}(x))^\top \left(A\mathbb{K}A^\top + \Sigma_{noise}\right)^{-1} A\boldsymbol{k}(x'), \quad x, x' \in \mathbb{R}^d$$

where $\boldsymbol{\mu} = \mu(X) = (\mu(x_1), \ldots, \mu(x_m))^\top$, $\mathbb{K}$ is the covariance matrix of $Y(X)$, $\boldsymbol{k}(x) = (K(x, x_1), \ldots, K(x, x_m))^\top$

Training a Gaussian process

- In general, we do not have enough information to define - a priori - the mean function and the kernel function
- Training the GP consists in selecting the kernel function (and the corresponding hyper-parameters) that best represents the data structure
- Hyper-parameters : $\mathbf{p} = (\mu(\cdot), \sigma, \theta, \Sigma)$
- Estimation of $\mathbf{p}$ by maximizing the likelihood function
- As $Y(X)$ is a Gaussian verctor with mean $\boldsymbol{\mu}$ and covariance matrix $\mathbb{K}$, the log likelihood function is given by :

$$\log \mathbb{P}(Y(X) = \boldsymbol{y} \mid \mathbf{p}) = -\frac{1}{2}\log\det(\mathbb{K}) - \frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})^{\top}\mathbb{K}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}) - \frac{n}{2}\log(2\pi)$$