Bayesian Finance

Josef Teichmann

Luminy 2018

Goal of this talk ...

- introduce a very general setting for stochastic models of financial markets with a Bayesian flavor (joint work with Christa Cuchiero and Irene Klein).
- introduce a machine learning setting where such models can be numerically evaluated (joint work with Hans Bühler, Lukas Gonon, Ben Wood).
- show alternative parametrizations of the problem in the spirit of reservoir computing.

Goal of this talk ...

- introduce a very general setting for stochastic models of financial markets with a Bayesian flavor (joint work with Christa Cuchiero and Irene Klein).
- introduce a machine learning setting where such models can be numerically evaluated (joint work with Hans Bühler, Lukas Gonon, Ben Wood).
- show alternative parametrizations of the problem in the spirit of reservoir computing.

Goal of this talk ...

- introduce a very general setting for stochastic models of financial markets with a Bayesian flavor (joint work with Christa Cuchiero and Irene Klein).
- introduce a machine learning setting where such models can be numerically evaluated (joint work with Hans Bühler, Lukas Gonon, Ben Wood).
- show alternative parametrizations of the problem in the spirit of reservoir computing.

Platonic financial market

Often accepted hypothesis in modeling financial markets:

Observations of prices are perfect and can be immediately included in trading decisions.

We consider here a setting where this hypothesis is not true anymore.

Platonic stock exchange: consider a financial market, where the prices of the assets are not fully revealed to the trader (like ideas in Platon's cave allegory) or where trading decisions are not immediately excuted. How to invest optimally?

Platonic financial market

A *platonic financial market* is given by a stochastic basis together with two filtrations $\mathbb{F} \subset \mathbb{G}$ and a family of càdlàg stochastic processes S adapted to $\mathbb{G}$. Almost sure statements are understood with respect to the larger filtration $\mathbb{G}$.

Trading in these assets is possible but only with $\mathbb{F}$-predictable strategies (made precise later).

Analyzing platonic financial markets we call *Bayesian Finance* since optional projections will play a key role such as in (Bayesian) Filtering, however, with respect to pricing measures.

Examples

- Trading with an execution delay.
- Trading with observational delay.
- Trading one underlying to hedge risks of another underlying.
- For some assets trading is only possible for restricted time sets (for example static trading)
- Prices are uncertain due to liquidity issues, transaction costs.
- Model prices differ from market prices, which means in principle that one believes market prices come with an error.
- Lack of information (e.g. discrete information versus continuous time modeling).
- quantify frictions in stochastic portfolio theory
- Model uncertainty.

References

- Youri Kabanov – Christophe Stricker: *The Dalang-Morton-Willinger theorem under delayed and restricted information* (2006).
- Moritz Dümbgen – Chris Rogers: *Estimate nothing* (2014).

The Kabanov-Stricker theorem: discrete time

Here we consider only a finite number of adapted processes $S^1, \ldots S^n$ on $(\Omega, \mathcal{G}, (\mathcal{G}_t)_{t=0}^T, P)$. Trading is allowed only with restricted information. To be precise, let $(\mathcal{F}_t)$, $t = 1, \ldots, T$, be a filtration such that $\mathcal{F}_t \subset \mathcal{G}_t$ for all t. For an $\mathbb{R}^n$-valued, $(\mathcal{F}_t)$ simply predictable process $\mathbf{H} = (H^1, \ldots, H^n)$

$$(\mathbf{H} \cdot \mathbf{S})_t = \sum_{k=1}^{n} \sum_{u=1}^{t} H_{u-1}^k (S_u^k - S_{u-1}^k)\,.$$

Possible portfolio processes:
$\mathcal{X} = \{X = (\mathbf{H} \cdot \mathbf{S}) : \mathbf{H}\ \mathbb{R}^n\text{-valued, } \mathcal{F}\text{-predictable}\}$ and $K_0 = \{X_T, X \in \mathcal{X}\}$.

Youri Kabanov and Christophe Stricker from 2006:

Theorem

The following conditions are equivalent:

(i) *NA, that means* $K_0 \cap L^0_+(\Omega, \mathcal{G}, P) = \{0\}$.

(ii) *there exists a probability measure* Q *with* $\frac{dQ}{dP} \in L^\infty(\Omega, \mathcal{G}, P)$ *such that*

$$E_Q[S^k_{t+1} | \mathcal{F}_t] = E_Q[S^k_t | \mathcal{F}_t],$$

for all $k = 1, \dots, n$, $t = 0, \dots T - 1$.

