Deep Hedging

Machine-driven trading of derivatives under market frictions

Swissquote Conference 2018 on Machine Learning in Finance
Geneva, Nov 9th 2018

Dr. Hans Buehler

J. P. Morgan

Joint work with
Lucas Gonon (ETH), Jonathan Kochems (JPM), Baranidharan Mohan(JPM), Josef Teichmann (ETH), Hans Buehler (JPM)

Disclaimer

Outline

- Models in an exotic derivatives business
- Teaching a machine to think like a trader
- First steps: toy model trading
- Moving further into the real world

How are models used in an exotic derivatives business?

Pricing new trades

- Classical risk-neutral models are ubiquitous

Price → $$V_0 = E^Q\left[\frac{V_T}{B_T}\right]$$ ← Expectation under risk-neutral measure

  - Disregard any existing portfolio and price the derivative under the assumption that perfect replication is possible
  - Apply **local** adjustments: hedging costs (trader's estimate), model limitation adjustments, …
  - For larger trades, consider **global** adjustments depending on existing portfolio: credit charge, concentration charge, etc.

How are models used in an exotic derivatives business?

Hedging

- Compute the price with the usual classical model

$$V_0 = E^Q\left[\frac{V_T}{B_T}\right]$$

- Then compute "greeks"

$$\frac{\partial V_0}{\partial X}$$

  - For factors which are stochastic in the model, and parameters which aren't (e.g. interest rates in a local volatility model)
- Based on the greeks, decide which hedging instruments to buy/sell
  - The right hedge is *not* just the model risk
  - Traders adjust the actual traded risk with "experience/skill"
  - He/she needs to be aware of transaction costs, market dynamics (such as vol-spot correlation), concentration and liquidity risk…

How are models used in an exotic derivatives business?

Apply constraints

- Internal: control the risks we take, ensure efficient use of capital
- External: regulatory, legal

Examples:

- Direct risk and stress limits based on the model:

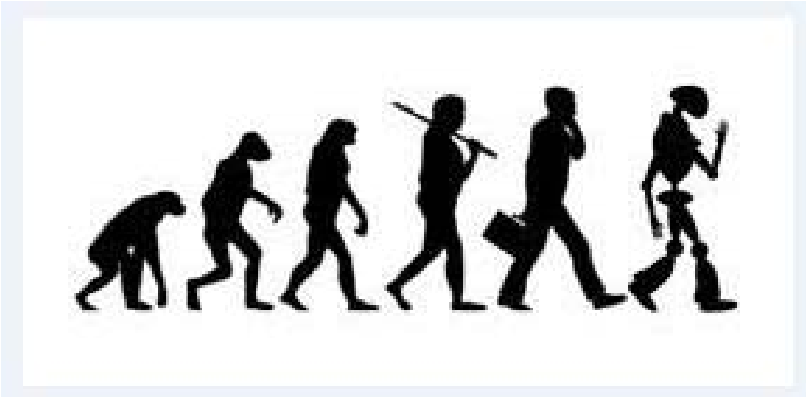
$$L < \frac{\partial V_0}{\partial X} < U$$

$$V_0(X) - V_0(X + \text{stress}) < M$$

- Limits on CVaR
- Capital requirements – many determining factors
- Short selling bans

- These constraints are not usually part of the valuation model

Beyond the classical approach



- We want to increase automation in the business
- The risk management model needs to do more

- It should include transaction costs, lack of liquidity, and constraints
- This means accepting that perfect replication is impossible…

Outline

- Models in an exotic derivatives business
- Teaching a machine to think like a trader
- First steps: toy model trading
- Moving further into the real world

