Deep Hedging

Ben Wood
Quantitative Research
JPMorgan

Oxford ML x Finance Summer School
August 2022

J.P.Morgan

Overview

J.P.Morgan

Derivatives trading

- Derivatives
- What a trader does
- Hedging as reinforcement learning

Deep Hedging

- Rewards and utility functions
- Optimized certainty equivalents
- Architecture
- Examples

Market simulation

- Option prices
- Compression
- Time series generation

Deep Bellman Hedging

- Beyond policy search

Derivatives primer

J.P.Morgan

Derivatives

- Financial contracts defining payments derived from the prices of underlying assets
- Stocks, indices, bonds, rates, FX, commodities

Markets

- Large in volume and notional terms
- Standard derivatives are traded on exchanges
- More complex derivatives are traded directly between counterparties (OTC)

Participants

- Sell side: banks, market making firms
- Buy side: asset managers, hedge funds, pension funds, insurance companies, retail investors

Activities

- Investment
- Hedging
- Market making

Examples

J.P.Morgan

FX futures contract

- Contract to exchange fixed amounts of two currencies on a given future date
  – British Pound Futures Sep 22 1.214

Equity index call option

- Contract that pays the amount by which the index level at **maturity** exceeds the **strike price**, if that amount is positive: $V_T = \max(0, I_T - K)$
  – Euro Stoxx 50 Dec 22 3800 CALL

Equity worst-of basket autocallable

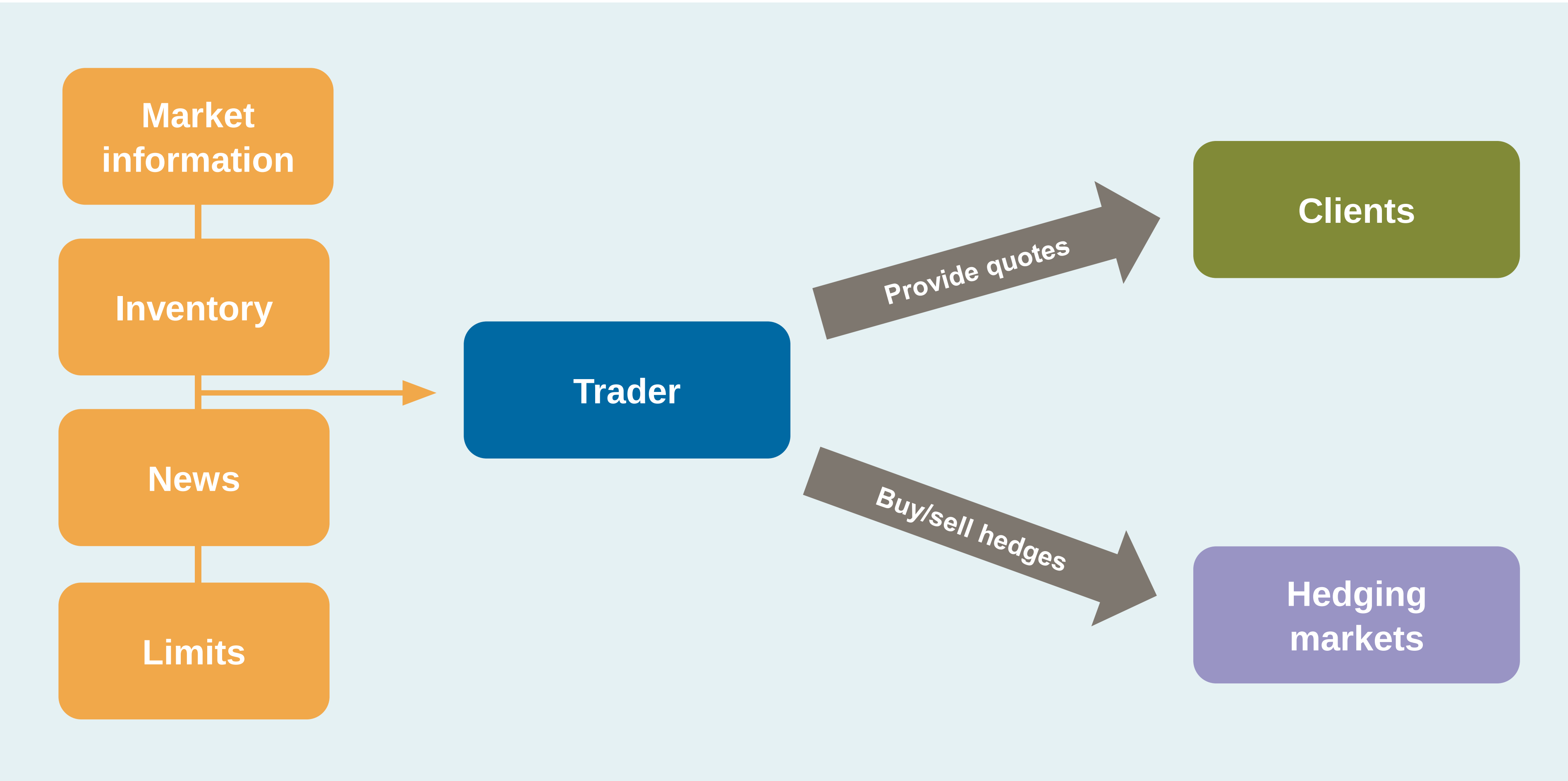
- Exotic option tracking the performance of the worst-performing stock in a basket
- Pays a quarterly coupon if the worst-of is above a coupon barrier threshold at quarter end
- Terminates early if the worst-of is above a higher knockout barrier threshold at quarter end
- Repays the notional on termination
- Repays the notional minus a down-and-in put at maturity

Derivatives trading

J.P.Morgan

What does a (sell-side) trader do?

- Key tasks are quoting and hedging
- Monitor the market, know her inventory, react to news, understand risk and PnL, make sure to comply with controls and limits



Hedging

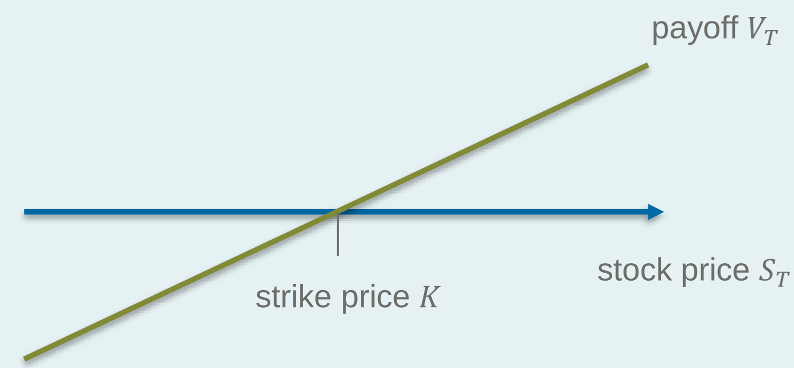
J.P.Morgan

Trading derivatives involves risk

- At least one future payment of uncertain amount
- How can we reduce the risk?
- Trade in the underlying asset

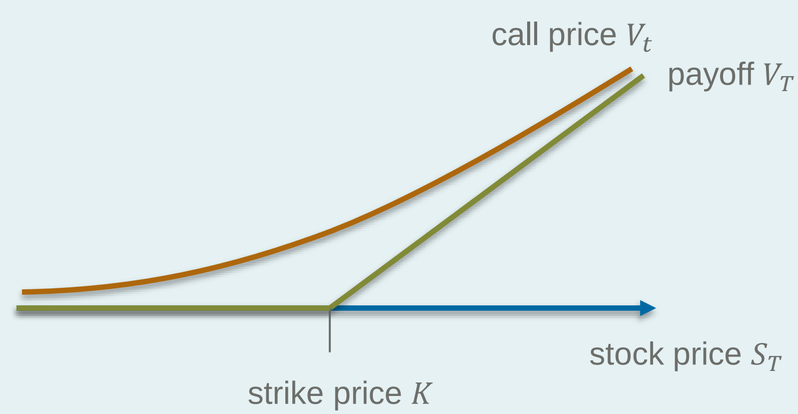
Example: forward contract

- At maturity T, we pay $V_T = S_T - K$
- To reduce the risk, we can simply borrow money and buy the stock today
- All the uncertainty has been removed



Example: call option

- At maturity T, we pay $V_T = N \max(0, S_T - K)$
- To reduce the risk, we can borrow money and buy the stock today
- We will need somewhere between 0 and N units
- If the stock price goes up, we need to buy more, and if it goes down, we need to sell



Classical models

J.P.Morgan

Risk-neutral valuation models from math finance

- Provide prices based on replication arguments

$$V_t = \mathbb{E}^Q[V_T]$$

- For simple exchange-traded products, the inputs to the pricing model are fitted to the market
- The model is used to interpolate the price – for complex products, there is uncertainty
- Provide greeks: sensitivities to market data inputs

$$\Delta = \frac{\partial V}{\partial S}$$

Quoting

Model price

↓

Adjustments

↓

Quote price

Hedge costs
Risk
Client information
Axe/inventory

Hedging

Model greeks

↓

Trader skill

↓

Hedge

Market view
Experience
Availability/cost
Limits

Why is hedging hard?

J.P.Morgan

Key decisions

- When to hedge
- What to buy/sell

Should I re-hedge now?	What should I buy?
How much unhedged exposure do I have?	Flattening all greeks is not practical
Am I near my risk limits?	Which hedges are cheap / expensive?
Is my current position carrying well?	How do the prices of the hedges move together?
How much will it cost to hedge?	Which hedges will need adjusting again later?