(iii) $\big(K_0 - L^0_+(\Omega, \mathcal{G}, P)\big) \cap L^0_+(\Omega, \mathcal{G}, P) = \{0\}$

FTAP in continuous time: setting

We consider a large financial market model in continuous time in the following way. Let I be a parameter space which can be any set, countable or uncountable.

Let $T = 1$ denote a finite time horizon and let $(\Omega, \mathcal{G}, P)$ be a probability space with a filtration $\mathbb{G} = (\mathcal{G}_t)_{t\in[0,1]}$. On this probability space we are given a family of $\mathbb{G}$-adapted stochastic processes $(S_t^i)_{t\in[0,1]}$, $i \in I$.

Notice that we do not assume any path properties nor semi-martingality properties.

Setting

We define, for each $n \geq 1$, a family $\mathcal{A}^n$ of eligible subsets of I, which contain exactly n elements:

$$\mathcal{A}^n = \{\text{all/some subsets } A \subseteq I, \text{ such that } |A| = n\}, \tag{1}$$

where $|A|$ denotes the cardinality of the set A. Furthermore we consider a family of filtrations $\mathbb{F}^A = (\mathcal{F}^A_t)_{t\in[0,1]}$, indexed by $A \in \mathcal{A}$ in $\mathbb{G}$. We assume that

$\mathcal{F}^A$ is refining and monotone

- for $A^1, A^2 \in \bigcup_{n\geq 1} \mathcal{A}^n$, then $A^1 \cup A^2 \in \bigcup_{n\geq 1} \mathcal{A}^n$.
- for two sets $A^1, A^2 \in \bigcup_{n\geq 1} \mathcal{A}^n$, such that $A^1 \subseteq A^2$, then we have that $\mathbb{F}^{A^1} \subseteq \mathbb{F}^{A^2}$

Setting

For each $A \in \bigcup_{n \geq 1} \mathcal{A}^n$ we define the set of portfolio wealth processes $\mathcal{X}^A$ based on simple strategies for deterministic time points in the *small financial market A* that are predictable with respect to the smaller filtration $\mathbb{F}^A = (\mathcal{F}_t^A)_{t \in [0,1]}$. To be precise the set of simple portfolio wealth processes for the small market given by $A = \{\alpha_1, \ldots, \alpha_n\} \in \mathcal{A}^n$ is defined as

$$\mathcal{X}^A = \left\{ (\mathbf{H}^A \cdot \mathbf{S}^A)_t, t \in [0,1] : \mathbf{H}^A \text{ is } \mathbb{F}^A\text{-predictable and simple} \right\}.$$

Setting

We define the set $\mathcal{X}^n$ of all portfolio wealth processes with respect to simple strategies that include at most n assets (but all possible different choices of n assets). Indeed, for each $n \geq 1$ we consider the following set $\mathcal{X}^n$

$$\mathcal{X}^n = \bigcup_{A \in \mathcal{A}^n} \mathcal{X}^A. \tag{2}$$

Note that the sets $\mathcal{X}^n$ are neither convex nor do they satisfy a concatenation property in the sense of, e.g., Kabanov (1997), because in both cases $2n$ assets could be involved in the combinations.

We introduce the convex sets of (simple) portfolio wealth processes and its terminal evaluation:

Wealth processes with restricted information

1. Define the set of all wealth processes defined on simple strategies involving a finite number of assets in the large financial market as $\mathcal{X} = \bigcup_{n \geq 1} \mathcal{X}^n$.
2. We denote by $K_0 = \{X_1 : X \in \mathcal{X}\}$ the evaluations of elements of $\mathcal{X}$ at terminal time $T = 1$.

Remark

Our setting is general and realistic. There are no path requirements on the involved processes. Every trading situation might, e.g., involve a degree of delay as well as an amount of market frictions as well as specific information available. Moreover we can have a different set of possible trading times for the assets (static trading in some assets).

Let C be the convex cone of all superreplicable claims (by simple strategies) in the large financial market, that is,

$$C = K_0 - L^0_{\geq 0}(\mathcal{G}, P).$$

Note also that the above setting includes as examples the large financial market based on a sequence of assets as well as bond markets (with a continuum of assets).