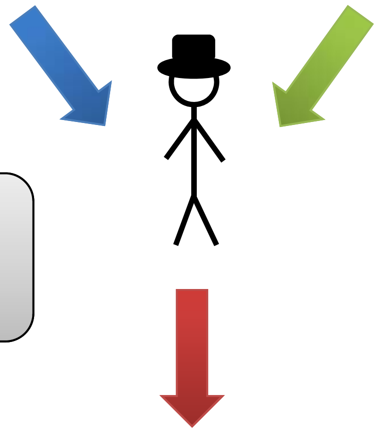
Trading inputs and outputs

- Risk management

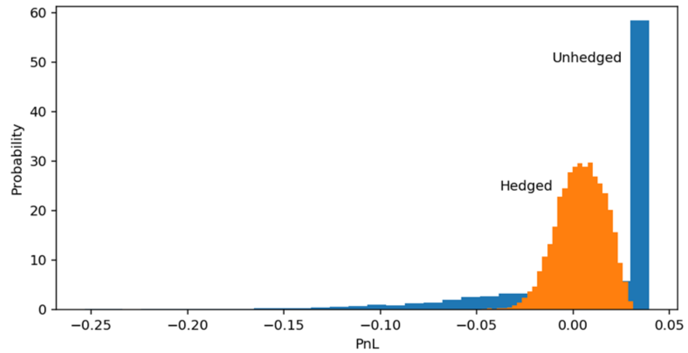
Portfolio of derivatives

Market state: prices of hedging instruments, cost of trading, liquidity,

Alternative Data: News, historic trading pattern



In the real world, we accept risk when we trade

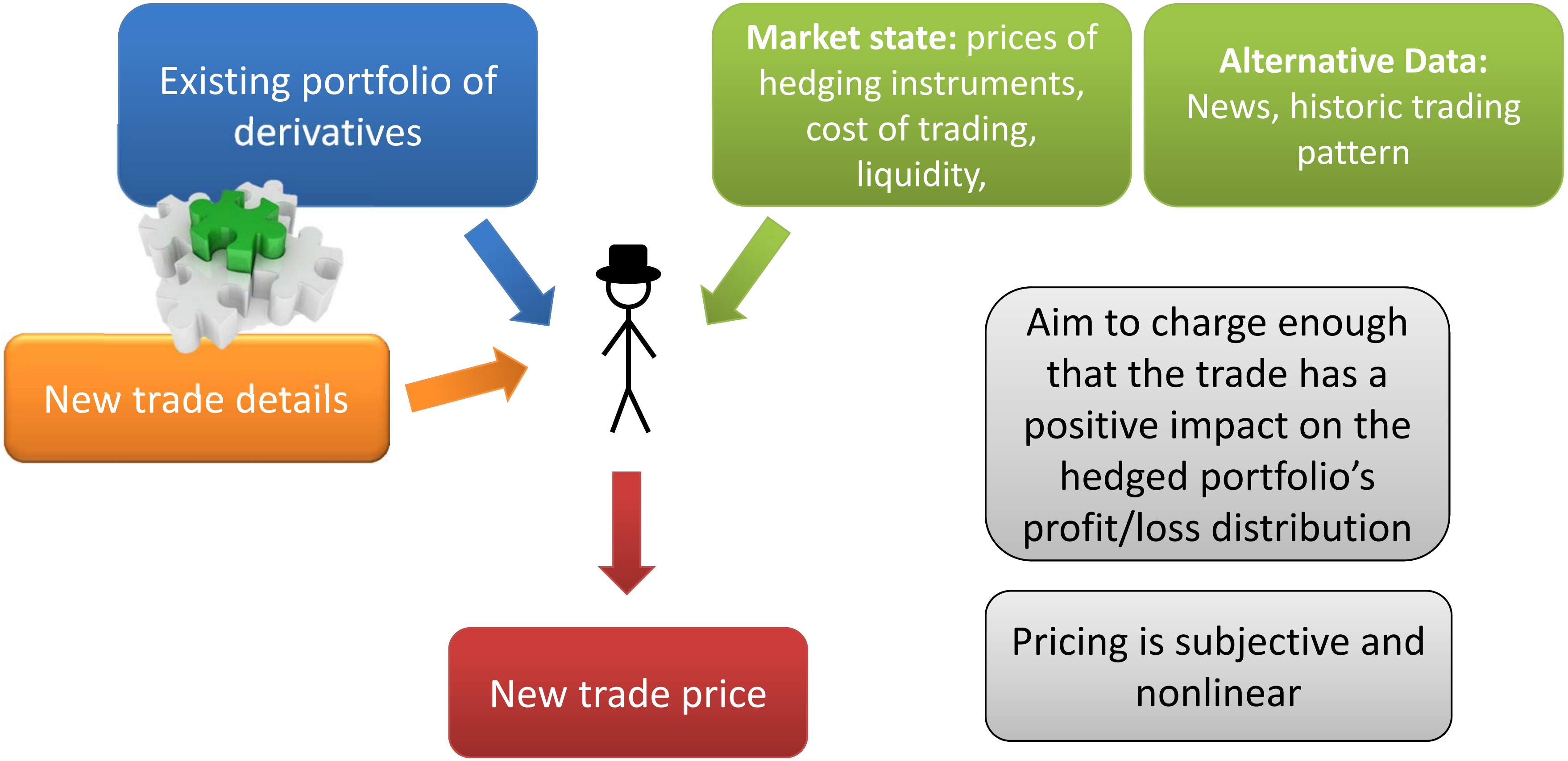


Buy/sell decision

Aiming to optimize future PnL *distribution*

Trading inputs and outputs

- Pricing



How to compare profit/loss distributions?

- We could use classical “Markoviz” portfolio optimization
  - Maximize expected return while penalizing variance

$$E(w_T) = \mathbb{E}[w_T] - \lambda \mathrm{Var}[w_T]$$

Risk-adjusted value

w_T is terminal wealth

λ is a measure of risk aversion

- Note that $E(w_T)$ is a function on the *distribution* of terminal wealth
- But mean-variance is not a good measure if terminal wealth is not normally distributed
  - Exist on-monotone cases where $X \geq Y$ but $E(X) < E(Y)$

How to compare profit/loss distributions?

- What are sensible conditions for our risk-adjusted value function $E(w_T)$?
- Monotonicity

$$X \geq Y \Rightarrow E(X) \geq E(Y)$$

More is better

- Convexity

$$E(\alpha X + (1-\alpha)Y) \geq \alpha E(X) + (1-\alpha)E(Y), \alpha \in [0,1]$$

We are risk-averse

- Cash invariance

$$E(X+c) = E(X) + c$$

There is no risk adjustment for cash

-$E(\cdot)$ is a **convex risk** measure

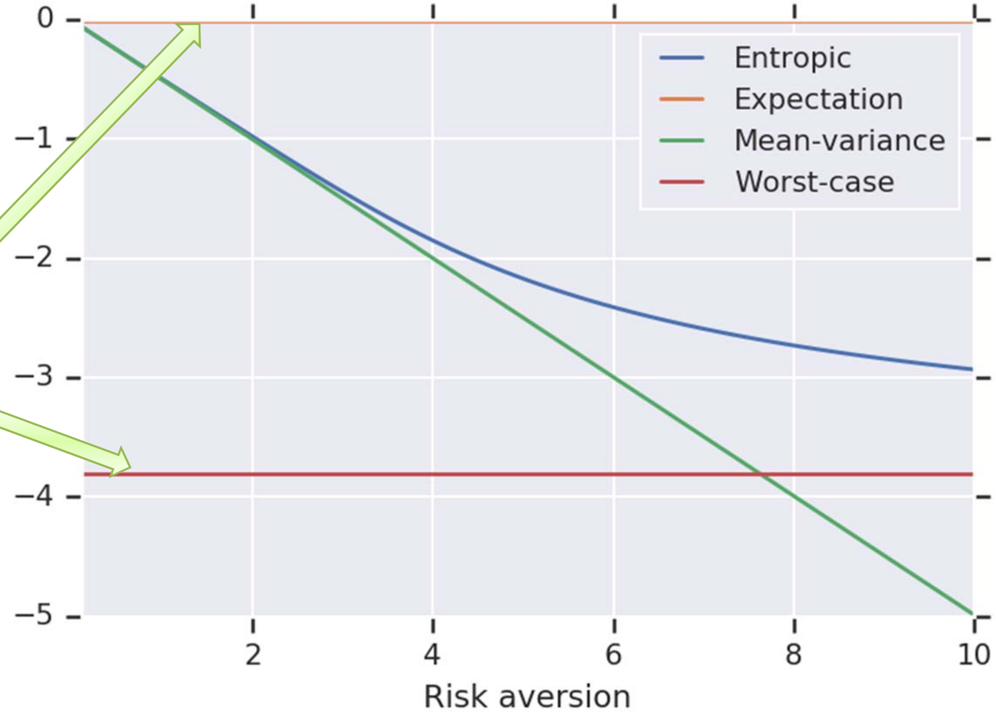
How to compare profit/loss distributions?