Trade-off of risk vs cost

AI for hedging

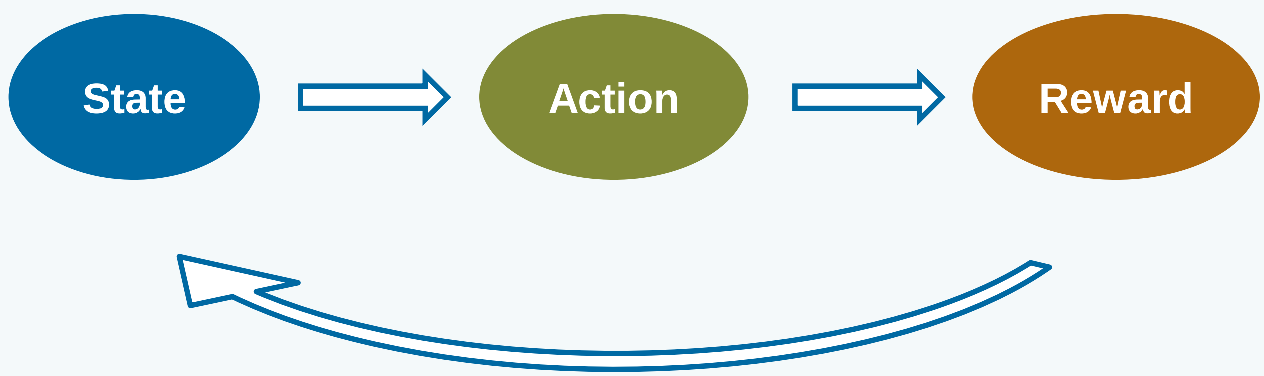
J.P.Morgan

The Reinforcement Learning paradigm is a good fit for trading

State
- Current portfolio
- Current market prices of hedging instruments
- History
- Signals: news, social media, …

Action
- Buy / sell hedging instruments

Reward
- Payments from client trades and hedges
- Includes fees (negative reward)
- Risk adjustment



Deep Hedging principles

J.P.Morgan

Problem statement

- Use AI to find optimal hedging strategies for derivatives
- Allow for important real-world effects (costs, discrete hedging, limits)
- Take a more systematic approach to hedging: less art, more science

Core ideas

- A hedging strategy is a **policy**: a function mapping state to action
- **State** includes the market and our portfolio
- **Actions** involve buying or selling liquid hedging instruments
- Cash payments from buying, selling, or from our existing portfolio are our **rewards**
- The policy will determine the profit and loss from hedging on any future path
- We define a **loss function** on the distribution of hedged P&L
- We optimize the policy with respect to this objective

J.P.Morgan

Episodic formulation

- Learn to hedge a **specific portfolio** Z to maturity
- Write the terminal gain of a set of hedging actions:

$$G^Z(a) = Z_T + \sum_{t=0}^{m-1} (\delta_t \cdot (H_{t+1} - H_t) - c_t(a_t)) = Z_T + \sum_{t=0}^{m-1} \left(a_t \cdot (H_T - H_t) - c_t(a_t)\right)$$

- Model the **action** as the output of a neural network, which represents our **policy**

$$a_t = a^\pi(s, t; \theta)$$

- Maximize the **utility** of the terminal gain distribution

$$\mathcal{L} = -U\left(G^Z(\theta)\right)$$

- Obtain sample paths of the market state, and evaluate the portfolio payments on each path

$$(H_0, H_1, \dots H_T)^i \to Z_T^i$$

- Train by applying **stochastic gradient descent** to the loss function in batches of samples

$$\theta \to \theta - \gamma \sum_i \nabla_\theta \mathcal{L}_i$$

Deep Hedging gains

J.P.Morgan

Deeper dive into the gains process

- Write the terminal gain of a set of hedging actions:

$$G^Z(a) = Z_T + \sum_{t=0}^{m-1} \left(\delta_t \cdot (H_{t+1} - H_t) \; - c_t(a_t)\right) = Z_T + \sum_{t=0}^{m-1} \left(a_t \cdot (H_T - H_t) - c_t(a_t)\right)$$

Portfolio cashflows

- Market state dependent
- Independent of actions

Transaction costs

- Typically a convex function of the action, e.g. proportional

Hedge instruments

- H_t is the vector of mid prices of the available hedge instruments
- Independent of actions

Actions

- a_t is the action at step t
- It depends on our policy
- Related to our hedge instrument holdings: $\delta_t = \delta_{t-1} + a_t$

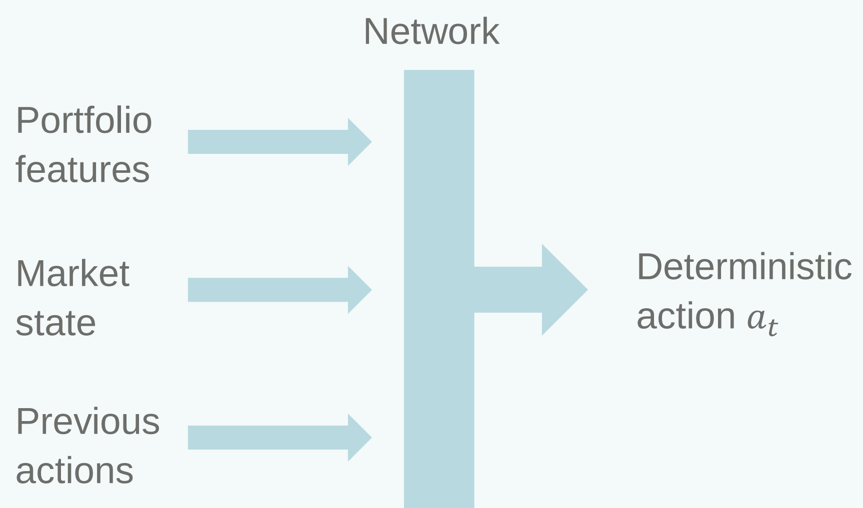
Deep Hedging actions

Meaning of the policy

- The actions represent how much of each hedge instrument to buy or sell at each step, in each state
- Modelled as the output of a neural network

$$a_t = a^{\pi}(s, t; \theta)$$

- At each time step:



- Dependence on previous actions introduces recursion

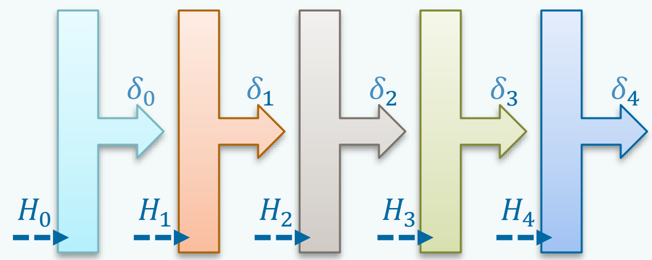
Deep Hedging actions

Architecture

- Different choices are possible, but all reflect the recursive nature of the problem

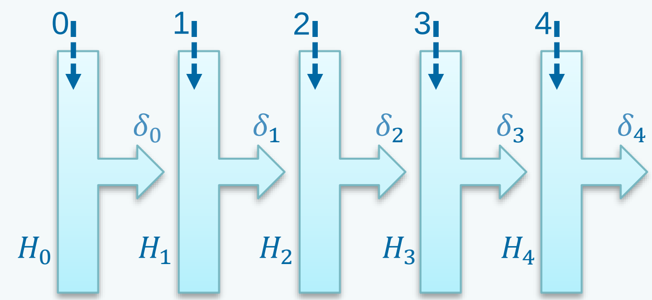
Sequential

$$a_t = f(H_t, \delta_{t-1}; \theta_t)$$



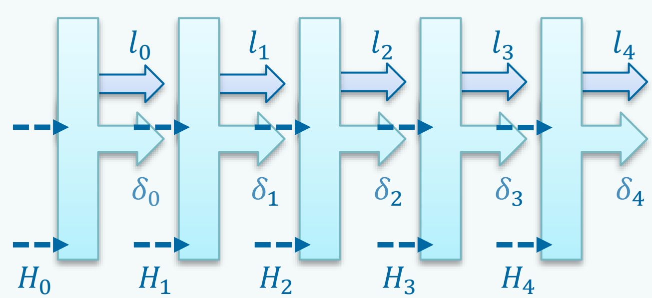
Shared weights

$$a_t = f(H_t, \delta_{t-1}, t; \theta)$$



LSTM

$$a_t = f(H_t, \delta_{t-1}, l_{t-1}; \theta)$$
$$l_t = g(H_t, \delta_{t-1}, l_{t-1}; \theta)$$



Utility functions

J.P.Morgan

What are the considerations for a utility function?

- Plays a critical role in determining the optimal policy
- Should reflect risk aversion and preference for positive PnL
- A classic choice in finance is **mean-variance**

$$U_\lambda(X) = \mathbb{E}[X] - \frac{\lambda}{2}\mathrm{Var}[X]$$

λ is a risk aversion parameter

- This is okay if X is normally distributed
- We can think of λ as the cash price of a unit of variance risk

When does mean-variance go wrong?

- Mean-variance is not monotonic
  - Consider two strategies X and Y
  - If $X > Y$ in all possible outcomes, X is clearly better
  - However, we may have $\mathbb{E}[X] - \frac{\lambda}{2}\mathrm{Var}[X] < \mathbb{E}[Y] - \frac{\lambda}{2}\mathrm{Var}[Y]$ and mean-variance then prefers Y

Utility functions

J.P.Morgan

Example

- Add a **free** lottery ticket to a portfolio

$$X = Z$$
$$Y = Z + L$$

L is the lottery ticket

$$U^{\lambda}(Y) = \mathbb{E}[Y] - \frac{\lambda}{2}\mathrm{Var}[Y] = U^{\lambda}(X) + \mathbb{E}[L] - \frac{\lambda}{2}\mathrm{Var}[L]$$
$$= U^{\lambda}(X) + pN\left(1 - \frac{\lambda}{2}(1-p)N\right)$$

- The free ticket is **rejected** if $\lambda > \frac{2}{N(1-p)}$

Desirable properties of utility functions

- Monotonicity

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

- Concavity, because we are risk-averse

$$U(\alpha X + (1-\alpha)Y) \geq \alpha U(X) + (1-\alpha)U(Y)$$

- Cash-invariance, to give U the meaning of a value

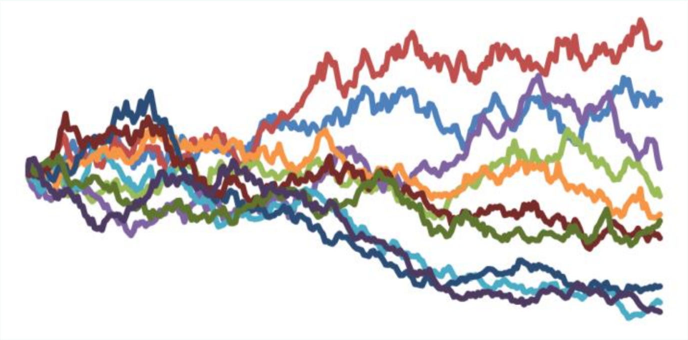
$$U(X + c) = U(X) + c$$

Training data

J.P.Morgan

Requirements

- We need samples of paths of the market, as long as the lifetime of the portfolio to be hedged
- For proof of concept, we can generate synthetic data
  - Use simple, off-the-shelf classical models, e.g. Black-Scholes, Heston, local volatility
  - Has the advantage of a baseline for performance
- For production use, we need more realistic data



Building the training dataset

Generate paths of hedge instruments

- Include asset spot prices, but usually also a grid of vanilla options
- Typically daily sampled
- Hedge instrument prices, including transaction costs, and payoffs

→

Decorate each path with cashflows from the portfolio to be hedged

- These are independent of actions

Training

J.P.Morgan

Optimization approach

- Finite time horizon
- Continuous, high-dimensional state and action space
- State is largely independent of actions: $S = (M, Z, \delta)$
- Objective based on terminal utility
- All motivate the choice of **gradient-based direct policy search**
  - Not common in the RL community
  - Related to REINFORCE, but deterministic policy

Stochastic gradient descent

- Compute the loss function for the current policy on a batch of paths
- Update the network parameters by following the gradient

$$\theta \to \theta - \gamma \sum_i \nabla_\theta \mathcal{L}_i$$

- Vanilla SGD / Adam / RMSProp

Deep Hedging

J.P.Morgan

Episodic formulation

- Learn to hedge a **specific portfolio** Z to maturity
- Write the terminal gain of a set of hedging actions:

$$G^Z(a) = Z_T + \sum_{t=0}^{m-1} \left(\delta_t \cdot (H_{t+1} - H_t) - c_t(a_t)\right) = Z_T + \sum_{t=0}^{m-1} \left(a_t \cdot (H_T - H_t) - c_t(a_t)\right)$$