NAFL condition

We use an L^p assumption as in Stricker (1990) (in the setting of one filtration and small markets). But only for some $P' \sim P$.

No Arbitrage Criterion with Integrability Assumption

Let $P' \sim P$ and $p \geq 1$. We assume that $S_t^i \in L^p(\Omega, \mathcal{G}, P')$, for all $i \in I$, $t \in [0,1]$ for some fixed $p \geq 1$.

Let $C_p(P') = C \cap L^p(\mathcal{G}, P')$. We say that the large financial market satisfies the condition *no asymptotic L^p-free lunch for P'* (NAFLp(P')) if the following holds:

$$\overline{C_p(P')} \cap L^p_{\geq 0}(\mathcal{G}, P') = \{0\}.$$

(By the integrability assumptions it is obvious that $C_p(P') \neq \emptyset$.)

Dual set

Fix $1 \leq q \leq \infty$ dual to p. We define the set of $L^q(P')$ probability measures such that the Q-optional projection of the process $(\mathbf{S}_t^A)$ for the filtration $\mathbb{F}^A$ is a Q-martingale, for all finite subsets A of I, as follows:

$$\mathcal{M}^q(P') = \{Q \sim P, \frac{dQ}{dP'} \in L^q(\mathcal{G}, P') : E_Q[S_t^{\alpha_i}|\mathcal{F}_u^A] = E_Q[S_u^{\alpha_i}|\mathcal{F}_u^A] \text{ a.s.}, \\ \text{for all } A = \{\alpha_1, \ldots, \alpha_l\} \in \bigcup_{n \geq 1} \mathcal{A}^n, 1 \leq i \leq l \text{ and all } u \leq t \in [0,1]\} \quad (3)$$

Note that always $\mathcal{M}^q(P') \subseteq \mathcal{M}^1(P)$ for the original measure P.

Admissibility

We emphasize that we do not assume any admissibility for our portfolio value processes, instead we assume $L^p(P')$-integrability with respect to some measure P' which is equivalent with respect to the physical measure P.

Main Theorem

Theorem

The following statements are equivalent:

1. *There exists* $p \geq 1$ *and* $P' \sim P$ *such that NAFLp(*P'*) holds*
2. *The set of equivalent martingale measures* Q*, such that optional projections are martingales, is not empty,* $\mathcal{M}^1(P) \neq \emptyset$.

Remarks:

- It is of great help that we do not need a stochastic integral in this case, which is still not fully available.
- By the equivalence of (1) and (2) we see that our NAFL condition does not depend on the choice of P. In the case of bounded S (2) shows that it is equivalent to NAFLVR (NFLVR).

Further Results

- Functional analytic refinements which allow to express NAFLp(P') differently.
- Super-replication, utility optimization, risk minimization, ...
- Frictions like trading delays are included, trading constraints, transaction costs or market impacts can also be easily included in this almost discrete continuous time setting.

Further Results

- Functional analytic refinements which allow to express NAFLp(P') differently.
- Super-replication, utility optimization, risk minimization, ...
- Frictions like trading delays are included, trading constraints, transaction costs or market impacts can also be easily included in this almost discrete continuous time setting.

Further Results

- Functional analytic refinements which allow to express NAFLp(P') differently.
- Super-replication, utility optimization, risk minimization, ...
- Frictions like trading delays are included, trading constraints, transaction costs or market impacts can also be easily included in this almost discrete continuous time setting.

Immediate Applications

- Non semi-martingale models can be used for prices, since the semi-martingale property does *NOT* follow from NAFLp.
- Prior information in the Bayesian sense on sets of prior measures in the sense of robust finance can be included.
- Dynamic changes of prior information can be made visible in the modeling setup.

Evaluate 2Filtration Setting by Machine Learning methods

- The 2Filtration Setting is numerically delicate, in particular in the presence of non-linear frictions.
- neither PDE methods nor dynamic programming is usually available.
- still machine learning methods allow to implement stochastic optimization problems efficiently even in absence of dynamic programming principles.

In the sequel we introduce a 2Filtration stochastic optimization problem with transaction costs and trading constraints and demonstrate that one can obtain satisfying solutions by machine learning techniques.

Evaluate 2Filtration Setting by Machine Learning methods

- The 2Filtration Setting is numerically delicate, in particular in the presence of non-linear frictions.
- neither PDE methods nor dynamic programming is usually available.
- still machine learning methods allow to implement stochastic optimization problems efficiently even in absence of dynamic programming principles.