- We will mostly use the *entropic* measure: $E(X) = -\frac{1}{\lambda}\ln \mathbb{E}\left[e^{-\lambda X}\right]$
- Equivalent to mean-variance for small risk-aversion parameter λ:

$$-\frac{1}{\lambda}\ln \mathbb{E}\left[e^{-\lambda X}\right] = \mathbb{E}[X] - \frac{1}{2}\lambda\,\mathrm{Var}[X] + \cdots$$

- Example: $X \sim \mathcal{N}(0,1)$
- Plot risk-adjusted value

Bounded by the two extreme risk-adjusted values: risk-neutral and worst-case



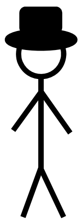
Hedging

- We can now express preferences on future profit/loss distributions
- *Hedging* is the act of buying and selling “hedging instruments” to optimize that distribution
- A *hedging strategy* is a function:

Liquid instruments with observable prices

$$a_t^\pi = a_t^\pi(t, s_t)$$

π parameterized the strategy; e.g. as the weights of neural networks



State s_t is the history of everything including our current book

- It tells us how much of each hedging instrument to buy or sell at each time t, for every possible state s_t
- Not all actions are possible – in general a_t^π will be subject to limits which are also state-dependent (e.g. short-sell constraints)

Hedging

- How does the hedging strategy π contribute to the terminal profit/loss?

$$w_T(\pi; z) = \sum_{j=1}^{T} \delta_j^\pi \cdot h_j + z_j - a_j^\pi \cdot H_j - c_j^\pi$$

δ_j^π: accumulated current positions ("deltas"), $\delta_j^\pi = \delta_{j-1}^\pi + a_j^\pi$

h_j: cashflows generated by hedging instruments

z_j: cashflows from our exotic derivatives portfolio

H_j: mid prices of hedging instruments

c_j^π: transaction costs incurred, $c_j^\pi = c(a_j^\pi, s_j)$

Note: all cash flows are discounted

Hedging

- The terminal profit/loss is not deterministic – our task is to optimize it
- That means maximizing

$$E[w_T(\pi; z)] = E\left[\sum_{j=1}^{T} \delta_j^\pi \cdot h_j + z_j - a_j^\pi \cdot H_j - c_j^\pi\right]$$

We apply the value function to the distribution over future real-world states

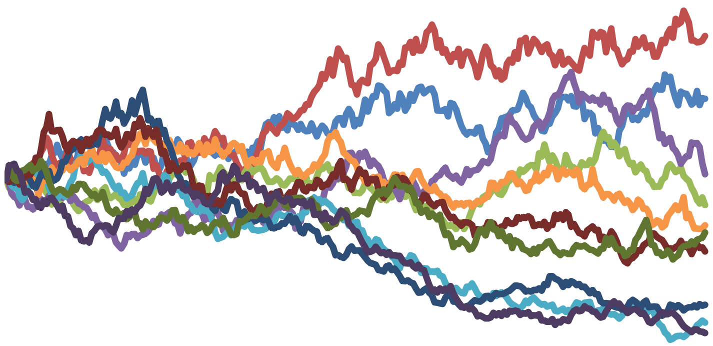
We need to find the optimal function a_j^π that meets our constraints.

Path dependency
The feasible set of allowed actions a_j^π depends on past decisions $a_{j-1}^\pi \dots a_0^\pi$.

- Two key challenges:
  - How to generate the distribution
  - How to optimize the hedging function

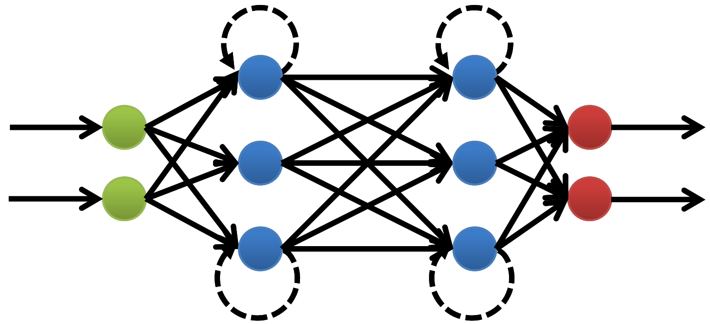
Market simulators

- To generate the profit/loss distribution for a given strategy, we need to simulate future states of the world
  - Prices of available hedging instruments
  - Corresponding cash flows from exotic derivatives
- We should be simulating in the real-world measure, not $\mathbb{Q}$
  - The real world has “statistical arbitrage”, i.e. with normal risk aversion some trades statistically make money (e.g. shorting options, sell long-dated bonds).
  - Deep Hedging will attempt to take advantage of these opportunities.
    - Absence of arbitrage =/=> absence of statistical arbitrage (e.g. GBM with drift)
    - Existence of arbitrage =/=> existence of statistical arbitrage (e.g. if risk-aversion is very high)
- For the experiments presented here, will use classical $\mathbb{Q}$ models