- Model the **action** as the output of a neural network which represents our **policy**

$$a_t = a^\pi(s, t; \theta)$$

- Maximize the **utility** of the terminal gain distribution

$$\mathcal{L} = -U\left(G^Z(\theta)\right)$$

- Obtain sample paths of the market state, and evaluate the portfolio payments on each path

$$(H_0, H_1, \dots H_T)^i \rightarrow Z_T^i$$

- Train by applying **stochastic gradient descent** to the loss function in batches of samples

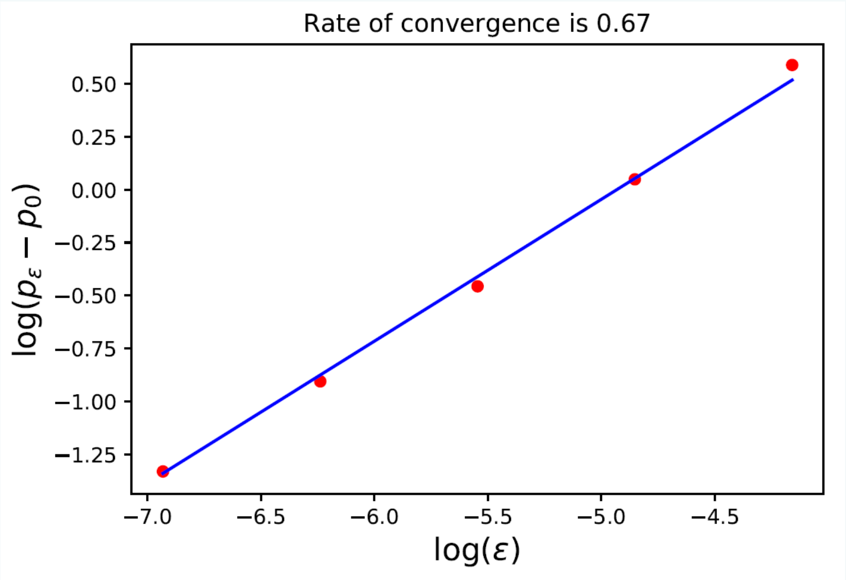
$$\theta \rightarrow \theta - \gamma \sum_i \nabla_\theta \mathcal{L}_i$$

Toy examples

J.P.Morgan

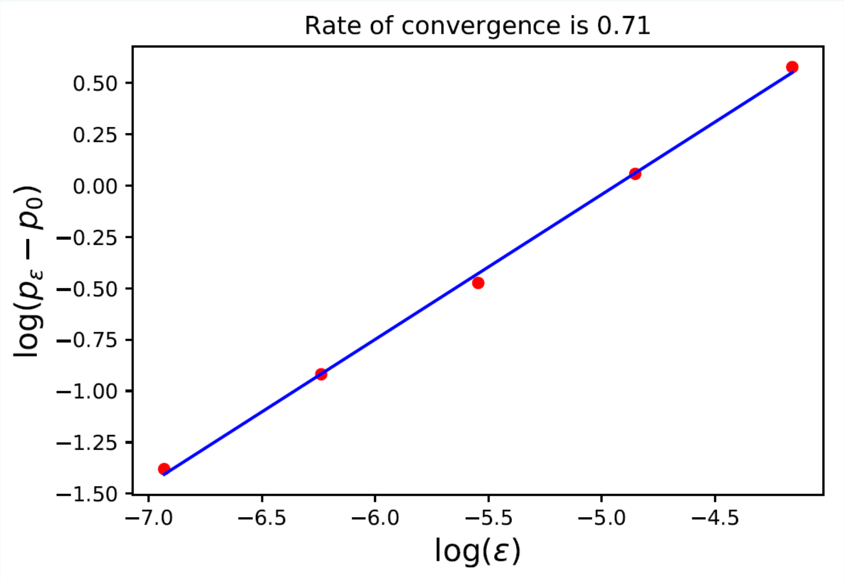
Example: Delta-hedging in a Black-Scholes world

- Delta-hedging a call option with transaction costs in a Black-Scholes world
- Compare with the known theoretical result



Example: Delta-hedging in a Heston world

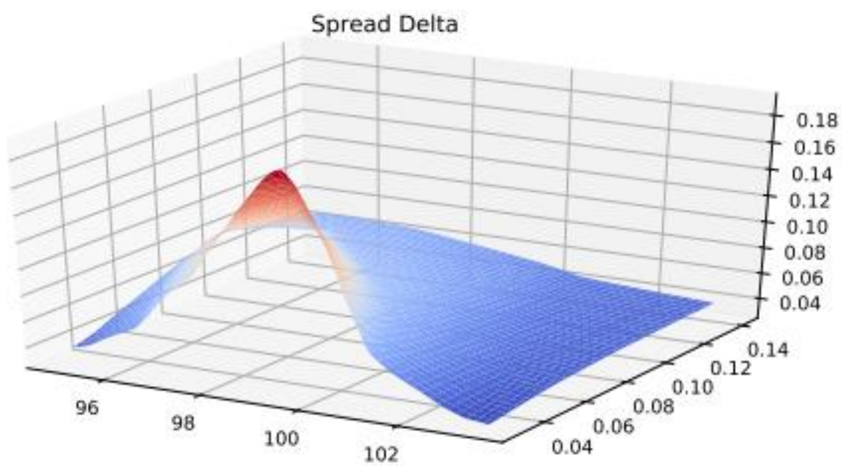
- Delta-hedging a call option with transaction costs in a Heston world
- No theoretical result

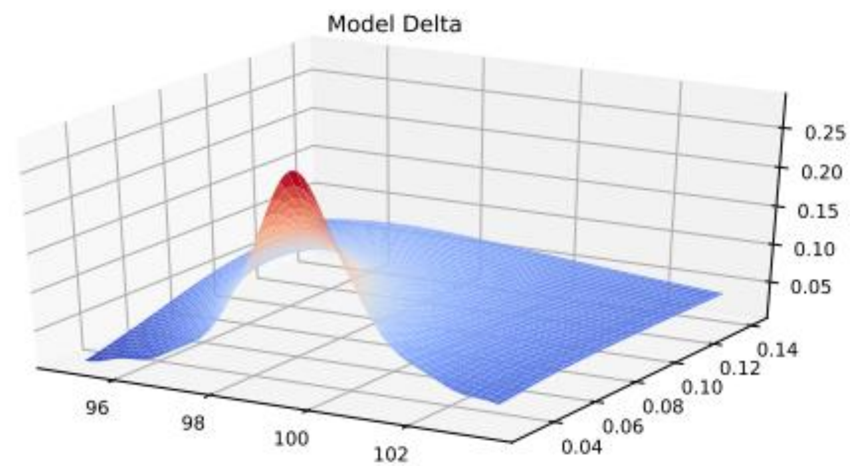


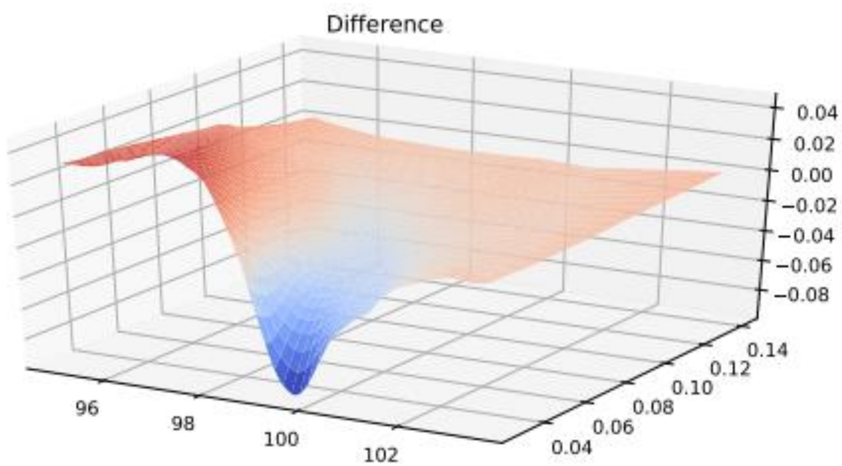
Toy examples – visualization

Example: Delta-hedging a call option in a Heston world

- Compare with the risk-neutral model result





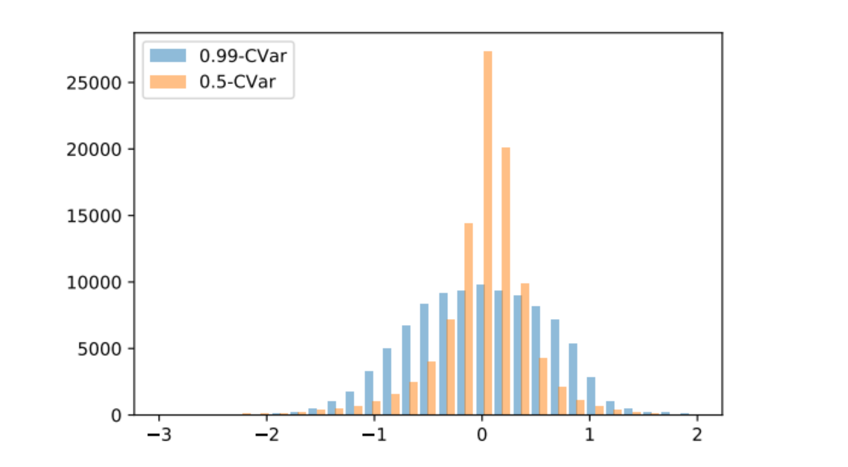


Performance metrics

J.P.Morgan

Out of sample performance

- PnL distribution
- Expected costs
- Easy to generate additional data in toy model settings
- Harder in the real world



Finding ways to understand hedging behaviour

- Visualisation becomes more challenging as the number of hedge instruments increases
- Plot quantiles of actions / cross-sections of network output

Beyond toy models

- Training data is the next challenge

Moving to the real world

J.P.Morgan

Do we have enough real-world data?

- Let's say we want to learn to hedge a product with 1Y maturity
  - Hedge frequency will usually be daily
- With 10Y of historical option price data, we have 10 fully independent paths
- Even if we allow overlapping paths, we only have ~2500 samples
- **Not enough data** to train a network hedger directly

What can we do?