In the sequel we introduce a 2Filtration stochastic optimization problem with transaction costs and trading constraints and demonstrate that one can obtain satisfying solutions by machine learning techniques.

Evaluate 2Filtration Setting by Machine Learning methods

- The 2Filtration Setting is numerically delicate, in particular in the presence of non-linear frictions.
- neither PDE methods nor dynamic programming is usually available.
- still machine learning methods allow to implement stochastic optimization problems efficiently even in absence of dynamic programming principles.

In the sequel we introduce a 2Filtration stochastic optimization problem with transaction costs and trading constraints and demonstrate that one can obtain satisfying solutions by machine learning techniques.

Discrete-time market model with frictions

- Trading: at time points $t_0 = 0 < t_1 < \ldots < t_n = T$.
- Prices of hedging instruments: stochastic process $(S_{t_k})_{k=0,\ldots,n}$ in $\mathbb{R}^d$.
- Work on a (finite) probability space $(\Omega, \mathcal{G}, \mathbb{P})$ with filtration $\mathbb{G} = (\mathcal{G}_{t_k})_{k=0,\ldots,n}$, for simplicity $\mathcal{G}_{t_k} = \sigma(S_{t_0}, \ldots, S_{t_k})$.
- At $t = 0$ sell a contingent claim with (random) payoff Z at $T > 0$.
- Specify a (smaller) filtration $\mathbb{F} \subset \mathbb{G}$ for hedging.
- Charging price p_0 and hedging according to an $\mathbb{F}$-predictable strategy δ, terminal profit and loss is (with $\cdot$ discrete-time stochastic integration)

$$\mathrm{PL}_T(Z, p_0, \delta) := -Z + \underbrace{p_0}_{\text{price}} + \underbrace{(\delta \cdot S)_T}_{\text{trading gains}} - \underbrace{C_T(\delta)}_{\text{cum. transaction costs}} .$$

Setup and problem formulation in detail

$$\mathrm{PL}_T(Z, p_0, \delta) := -Z + \underbrace{p_0}_{\text{price}} + \underbrace{(\delta \cdot S)_T}_{\text{trading gains}} - \underbrace{C_T(\delta)}_{\text{cum. transaction costs}} . \quad (4)$$

(4) in more detail:

- $(\delta \cdot S)_T = \sum_{k=1}^{n} \delta_{t_k} \cdot (S_{t_k} - S_{t_{k-1}})$.
- $C_T(\delta) = \sum_{k=0}^{n} c_k(\delta_{t_k} - \delta_{t_{k-1}}, S_{t_0}, \ldots, S_{t_k})$ with $\delta_{t_{-1}} := 0, \delta_{t_n} := 0$.
- Example: transaction costs proportional to transaction amount, i.e. $c_k(\delta_{t_k} - \delta_{t_{k-1}}, S_{t_0}, \ldots, S_{t_k}) = \sum_{i=1}^{d} \varepsilon_i |\delta^i_{t_k} - \delta^i_{t_{k-1}}| S^i_{t_k}$.
- Note: $\mathrm{PL}_T(Z, p_0, \delta) \geq 0$ represents a gain for seller.

Setup and problem formulation in detail

$$\mathrm{PL}_T(Z, p_0, \delta) := -Z + \underbrace{p_0}_{\text{price}} + \underbrace{(\delta \cdot S)_T}_{\text{trading gains}} - \underbrace{C_T(\delta)}_{\text{cum. transaction costs}} . \qquad (4)$$

(4) in more detail:

- $(\delta \cdot S)_T = \sum_{k=1}^{n} \delta_{t_k} \cdot (S_{t_k} - S_{t_{k-1}})$.
- $C_T(\delta) = \sum_{k=0}^{n} c_k(\delta_{t_k} - \delta_{t_{k-1}}, S_{t_0}, \ldots, S_{t_k})$ with $\delta_{t_{-1}} := 0, \delta_{t_n} := 0$.
- Example: transaction costs proportional to transaction amount, i.e. $c_k(\delta_{t_k} - \delta_{t_{k-1}}, S_{t_0}, \ldots, S_{t_k}) = \sum_{i=1}^{d} \varepsilon_i |\delta^i_{t_k} - \delta^i_{t_{k-1}}| S^i_{t_k}$.
- Note: $\mathrm{PL}_T(Z, p_0, \delta) \geq 0$ represents a gain for seller.