Optimizing the hedging strategy

- We use a *deep neural network* to represent the strategy a_j^π



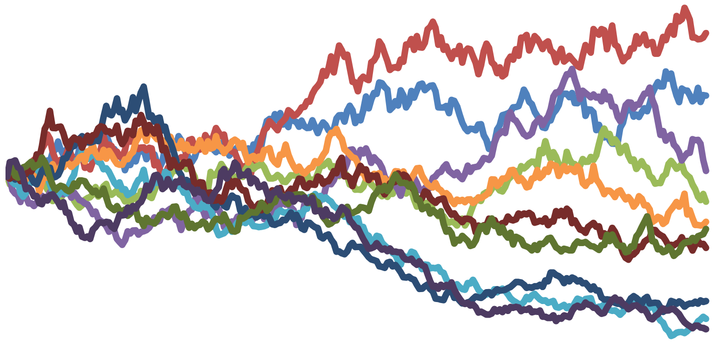
- Inputs:
  - Current market state ← Prices of all hedging instruments
  - Relevant product state variables ← Harvested automatically
- LSTM cells to capture path dependence
  - Potentially important when we have transaction costs
  - Allows memory of our previous hedging decisions

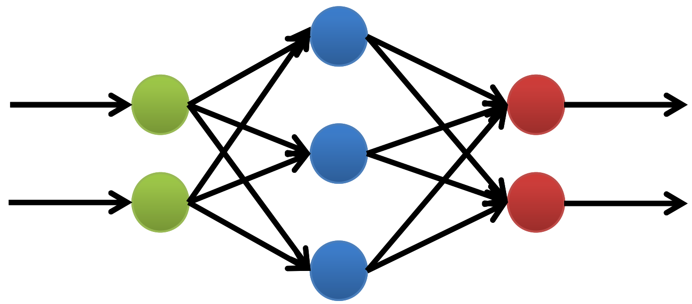
Deep Hedging

Simulate many future states of the world

Compute risk-adjusted value on a batch of paths for neural network strategy

Update network parameters by following gradient





$$\frac{\partial E(w_T(\pi; z))}{\partial W_i}$$

Outline

- Models in an exotic derivatives business
- Teaching a machine to think like a trader
- First steps: toy model trading
- Moving further into the real world

Start simple

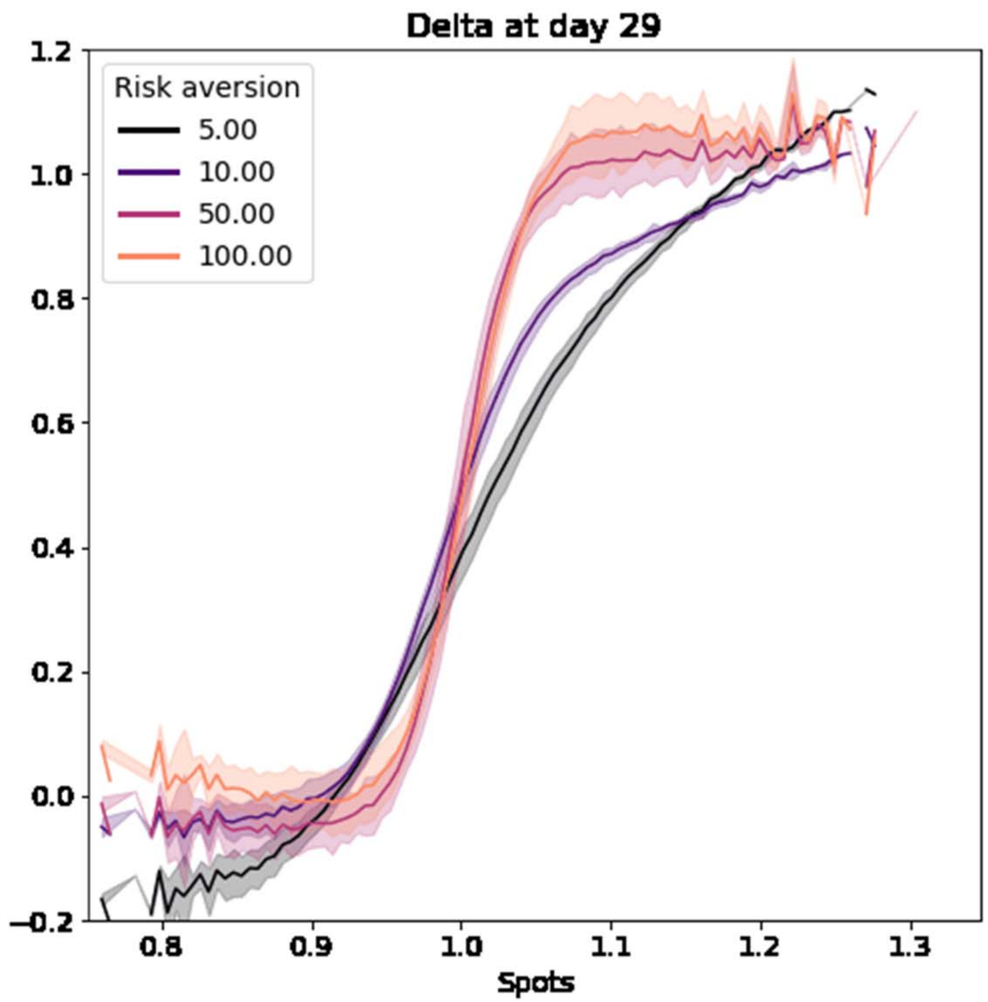
- Hedge a short at-the-money 30-day European call

$$z_T = -(S_T - K)^+$$

- Generate paths in Black-Scholes
- Check the impact of transaction costs, risk aversion, and risk limits