- Create realistic synthetic data
- This means building market simulators
- For our equity derivatives applications, we need to learn to simulate the entire vanilla option market

Information content

- A simulator trained on historical data adds no new information
- Instead if provides controlled, independently testable, smooth interpolation of the data
- Could be overlaid with additional features by a human expert: alpha, tail risk

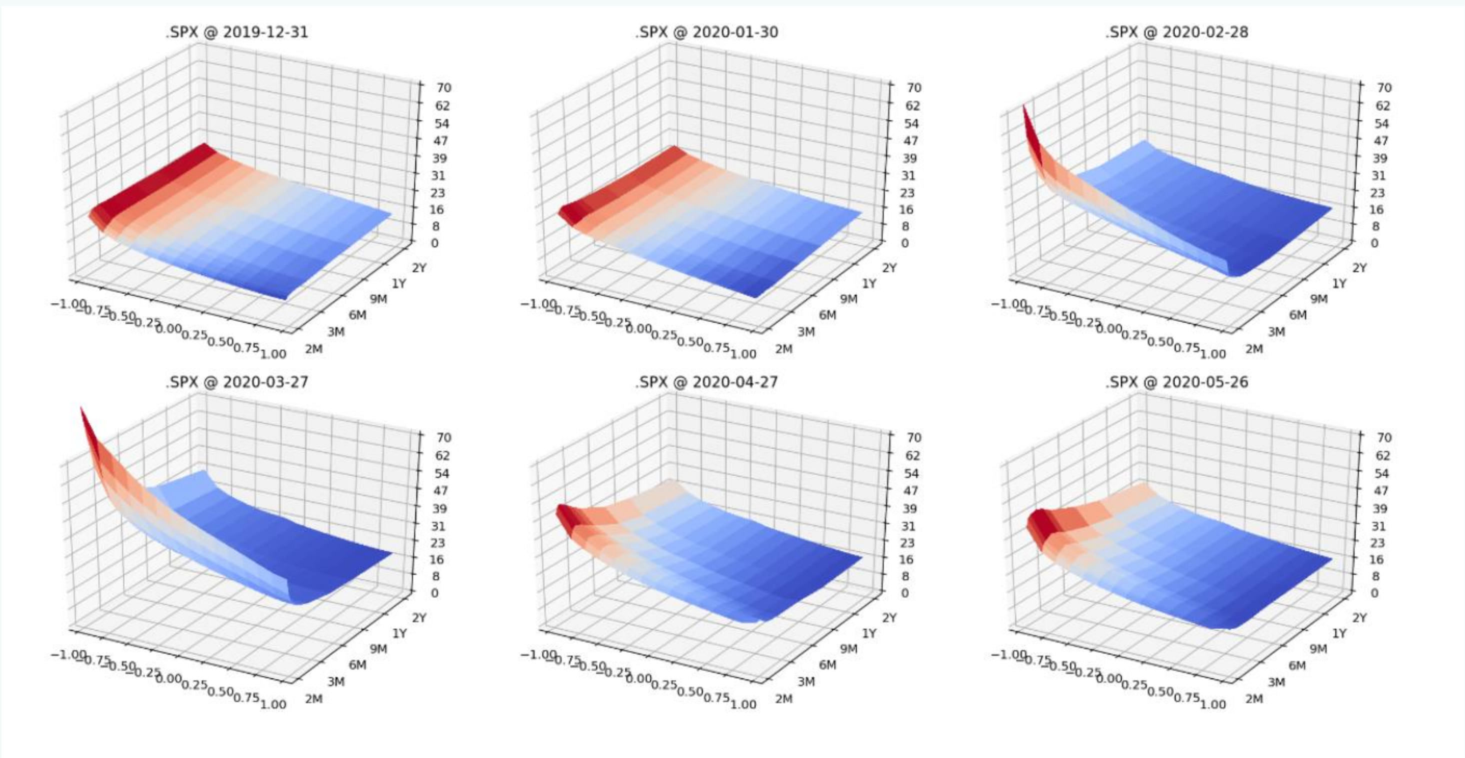
Equity option markets

What does the option market look like?

- For major indices, thousands of listed call and put options with different **strikes** and **maturities** are available and traded in volume at any time
- We will aim to generate realistic daily time series for a coarser grid, e.g. 4×9

Historical option price grids for SPX

- Option prices are conventionally described in terms of implied volatilities



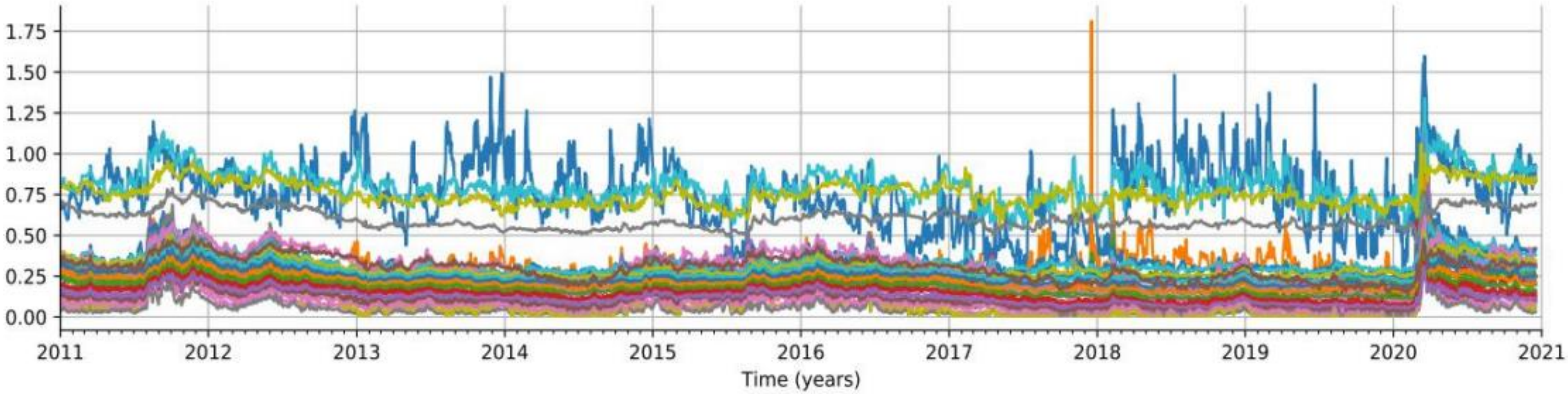
Equity option markets

J.P.Morgan

Volatility time series

- Skew / smile
- Clustering – high autocorrelation
- Mean reversion
- High cross-correlation of levels and returns
- Generally negative correlation of returns with spot

Time series of reparameterized volatilities for SX5E options



J.P.Morgan

Market simulation

A simulator generates the next state in a realistic way

- Historical market states $(x_t)_t \sim p$
- Build a network-based simulator G_θ

$$X_{t+1} = G_\theta\left(Z_t; x_t, \dots, x_{t-p+1}\right) \sim p_\theta(\cdot; x_t, \dots x_{t-p+1})$$

- Takes a historical state x_t and iid noise Z_t, and generates a new random state X_{t+1}
- Objective: calibrate G_θ such that p_θ is "close" to p

Measuring realism

- Use a range of performance metrics to assess simulated data against historical
  - Distributional metrics
    - Unconditional moments, density, cross-correlations
  - Dependency metrics
    - Autocorrelation of returns, levels

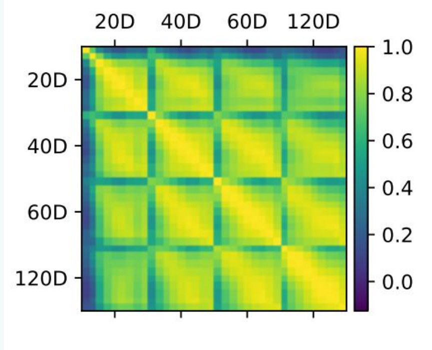
Challenges

- High dimension of x_t, e.g., 40
- Small dataset: ~2500 samples
- State dependence: next step is conditonal on current state
- Tail sampling: by definition, tail events are rare and hard to sample

Compression

Address the high dimension of the data

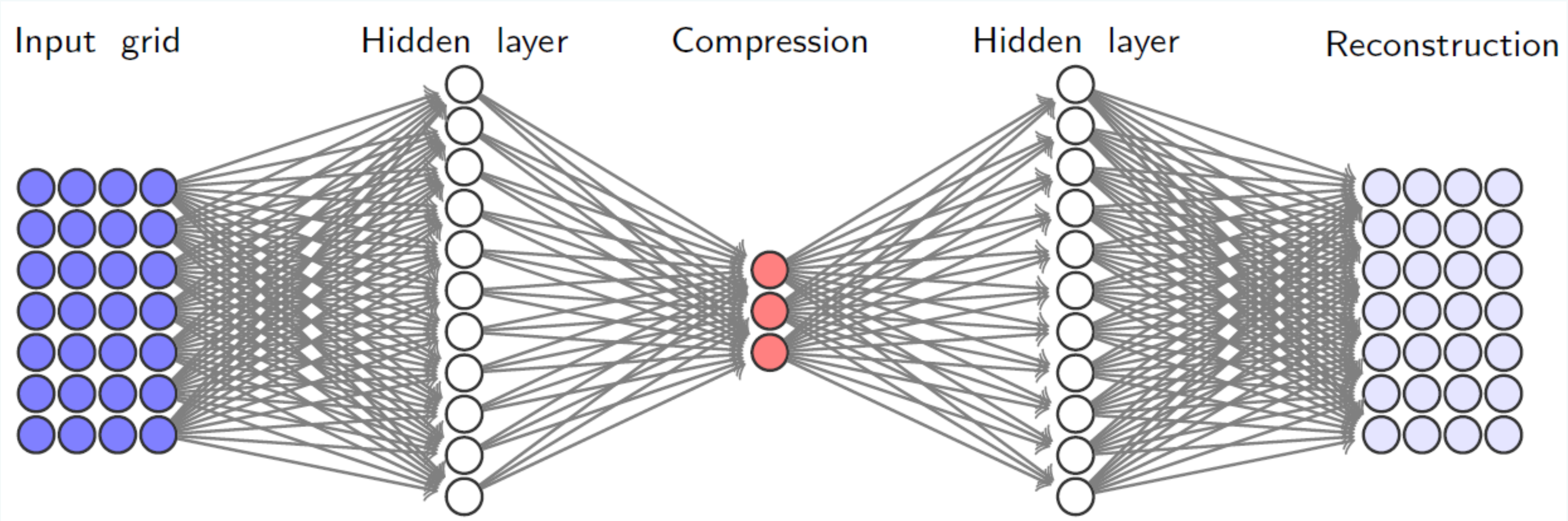
- Exploit the high cross-correlation and look for a low-dimensional representation
- Any encoding should be invertible – we need to simulate the full grid
- Minimize information loss under encoding/decoding round trip
- Target a “nice” distribution of samples in latent space



Autoencoder structure

- Image loss objective:

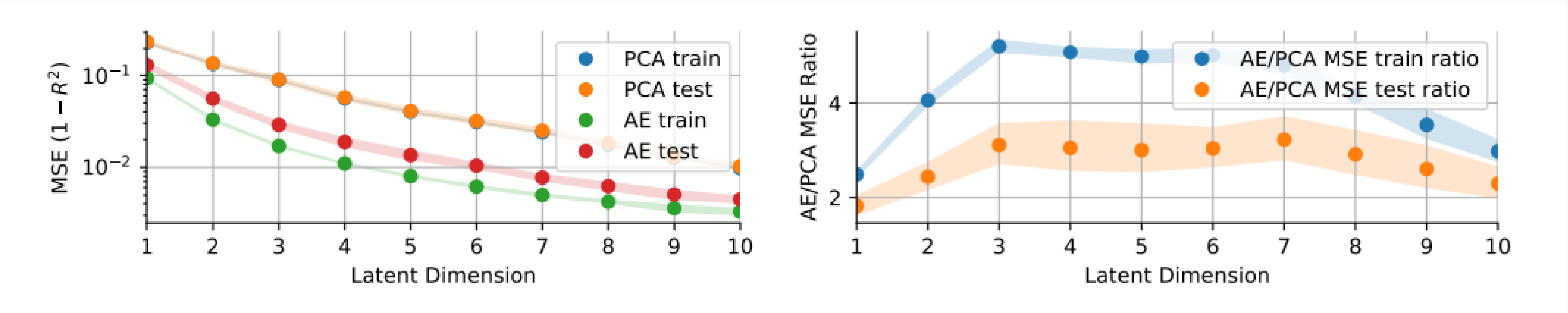
$$\mathcal{L} = \sum_i \left(x_i - D_\theta\left(E'_\theta(x_i)\right)\right)^2$$



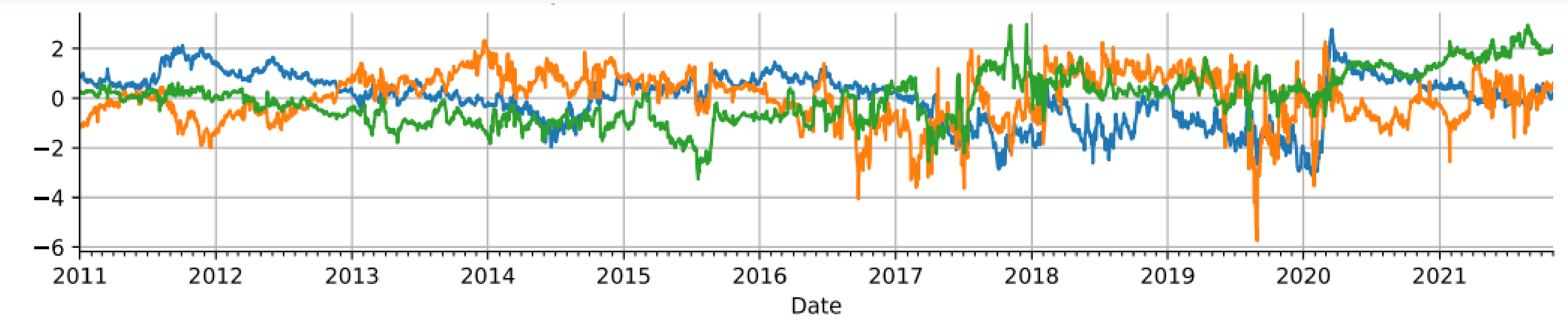
Compression

Autoencoder performance

- Linear compression (PCA) provides a baseline
- Network-based compression is approximately twice as efficient



- Compressed representation time series:



Time series generation

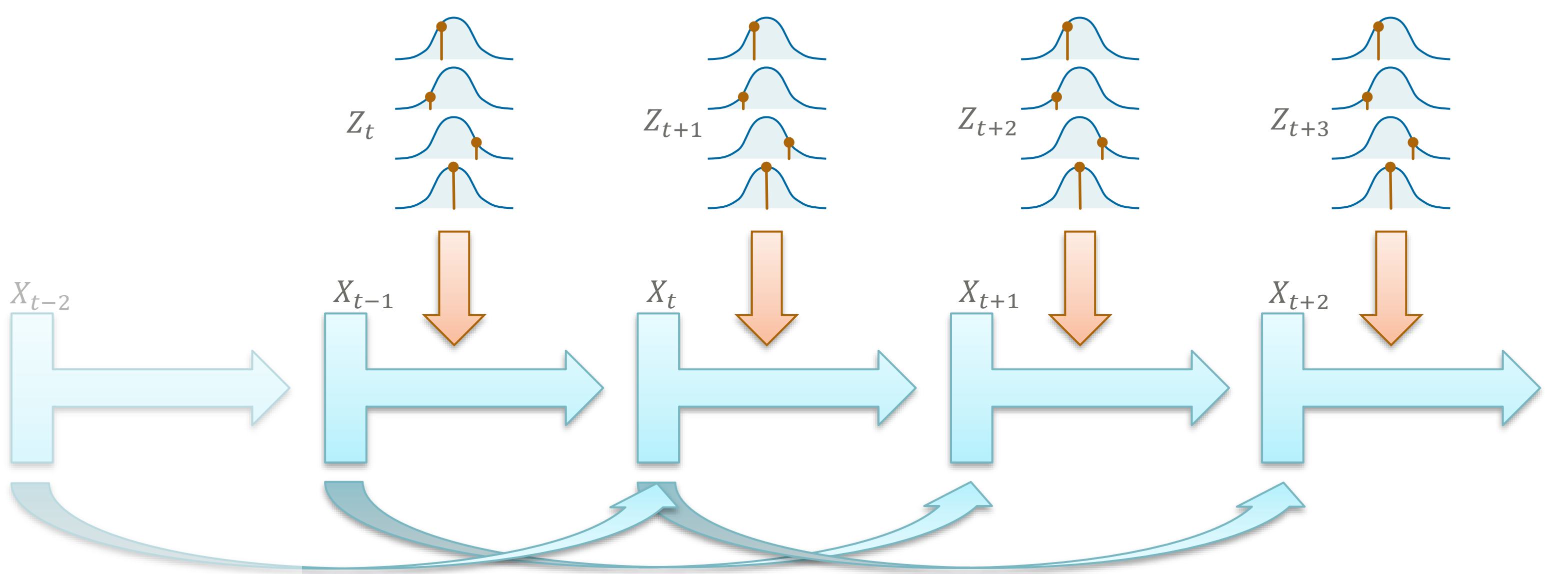
J.P.Morgan

Generator structure

- Build a network-based simulator G_θ for encoded state variables y_t

$$Y_{t+1} = G_\theta\left(Z_t; y_t, \dots, y_{t-p+1}\right) \sim p_\theta(\cdot; y_t, \dots y_{t-p+1})$$

- The generator takes the current state and a sample of iid noise, and gives the next state
- Repeatedly applying the generator allows us to build time series
- Classical approaches: VAR / GARCH
- Network-based generators can offer better performance



Time series generation

J.P.Morgan

Generator training

- Requires some measure of distributional distance between real and generated paths

Network-based generators

- Generative adversarial networks (GANs)

$$\min_G \max_D \mathbb{E}_{Y_{0,p}\sim\mu}\left[\mathbb{E}_{\mu}\left[\ln\left(D(Y)\right)|Y_{0,p}\right] + \mathbb{E}_{\nu(G)}\left[\ln\left(1 - D(Y)\right)|Y_{0,p}\right]\right]$$

- Conditional Signature Wasserstein distance

$$c\mathcal{W}^{\mathrm{Sig}}(p, p_\theta) = \mathbb{E}_p\left(\left\|\mathbb{E}_p\left[\mathcal{S}\left(Y_{t+1,t+q}\right)|Y_{t'\le t}\right] - \mathbb{E}_{p_\theta}\left[\mathcal{S}\left(Y_{t+1,t+q}\right)|Y_{t'\le t}\right]\right\|_2\right)$$

- Normalizing flows

$$\mathrm{KL}(p, p_\theta) = -\mathbb{E}_p[\mathbb{E}_p[\ln p(Y_{t+1}|Y_{t'\le t}) - \ln p_\theta(Y_t|Y_{t'\le t})\,|Y_{t'\le t}]]$$

Signature Wasserstein-1

J.P.Morgan

Conditional Signature Wasserstein distance

- The **Wasserstein distance** is a measure of distance between distributions ("earth mover's distance")

$$\mathcal{W}(p, p_\theta) = \sup_f \mathbb{E}_p[f(Y)] - \mathbb{E}_{p_\theta}[f(Y)]$$

- To compare distributions of paths, we can use a related metric based on path **signatures**
  - The signature is a path transformation using iterated integrals, with powerful properties
  - In particular, if two processes have the same expected signature, they have the same law
- The Signature Wasserstein-1 metric is

$$\mathcal{W}^{\mathrm{Sig}}(p, p_\theta) = \left\| \mathbb{E}_p[\mathcal{S}(Y)] - \mathbb{E}_{p_\theta}[\mathcal{S}(Y)] \right\|_2$$

- For our generator, we need the **conditional** Signature Wasserstein-1 distance

$$c\mathcal{W}^{\mathrm{Sig}}(p, p_\theta) = \mathbb{E}_p\left(\left\| \mathbb{E}_p\left[\mathcal{S}\left(Y_{t+1,t+q}\right) \middle| Y_{t' \le t}\right] - \mathbb{E}_{p_\theta}\left[\mathcal{S}\left(Y_{t+1,t+q}\right) \middle| Y_{t' \le t}\right] \right\|_2 \right)$$

- We still need to estimate the two conditional expected signatures
  - We can generate Monte Carlo estimates for the latter, and use regression for the former

Normalizing flows

J.P.Morgan

Generating a distribution

- The previously presented generators produce a new state sampled from the optimized conditional distribution
- If we can generate a **distribution** instead, then we can more efficiently compute a standard distributional distance like the KL-divergence
- Another desirable property for a generator is invertibility, i.e. we can compute

$$Z_t = G_\theta^{-1}(Y_{t+1}; y_t, \dots . y_{t-p+1})$$

- Being able to back out the driving noise is useful
  - Validate distributional assumptions
  - Introduce noise structure between two or more simulators
- A **normalizing flow** generator gives us these properties

Normalizing flows

J.P.Morgan

Objective function

- We consider the series of conditional cumulative distribution functions for each element in turn

$$F_1(y_1) = \mathbb{P}[Y_1 \leq y_1]$$
$$F_k(y_k; y_1, .., y_{k-1}) = \mathbb{P}[Y_k \leq y_k | Y_1 = y_1, \dots, Y_{k-1} = y_{k-1}]$$

- Note that indices here refer to elements, not time
- The conditional CDFs are invertible
- We can use an efficient **linear neural spline** representation, which allows us to compute $p_\theta(Y_t|\mathcal{F}_t)$
- To train the generator, we can use the expected conditional KL divergence, which we can now evaluate using a simple Monte Carlo estimate

$$\mathrm{KL}(p, p_\theta) = -\mathbb{E}_p\left[\mathbb{E}_p[\ln p_\theta(Y_{t+1}|Y_t)\,|Y_t]\right] + \mathrm{const} \approx -\sum_t^T \ln p_\theta(y_{t+1}|y_t) + \mathrm{const}$$