Indifference pricing and optimal hedging:

- Describe risk-preferences by a convex risk-measure ρ.
- Denote $\mathcal{H}$ the set of available hedging strategies.
- The indifference price is the (unique) solution $p(Z)$ to

$$\inf_{\delta \in \mathcal{H}} \rho \left(\mathrm{PL}_T(Z, p(Z), \delta) \right) = \inf_{\delta \in \mathcal{H}} \rho \left(\mathrm{PL}_T(0, 0, \delta) \right). \tag{5}$$

- Optimal hedging strategy is minimizer δ^* (if it exists) in left-hand-side of (5).

Numerical calculation of $p(Z)$ and δ^*:

- Highly challenging by classical numerical techniques (very high-dimensional problem) in particular in the 2Filtration setting.
- $\rightarrow$ in practice more simplistic models are used (parametric, continuous-time, no transaction costs).

Indifference pricing and optimal hedging:

- Describe risk-preferences by a convex risk-measure ρ.
- Denote $\mathcal{H}$ the set of available hedging strategies.
- The indifference price is the (unique) solution $p(Z)$ to

$$\inf_{\delta \in \mathcal{H}} \rho\left(\mathrm{PL}_T(Z, p(Z), \delta)\right) = \inf_{\delta \in \mathcal{H}} \rho\left(\mathrm{PL}_T(0, 0, \delta)\right). \qquad (5)$$

- Optimal hedging strategy is minimizer δ^* (if it exists) in left-hand-side of (5).

Numerical calculation of $p(Z)$ and δ^*:

- Highly challenging by classical numerical techniques (very high-dimensional problem) in particular in the 2Filtration setting.
- $\rightarrow$ in practice more simplistic models are used (parametric, continuous-time, no transaction costs).

- an approximate calculation is feasible thanks to modern deep learning techniques (exploited for other problems in finance e.g. in works by Cont, Sirignano, E, Han, Jentzen, Cheridito, Becker, ...).
- Straight forward Approach: consider only hedging strategies $\delta = (\delta_{t_k})_{k=1,\dots,n}$ of the form

$$\delta_{t_k} = F^{\theta_k}(S_{t_{k-1}}, \delta_{t_{k-1}}), \quad k = 1, \dots, n$$

  where F^{θ_k} is a neural network with weights parametrized by θ_k.
- Key point 1: neural networks are surprisingly efficient at approximating multivariate functions (see works by Bölcskei, Grohs, Kutyniok, Petersen, Wiatowski, ...).
- Key point 2: efficient machine learning optimization algorithms (stochastic gradient-type and backpropagation) and implementations (Tensorflow, Theano, Torch, ...) are available.

Approximate indifference price

- By cash-invariance, the indifference price $p(Z)$ is given as

$$p(Z) = \pi(-Z) - \pi(0), \text{ where}$$
$$\pi(X) := \inf_{\delta \in \mathcal{H}} \rho(X + (\delta \cdot S)_T - C_T(\delta)).$$

- For suitable parameter set $\Theta_M \subset \mathbb{R}^r$ and $\delta^\theta_{t_k} = F^{\theta_k}(S_{t_{k-1}}, \delta^\theta_{t_{k-1}})$ as above, approximate $\pi(X)$ by

$$\pi_M(X) := \inf_{\theta \in \Theta_M} \rho\Big(X + (\delta^\theta \cdot S)_T - C_T(\delta^\theta)\Big).$$

- Directly amenable to machine learning optimization algorithms (stochastic gradient-type and backpropagation) if ρ is entropic risk measure ($\rightarrow$ exponential utility indifference price) or more generally an *optimized certainty equivalent*, i.e.

$$\rho(X) := \inf_{w \in \mathbb{R}} \{w + E[\ell(-X - w)]\} \qquad (6)$$

for $\ell \colon \mathbb{R} \rightarrow \mathbb{R}$ continuous, non-decreasing and convex.