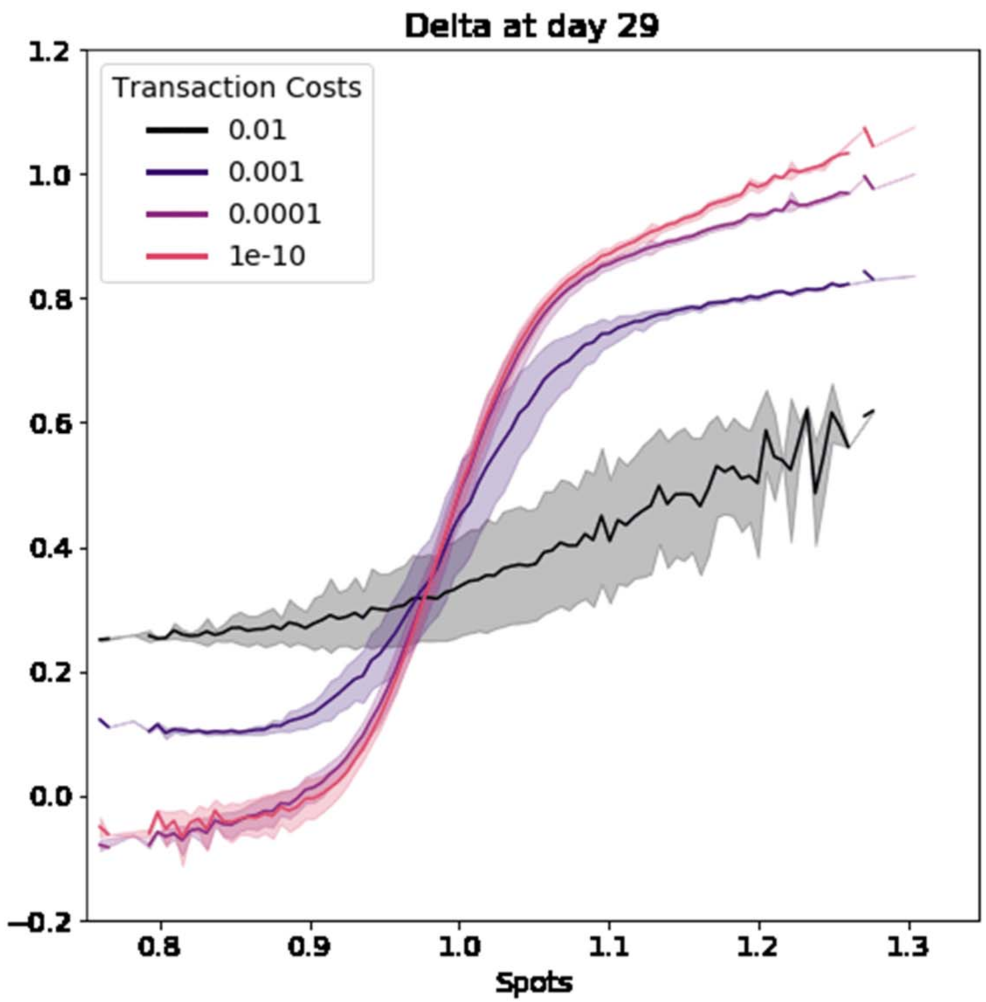
Risk Aversion (Entropy)

- Vanilla option delta
- 10bps cost
- No limits
- Entropic value
- Black-Scholes simulator



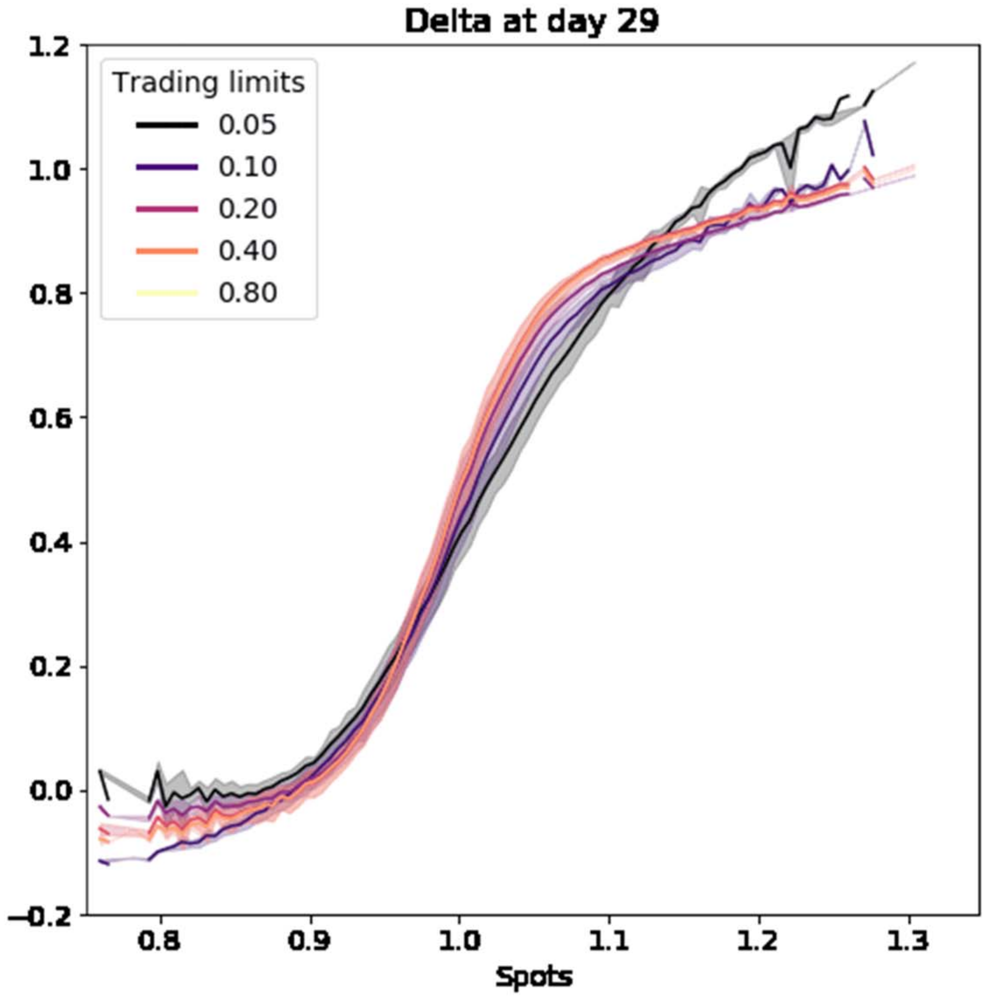
Transaction costs (Entropy)