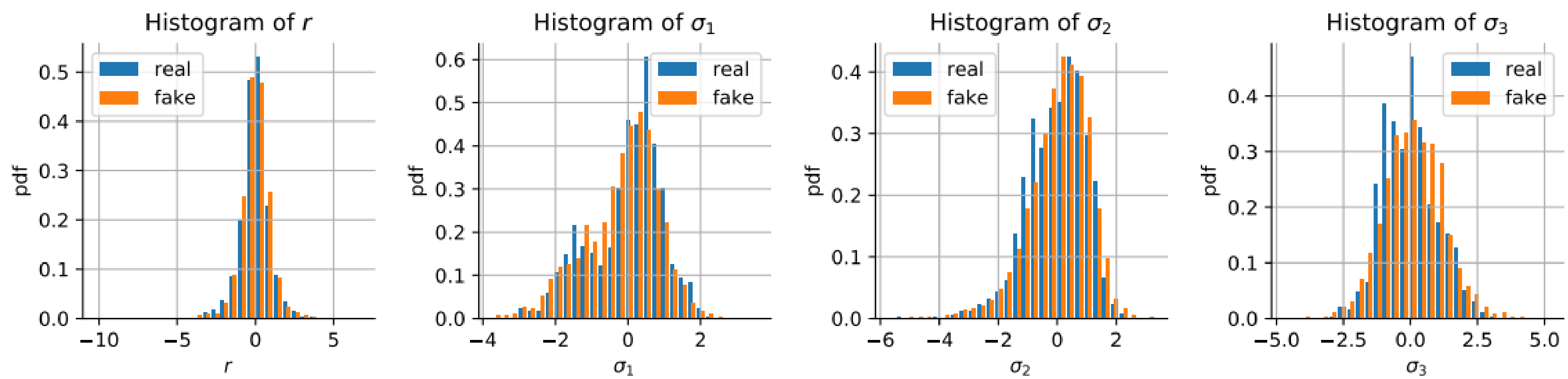
Challenge

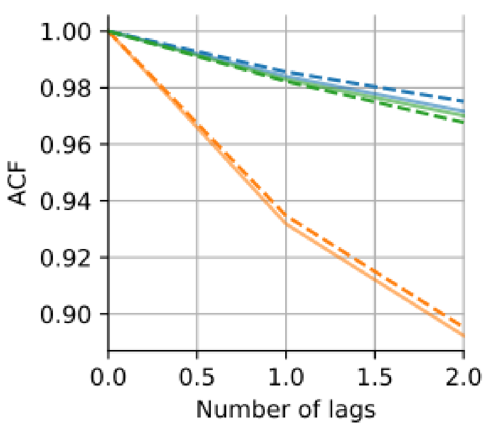
- As the dimension increases, the later conditional distribution functions become noisier and harder to estimate

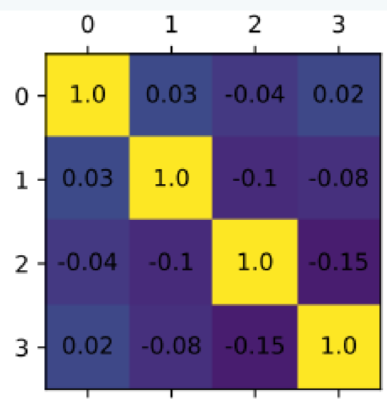
Normalizing flow simulator performance

J.P.Morgan

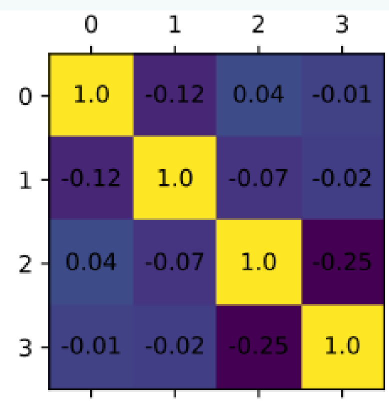
Level process metrics







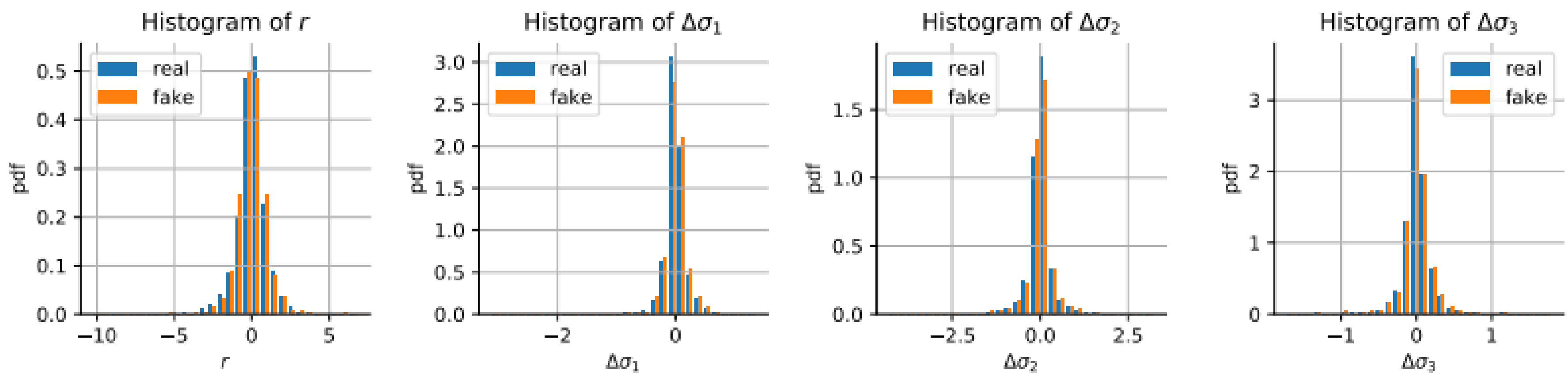
Historical cross-correlation matrix

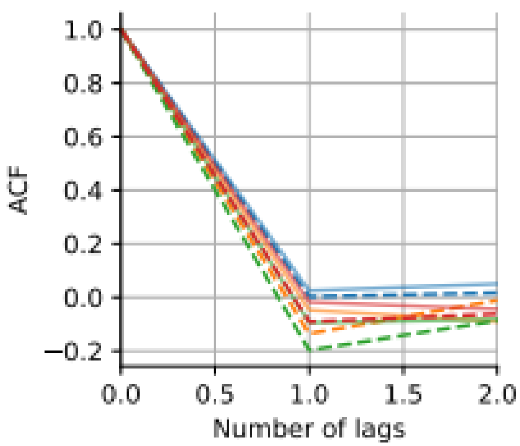


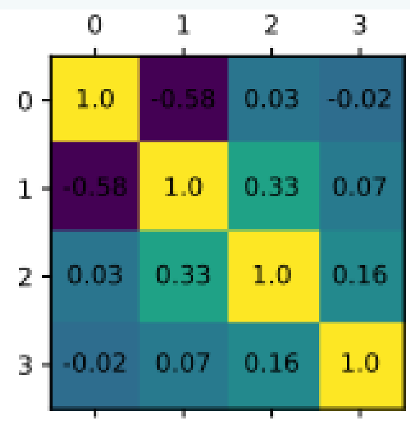
Generated cross-correlation matrix

Normalizing flow simulator performance

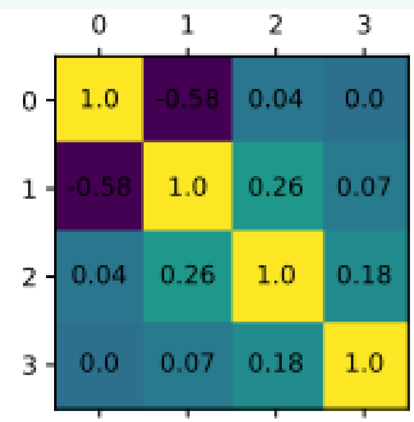
Returns process metrics







Historical cross-correlation matrix

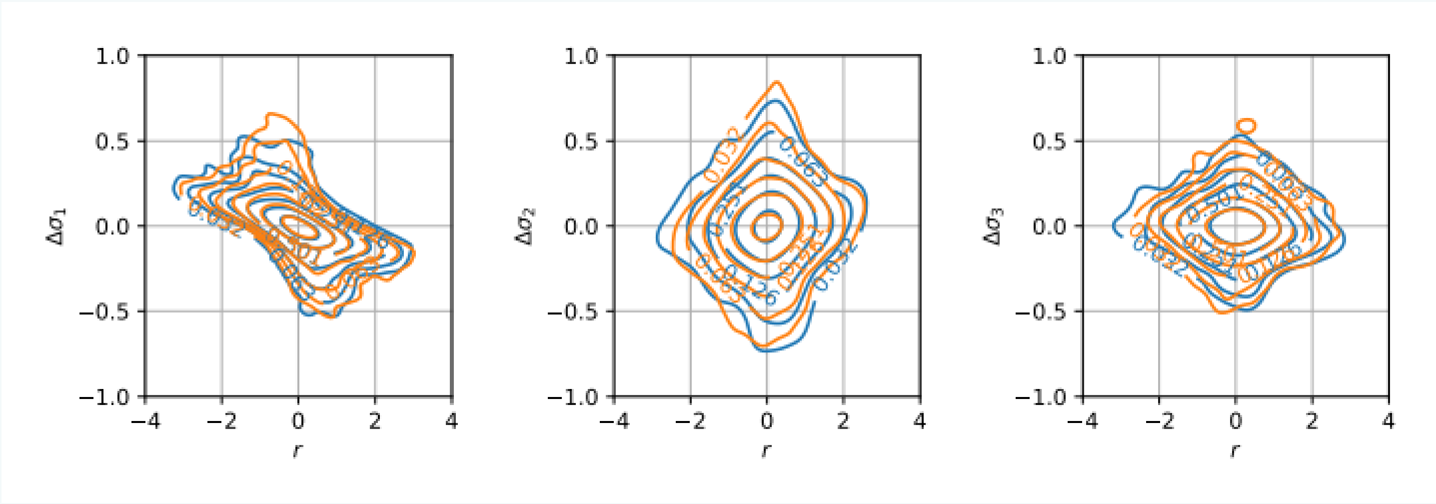


Generated cross-correlation matrix

Normalizing flow simulator performance

J.P.Morgan

Joint return distributions



Extrapolation problem

- We consistently find that a small but non-zero proportion of generated paths need to be rejected: this is a consequence of **extrapolation**
- Controlling this behaviour is surprisingly difficult

Simulator research

- Multi-asset simulation, asset classes beyond equities, realized volatility, tail sampling, arbitrage control

Beyond policy search

J.P.Morgan

Value function approach

- More familiar to Reinforcement Learning practitioners
- Necessary if we want to go beyond managing a fixed portfolio to expiry
- Naturally leads towards **universal** hedging

Bellman equation for Deep Hedging

- Most RL problems involve simple maximization of expected rewards, without risk aversion
  - In this setting, we can easily write the value function in terms of a **Bellman equation**

$$V^\pi(s) = \mathbb{E}[G^\pi|s] = \mathbb{E}[R^\pi(s) + V^\pi(s')|s] = R^\pi(s) + \mathbb{E}[V^\pi(s')|s]$$