Approximate indifference price

- By cash-invariance, the indifference price $p(Z)$ is given as

$$p(Z) = \pi(-Z) - \pi(0), \text{ where}$$
$$\pi(X) := \inf_{\delta \in \mathcal{H}} \rho(X + (\delta \cdot S)_T - C_T(\delta)).$$

- For suitable parameter set $\Theta_M \subset \mathbb{R}^r$ and $\delta^\theta_{t_k} = F^{\theta_k}(S_{t_{k-1}}, \delta^\theta_{t_{k-1}})$ as above, approximate $\pi(X)$ by

$$\pi_M(X) := \inf_{\theta \in \Theta_M} \rho\Big(X + (\delta^\theta \cdot S)_T - C_T(\delta^\theta)\Big).$$

- Directly amenable to machine learning optimization algorithms (stochastic gradient-type and backpropagation) if ρ is entropic risk measure ($\rightarrow$ exponential utility indifference price) or more generally an *optimized certainty equivalent*, i.e.

$$\rho(X) := \inf_{w \in \mathbb{R}} \{w + E[\ell(-X - w)]\} \tag{6}$$

for $\ell \colon \mathbb{R} \to \mathbb{R}$ continuous, non-decreasing and convex.

Example Study: Heston model with CVar

$$dS_t^{(1)} = \sqrt{V_t} S_t^{(1)} dB_t, \quad S_0^{(1)} = s_0$$

$$dV_t = \alpha(b - V_t)dt + \sigma\sqrt{V_t} dW_t, \quad V_0 = v_0$$

B and W are Brownian motions with $d\langle B, W\rangle = \rho dt$

$$(\alpha, b, \rho, \sigma, v_0, s_0) = (1, 0.04, -0.7, 2, 0.04, 100)$$

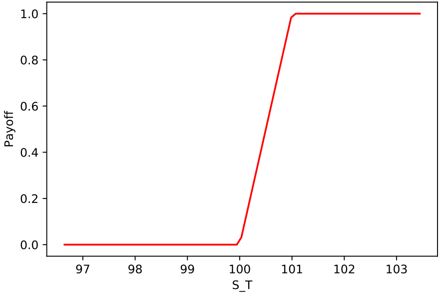
Payoff and Hedging

- Payoff: Call spread (see next slide) with maturity $T = 30$ days.
- Hedging instruments: Trade in $S^{(1)}$ and variance swap $S^{(2)}$.
- Trading: Daily rebalancing of portfolio.
- Risk-measure: α-CVar (expected shortfall),

$$\rho(X) := \inf_{w \in \mathbb{R}} \left\{ w + \frac{1}{1-\alpha} E[(-X - w)^+] \right\}.$$

Call spread

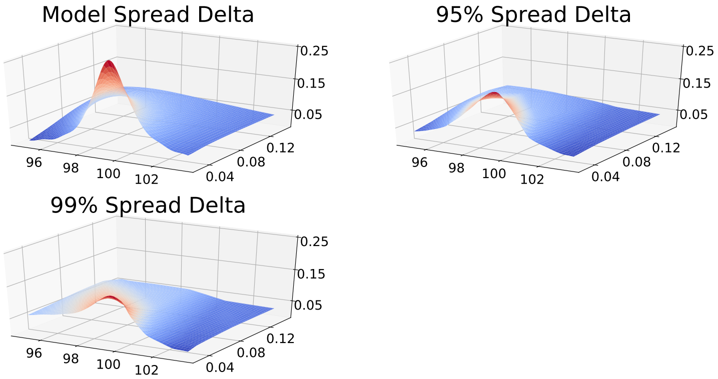
- Used by traders for (approximate) pricing / hedging of binary options.
- Payoff: $-\frac{1}{K_2-K_1}[(S_T^{(1)}-K_1)^+-(S_T^{(1)}-K_2)^+]$ for $K_1<K_2$.
- Here $K_1=s_0=100$, $K_2=101$:



Neural network approximation

- $\delta_{t_k} = F^{\theta_k}(S^{(1)}_{t_{k-1}}, S^{(2)}_{t_{k-1}}, \delta_{t_{k-1}})$ and for each k, F^{θ_k} is a feed-forward neural network with two hidden layers (15 nodes each) and ReLU activation function ($x \mapsto x^+$).
- Use Adam (batch size 256) for training.