- Vanilla option delta
- No limits
- Entropic value
- Risk aversion 10
- Black-Scholes simulator



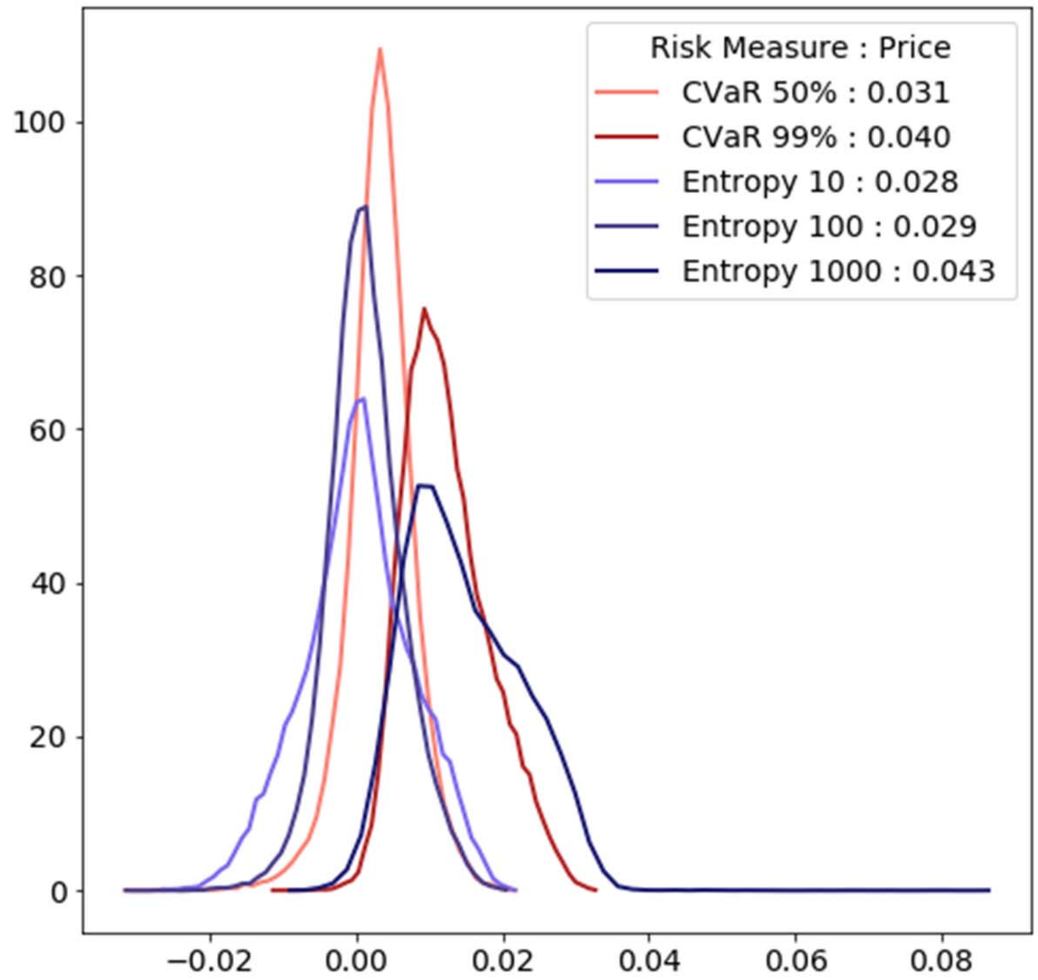
Trading limits

- Vanilla option delta
- 0.01% proportional cost
- Entropic value
- Risk aversion 10
- Black-Scholes simulator



Risk measure

- Vanilla option PnL distribution
- 0.01% cost
- No limits
- Black-Scholes simulator



Forward-starting options

- Increase the complexity: simulate with Heston model
- Compute optimal spot-only hedges for forward-starting options

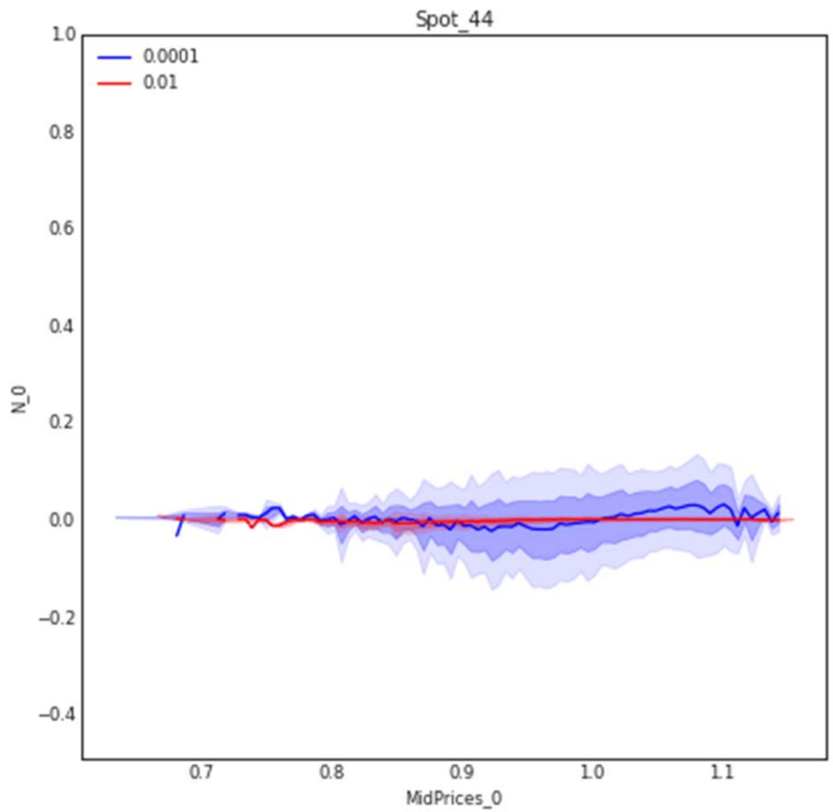
$$z_T = \max\left(0, \frac{S_T}{S_t} - K\right)$$