- But hedging requires risk-aversion:

$$V^\pi(s) = U(G^\pi|s)$$

- Most reasonable utility functions are not **time consistent**

$$U(G^\pi) \neq U\big(U(G^\pi|s'')\big), \qquad U \neq \text{exponential utility}$$

- However, exponential utility does have this property, which leads to a Bellman equation

$$V^\pi(s) = U\big(R^\pi(s) + G^\pi(s')\big) = U\big(R^\pi(s) + V^\pi(s')\big), \qquad U = \text{exponential utility}$$

- The optimal policy maximizes the utility

$$V^{\pi^*}(s) = \sup_\pi U\big(R^\pi(s) + V^\pi(s')\big)$$

Implementing Bellman hedging

J.P.Morgan

Actor / critic

- We need to train two networks
  - The **actor** learns the optimal policy
    - The loss function is related to the negative utility

$$\mathcal{L}^A = \mathbb{E}\left[e^{-\lambda\left(R^{\pi}(s)+V^{\pi}(s')-V^{\pi}(s)\right)}\right]$$

  - The **critic** learns the value function
    - The loss function is related to the **temporal difference** error

$$\mathcal{L}^C = \mathbb{E}\left[e^{-\lambda\left(R^{\pi}(s)+\bar{V}^{\pi}(s')-V^{\pi}(s)\right)} - \lambda V^{\pi}(s)\right]$$

- Alternately update actor and critic
  - Apply averaging to stabilize critic updates

State dependence

- Both actor and critic depend on market and portfolio state
- We will use a portfolio state representation that mimics human trading

Hedging like a trader

J.P.Morgan

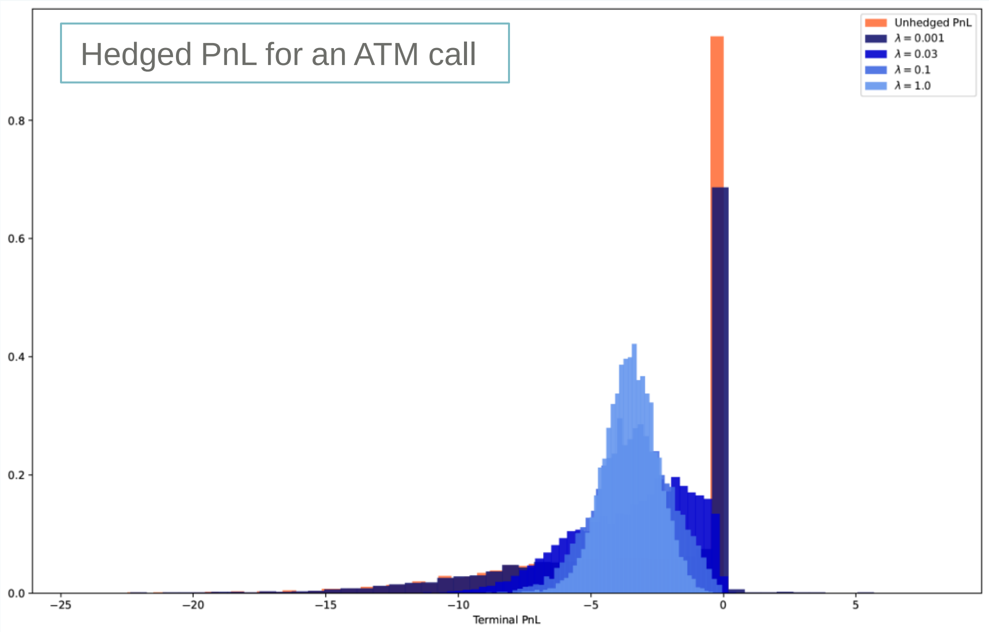
Towards universal hedging

- Learn the optimal hedge for a general class of options and a range of risk aversion levels
- Provide the agent with prices and greeks from a simple, **wrong** model
- The agent must learn to correct the greeks and apply a suitable risk adjustment on top

Toy example

- In a world with stochastic volatility, learn to delta-hedge a class of vanilla options using Black-Scholes greeks

Impact of risk aversion on hedged PnL



Details

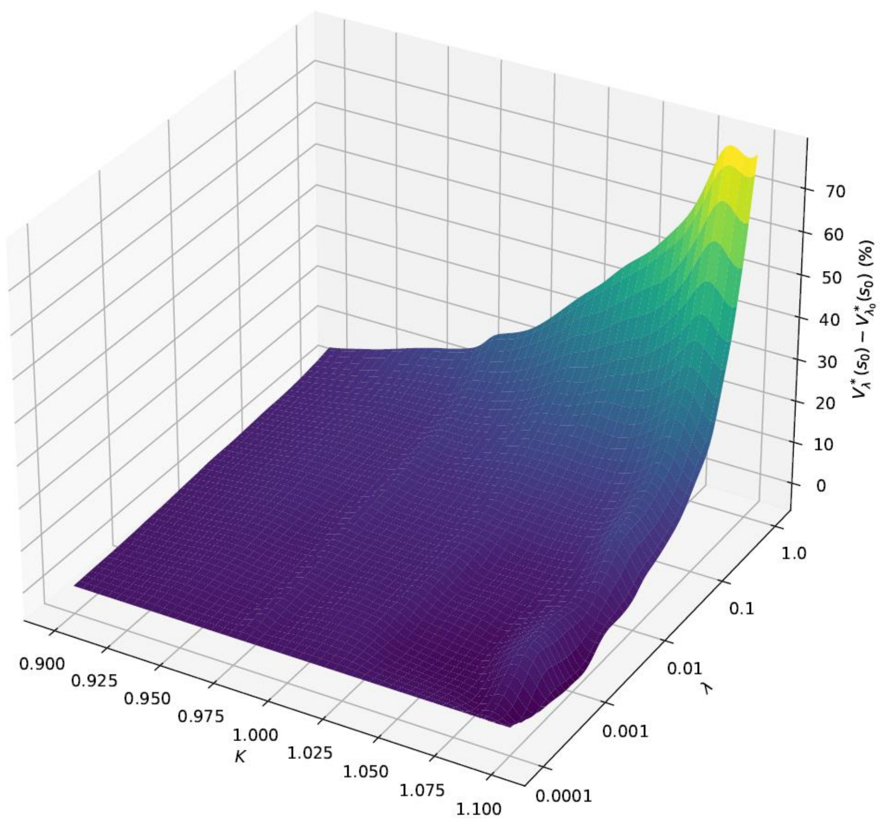
- Heston model world
- Provide Black-Scholes greeks as features
- Train on fixed-maturity, fixed-notional options with different strikes (one option at a time)
- Hedge with spot only
- Proportional transaction costs
- Finite time horizon (1M)
- Note: no statistical arbitrage

General risk-adjusted hedging

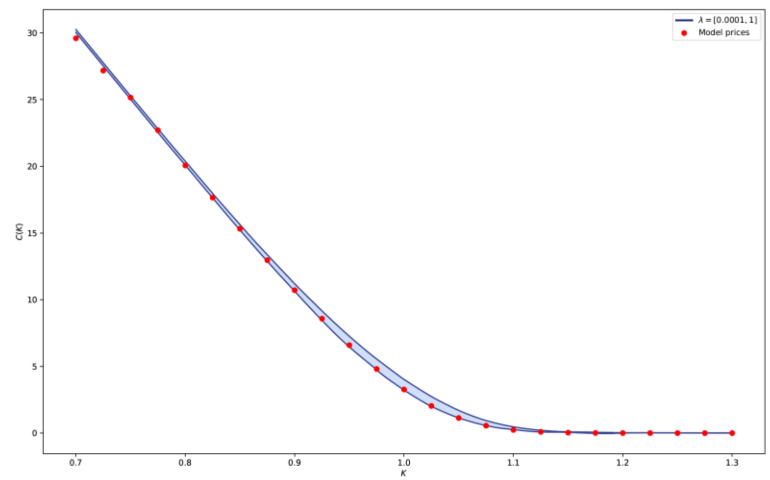
J.P.Morgan

Impact of risk aversion on values

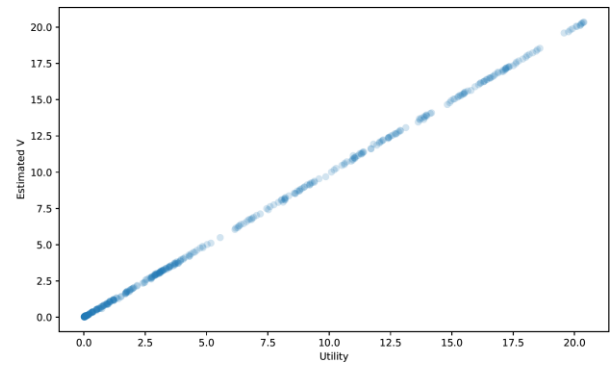
- The risk adjustment of an option position increases with risk aversion
- Converge to the risk-neutral model price as $\lambda \to 0$



Relative option value adjustment as a function of option strike and risk aversion



Option value adjustment as a function of strike, for different risk aversion levels



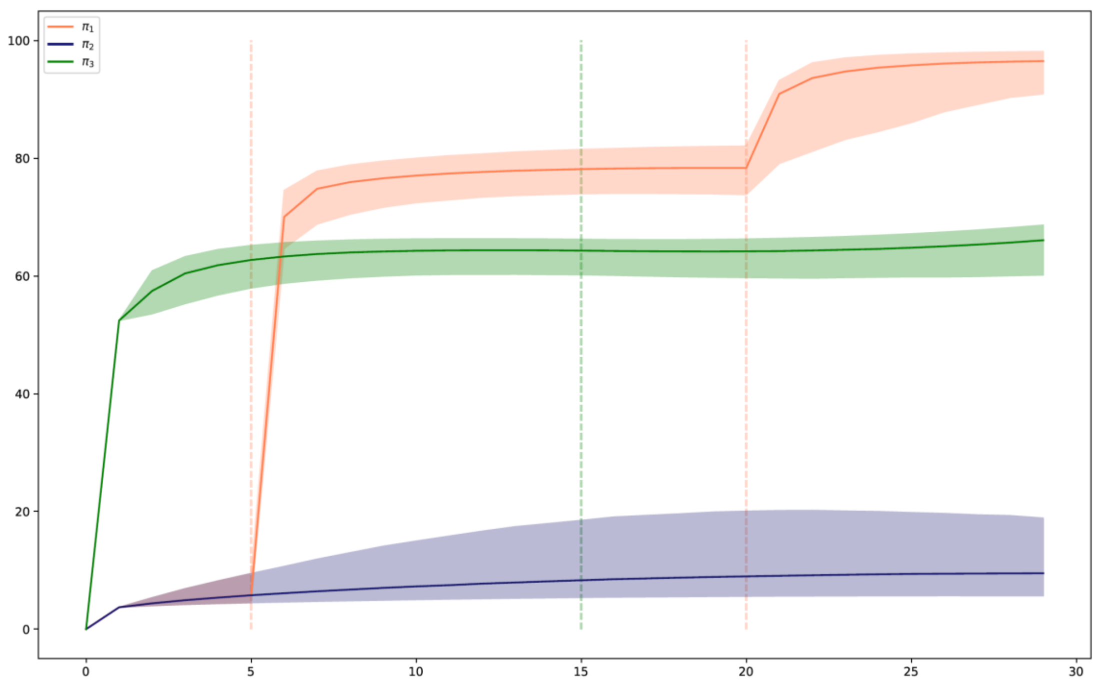
Critic performance: predicted vs measured utility