$\delta_t^{(s)}$ as a function of (s_t, v_t) for $t = 15$:



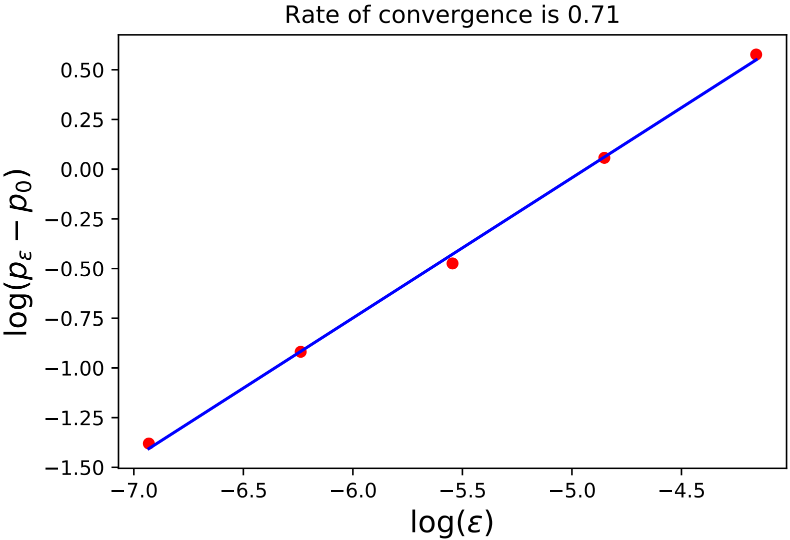
Higher risk-aversion $\leftrightarrow$ barrier shift

Price asymptotics: proportional transaction costs

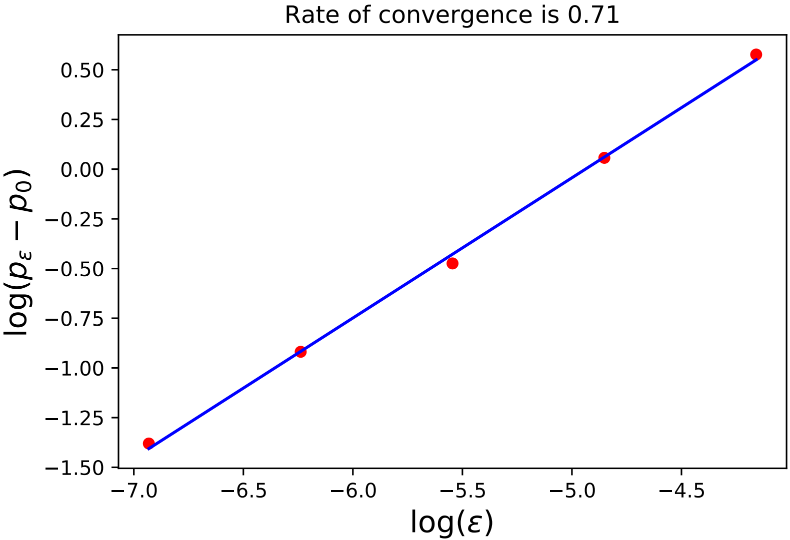
- $p_\varepsilon = p_\varepsilon(Z)$ is the exponential utility indifference price of Z for proportional transaction costs of size ε.
- For continuous-time models with $d = 1$:

$$p_\varepsilon - p_0 = O(\varepsilon^{2/3}), \qquad \text{as } \varepsilon \downarrow 0. \tag{7}$$

- Our methodology is good enough to reproduce (7) in a Heston model with $d = 2$ hedging instruments. For this case (or any other model with $d > 1$) no results on (7) have been available previously (neither theoretical nor numerical).



Another instance of the Unreasonable effectiveness of neural networks!



Another instance of the Unreasonable effectiveness of neural networks!

Reservoir Computing

- learning hedging strategies in a 2Filtration setting falls in the category of *learning dynamic phenomena*, i.e. we want to calculate as easily as possible the map from the price path of S to the hedging strategy.
- paradigm of reservoir computing: split the input-output map into a generic part (the *reservoir*), which is *not* trained and a readout part, which is trained, see work of Lyudmila Grigoryeva, Juan-Pablo Ortega, et al.
- how to choose the reservoir in case of learning hedging strategies?
- comparable question: learning the solution of a semi-martingale driven unknown equation

$$dX_t = \sum_{i=1}^{d} V_i(X_t) \circ dS_t^i \, , \; X_0 = x \in \mathbb{R}^N$$

for some smooth vector fields $V_i : \mathbb{R}^N \to \mathbb{R}^N$, $i = 1, \ldots, d$.

The signature process

Task: a large amount of tuples $(S(\omega), X(\omega))$ is given (the training data), describe a non-linear map which stores this information and allows to *generalize*. Notice that we do not neither need nor re-construct V_i for that.