- 15-day forward start
- 45-day maturity
- Daily hedging
- Entropic value with risk aversion 50
- No limits

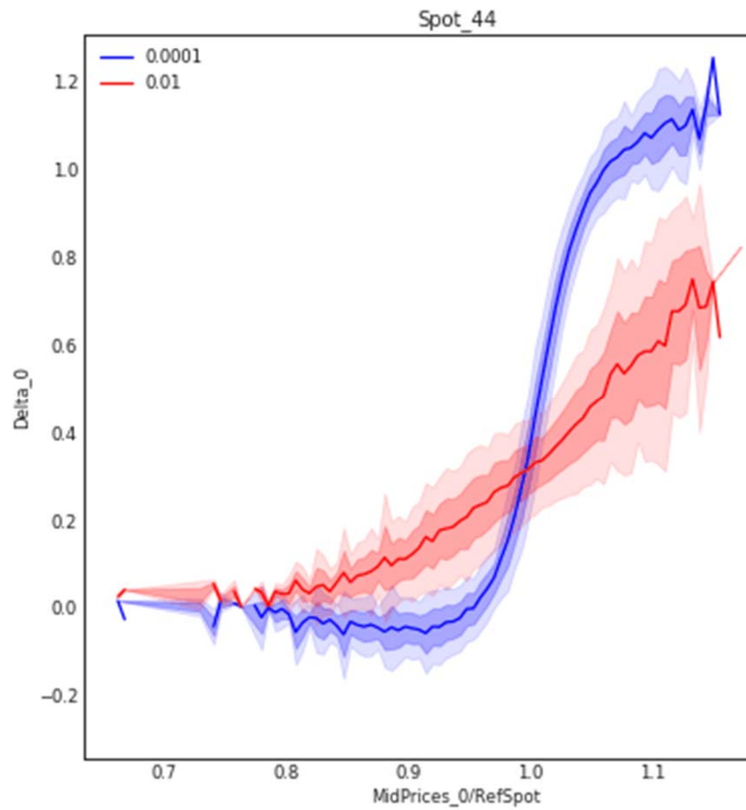
Forward-starting options

- Impact of transaction costs on incremental and total delta

Daily Delta Amount from 0th day to Maturity



Total Delta Amount from Ref day to Maturity



Outline

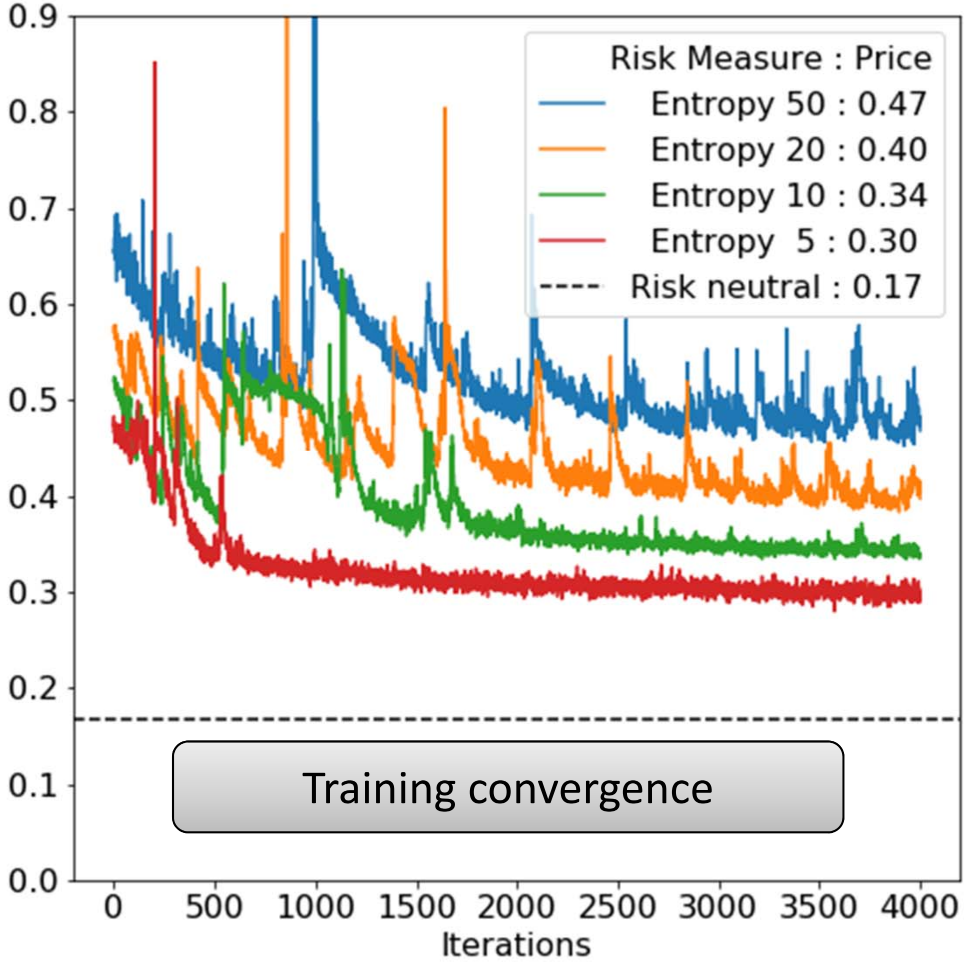
- Models in an exotic derivatives business
- Teaching a machine to think like a trader
- First steps: toy model trading
- Moving further into the real world

"Autocallable" note

- Popular retail payoff:
  - Client is short a down-and-in put paid at maturity
  - Upper knockout barrier
  - Fixed coupons until KO
- 0.1% transaction costs
- No limits
- Risk aversion 20
- Entropic value
- Monthly hedging