General risk-adjusted hedging

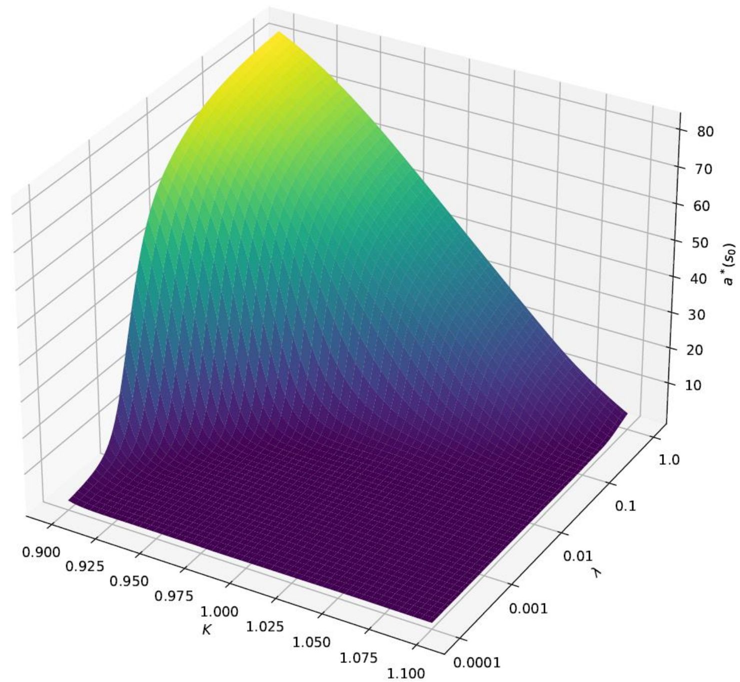
J.P.Morgan

Impact of risk aversion on actions

- Hedging activity smoothly increases with risk aversion
- If we change our risk aversion mid-strategy, the agent adapts



Changing risk aversion



Initial hedge as a function of option strike and risk aversion

Conclusion

J.P.Morgan

Deep Hedging

- Formulate hedging a derivatives portfolio as a reinforcement learning problem
- Use a loss function that penalizes risk
- Represent the hedging strategy as a neural network
- Solve the episodic problem with direct policy search

Market simulation

- Realistic synthetic data is necessary to train a Deep Hedging agent effectively
- Networks allow efficient encoding/decoding and realistic time series generation

Bellman hedging

- Points the way to more general, more powerful Deep Hedging agents
- We can learn the optimal risk-adjusted hedge from the greeks of a simple, wrong model

References

J.P.Morgan

Deep Hedging

- **Deep Hedging**, Hans Buehler, Lukas Gonon, Josef Teichmann, Ben Wood
  https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3120710
- **Deep Hedging: Hedging Derivatives Under Generic Market Frictions Using Reinforcement Learning**, Hans Buehler, Lukas Gonon, Josef Teichmann, Ben Wood, Baranidharan Mohan, Jonathan Kochems
  https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3355706

Market simulation

- **Multi-Asset Spot and Option Market Simulation**, Magnus Wiese, Ben Wood, Alexandre Pachoud, Ralf Korn, Hans Buehler, Phillip Murray, Len Bai https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3980817
- **Deep Hedging: Learning to Simulate Equity Option Markets**, Magnus Wiese, Len Bai, Ben Wood, Hans Buehler
  https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3470756

Bellman hedging

- **Deep Hedging: Continuous Reinforcement Learning for Hedging of General Portfolios across Multiple Risk Aversions**, Phillip Murray, Ben Wood, Hans Buehler, Magnus Wiese, Mikko Pakkanen
  https://arxiv.org/abs/2207.07467?context=stat.ML
- **Deep Bellman Hedging**, Hans Buehler, Phillip Murray, Ben Wood
  https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4151026

Statistical arbitrage

J.P.Morgan

Definition

- A market has statistical arbitrage if it admits a positive risk-adjusted return for an initially empty portfolio

$$\sup_a U\left(G^0(a)\right) > 0$$

  – Intuition: with no existing portfolio, the optimal action is not to do nothing

- For reasonable (risk-averse) utility functions:

$$\sup_a \mathbb{E}[G^0(a)] = 0 \quad \Rightarrow \quad \text{no statistical arbitrage}$$

- We can go further:

$$H_t - \gamma_t \leq \mathbb{E}[H_T|\mathcal{F}_t] \leq H_t + \gamma_t \quad \Leftrightarrow \quad \text{no statistical arbitrage}$$

- Here γ_t is the small-order-size limit of trading cost per unit price

Technical conditions

- Convex trading costs
  – Proportional in the small-trade limit
- Convex, bounded admissible actions
- Bounded tradable instrument values H_t

Utility functions

- Focus on exponential utility / entropic risk

$$U_\lambda(X) = -\frac{1}{\lambda}\log \mathbb{E}\left[e^{-\lambda X}\right]$$

- λ is the risk aversion

$$U_0(X) = \mathbb{E}[X]$$

Controlling statistical arbitrage

J.P.Morgan

Motivation

- Am I hedging or trading for profit?
- How well can I predict future statistical arbitrage opportunities?

Removing the drift

- Find a change of measure that removes statistical arbitrage opportunities
- We want the **minimal measure change**, with the smallest distance between $\mathbb{P}$ and $\mathbb{Q}$
- Remarkably, we can derive this measure directly from the optimal statistical arbitrage strategy under $\mathbb{P}$

$$\frac{d\mathbb{Q}^*}{d\mathbb{P}} = \frac{e^{-G^0(a^*)}}{\mathbb{E}\left[e^{-G^0(a^*)}\right]}$$

- $\mathbb{Q}^*$ is the closest martingale measure to $\mathbb{P}$ with respect to the relative entropy

$$H(\mathbb{Q}|\mathbb{P}) = \mathbb{E}^{\mathbb{P}}\left[\frac{d\mathbb{Q}}{d\mathbb{P}} \log \frac{d\mathbb{Q}}{d\mathbb{P}}\right]$$

- This is the **minimal entropy martingale measure**
- The result generalizes to other utility functions and to trading with transaction costs
- With transaction costs, we obtain a **near-martingale measure**

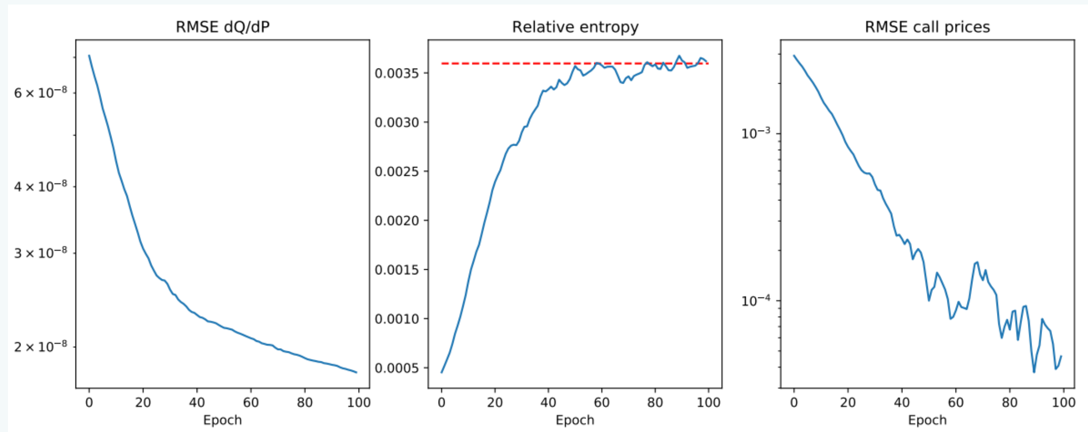
Drift removal – simple examples

Toy example: Black-Scholes model with spot price drift

- Trade spot
- Statistical arbitrage strategy in $\mathbb{P}$: **buy and hold**

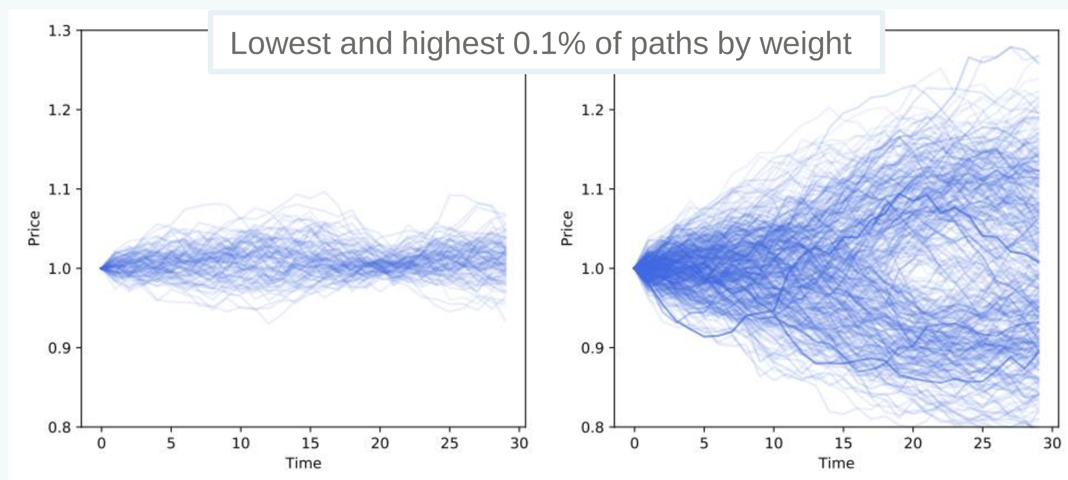
$$\frac{d\mathbb{Q}^*}{d\mathbb{P}} = \exp\left(-\frac{\mu}{\sigma}W_T - \frac{\mu^2}{2\sigma^2}T\right)$$

$$H(\mathbb{Q}^*|\mathbb{P}) = \frac{\mu^2}{2\sigma^2}T$$



Toy example: Black-Scholes model with implied volatility risk premium

- Trade spot and vanilla calls
- Statistical arbitrage strategy in $\mathbb{P}$: **sell options and delta hedge**
- Change of measure should upweight paths with high realized volatility, and downweight paths with low realized volatility



Removing drift from a realistic option market model

J.P.Morgan

VAR model for spot and option prices

- Fit a VAR(2) model to historical spot and (reparametrized) option prices for EURO STOXX 50
- Floating grid of relative-strike, relative-maturity options
- Check expected option payoffs against prices, before and after drift removal

Expected difference between option payoff and price

Remark

- On a discrete measure, the boundary between static and statistical arbitrage is less clear