In order to learn the map $(S_t)_{0 \le t \le T} \mapsto X_T$ from the paths of S to the solution of the above equation one considers linear systems in the free algebra with d generators (the product is denoted by $\otimes$).

$$dY_t = \sum_{i=1}^{d} Y_t \otimes e_i \circ dS_t^i \, , \; Y_0 = 1$$

This is an infinite dimensional system whose solution is just the collection of all iterated Ito-Stratonovich integrals.

Rough path theory tells that Y is a reservoir on which the dynamics of X can be represented: X_t can be written as a linear functional on Y_t, the linear functional depends solely on the vector fields $V_1, \dots, V_d$ and can therefore learned given enough training data.

A random localized signature

Instead of the previous infinite dimensional system there might be better choices to construct a reservoir: fix an activation function σ.

A random localized signature

- there is a set of hyper-parameters $\theta \in \Theta$, and a dimension M.
- depending on θ choose randomly matrices $A_1, \ldots, A_d$ on $\mathbb{R}^M$ as well as shifts $\beta_1, \ldots, \beta_d$ such that we have maximal non-integrability on a starting point $x \in \mathbb{R}^M$.
- one can tune the hyper-parameters $\theta \in \Theta$ such that

$$dZ_t = \sum_{i=1}^{d} \sigma(A_i Z_t + \beta_i) \circ dS_t^i \, , \; Z_0 = z$$

locally (in time) approximates X_t via a linear readout.

Elements of the proof

Elements of the proof

- Not only the signature process but all stochastic processes sharing this maximal non-integrability have the representation property stated in rough path theory.
- The cut-off dimension M leads to limited quality of approximation, but with high dimension it tends to 0.
- The tuning hyper-parameters θ allow to regularize the regression.
- The representation holds locally depending on the activation function.

Elements of the proof

Elements of the proof

- Not only the signature process but all stochastic processes sharing this maximal non-integrability have the representation property stated in rough path theory.
- The cut-off dimension M leads to limited quality of approximation, but with high dimension it tends to 0.
- The tuning hyper-parameters θ allow to regularize the regression.
- The representation holds locally depending on the activation function.

Elements of the proof

Elements of the proof

- Not only the signature process but all stochastic processes sharing this maximal non-integrability have the representation property stated in rough path theory.
- The cut-off dimension M leads to limited quality of approximation, but with high dimension it tends to 0.
- The tuning hyper-parameters θ allow to regularize the regression.
- The representation holds locally depending on the activation function.

Elements of the proof

Elements of the proof

- Not only the signature process but all stochastic processes sharing this maximal non-integrability have the representation property stated in rough path theory.
- The cut-off dimension M leads to limited quality of approximation, but with high dimension it tends to 0.
- The tuning hyper-parameters θ allow to regularize the regression.
- The representation holds locally depending on the activation function.

Outlook

- find optimal reservoirs for non-standard stochastic optimization problems to facilitate machine learning solutions.
- combine machine learning optimization techniques with Bayesian techniques to improve knowledge about the model.
- what can be done in case of rough paths can be also done in case of regularity structures: construct optimal representation systems for solutions of singular SPDEs to facilitate learning of solutions.

Outlook

- find optimal reservoirs for non-standard stochastic optimization problems to facilitate machine learning solutions.
- combine machine learning optimization techniques with Bayesian techniques to improve knowledge about the model.
- what can be done in case of rough paths can be also done in case of regularity structures: construct optimal representation systems for solutions of singular SPDEs to facilitate learning of solutions.

Outlook

- find optimal reservoirs for non-standard stochastic optimization problems to facilitate machine learning solutions.
- combine machine learning optimization techniques with Bayesian techniques to improve knowledge about the model.
- what can be done in case of rough paths can be also done in case of regularity structures: construct optimal representation systems for solutions of singular SPDEs to facilitate learning of solutions.

References

- H. Bühler, L. Gonon, J. Teichmann, and B. Wood:
  Deep Hedging, Arxiv, 2018.
- C. Cuchiero, I. Klein, and J. Teichmann:
  A fundamental theorem of asset pricing for continuous time large financial markets in a two filtration setting, Arxiv, 2017.
- M. Hairer:
  Theory of regularity structures, Inventiones mathematicae, 2014.

Thanks for your inspiring and deep research!

Thanks for all your generous work for the community

On many happy and active years for you to come,
dear Yuri!