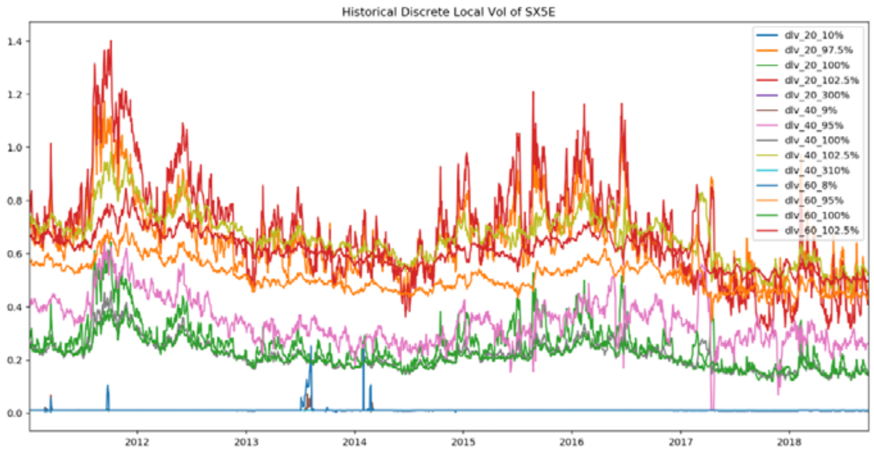
Portfolio of autocallables

- Based on a real portfolio
- 0.1% transaction costs
- No limits
- Local volatility simulator
- Monthly hedging

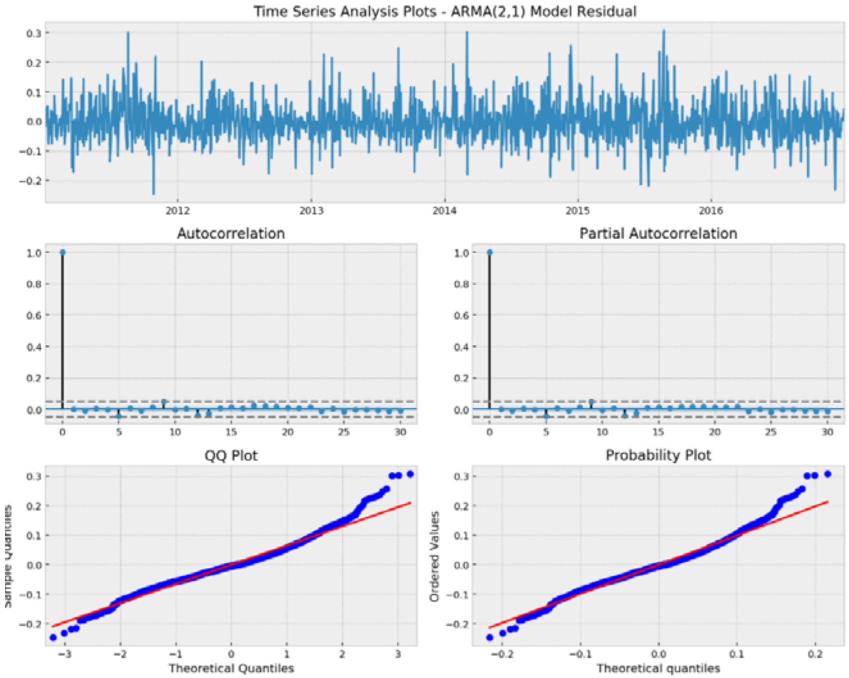


Market simulator

- Go beyond "classical" models – build a statistical model instead



Challenge: avoid arbitrage when simulating options



Simulate discrete local volatilities to avoid static arbitrage

Dynamic arbitrage still a challenge

Market simulator

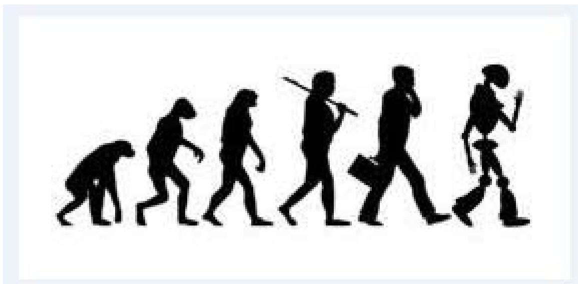
- How do we build a full statistical market simulator that reflects real-world drifts but is arbitrage-free?
  - Does it need to be fully arbitrage-free?
- What about rare events?
  - A statistical simulator is not likely to capture these well
- In particular, we want the model to behave well in a stress scenario, and to price in the risk appropriately
  - Should we insert stress events into the market simulator?
  - With what probability? Historical likelihood?
- For equities we focused on spot and volatility – there’s lots more
  - Rates, spreads, FX, …

Conclusions

- We formalized the task of pricing and managing the risk of an exotic derivatives portfolio
- Obtaining the optimal hedging strategy is a difficult problem
- Representing the strategy as a neural network makes it tractable
  - Optimization typically takes minutes on CPU for the toy examples here
- So far it works for:
  - Vanillas, cliquets, barrier options, large portfolios
  - With transaction costs and risk limits
  - Simulators based on classical pricing models (Black Scholes, local volatility, Heston)

Many more interesting challenges ahead

- Developing statistical ($\mathbb{P}$) market simulators for options
- Do we need to compute hedging strategies all the way to maturity?
  - Can we come up with an efficient way to represent a portfolio of exotics as a state?
- How do we choose our risk measure? Can we derive effective real-world risk-measures from the choices people make?
- Go beyond equities: FX, rates, etc.
- Ultimate goal: automated pricing and hedging of exotic derivatives



References and thanks

- Credits:
  - Hans Buehler, Lukas Gonon, Josef Teichmann, Hans Buehler, Jonathan Kochems, Barani Mohan, Blanka Horvath, Len Bai, Pradeepta Das
- Paper:
  - **Deep Hedging,** Hans Buehler, Lukas Gonon, Josef Teichmann, Hans Buehler, https://arxiv.org/abs/1802.03042
  - Note that the architecture has moved on since the paper