Lucerne University of
Applied Sciences and Arts

HOCHSCHULE
LUZERN

New Frontiers in Quantitative Risk Management

IFZ FinTech Colloquium

Thomas Krabichler

November 27, 2019

Purpose of the Talk

Objective

This presentation is supposed to provide you with

- **selected challenges** that arise in the financial industry,
- an introduction to how these challenges can be tackled by means of **machine learning** techniques.

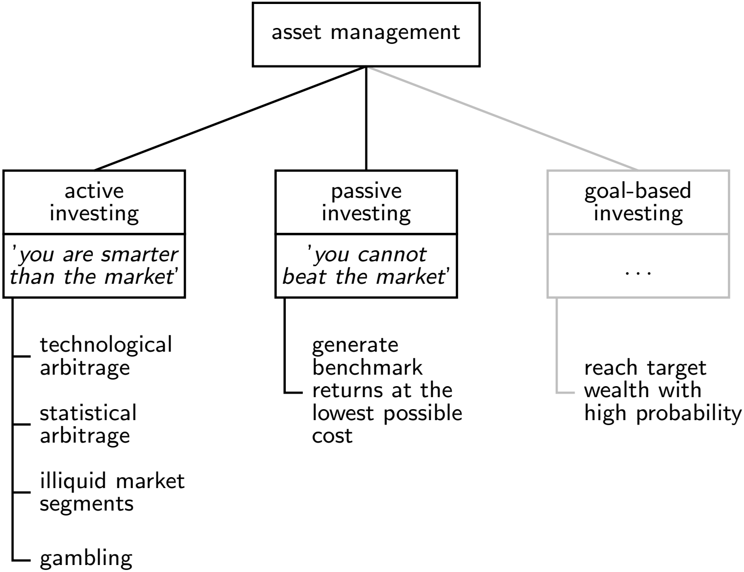
Disclaimer

- This introduction does **not** provide a **comprehensive** overview of how machine learning techniques are applied in the financial industry.
- The presented topics may grant an essential competitive advantage. However, please be aware of **inherent risks**.
- This talk does not disclose any profitable investment strategies.

Outline

1. Challenges
   - Asset Management
   - Pricing and (Over-)Hedging
2. Neural Networks
3. Machine Learning
   - Supervised Learning
   - Re-inforcement Learning
4. Applications

Asset Management



Valuation and (Over-)Hedging

What is a fair price $P(0, T)$ of getting one monetary unit at time $T > 0$ as seen from $t = 0$?

- naive approach:

$$P(0, T) = 1$$

  issues: **inflation risk**, **credit risk**, **liquidity risk**

- static approach:

$$P(0, T) = \frac{1}{(1 + r)^T}$$

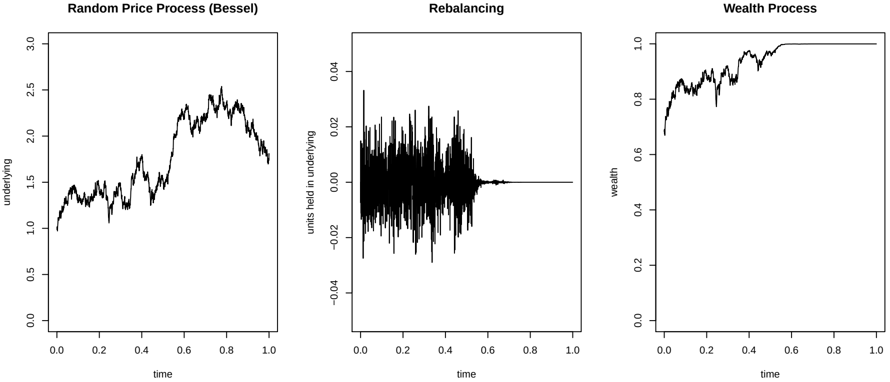
  for some interest rate r

Risk-Adjusted Valuation

$P(0, T)$ is the **minimal cost** to (super-)replicate the desired payoff.

Valuation and (Over-)Hedging

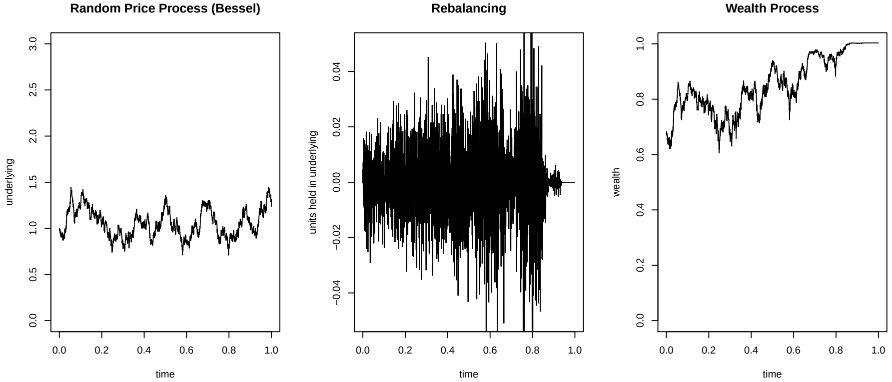
Monte-Carlo



- random price process $S_t = \sqrt{{W_{1,t}}^2 + {W_{2,t}}^2 + {W_{3,t}}^2}$
- (almost) frictionless (delta-)hedging results in minimal super-replication cost of 0.68

Valuation and (Over-)Hedging

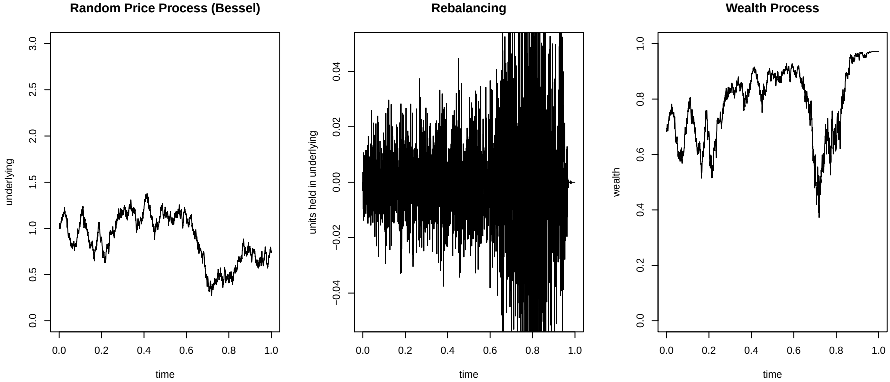
Monte-Carlo



- random price process $S_t = \sqrt{{W_{1,t}}^2 + {W_{2,t}}^2 + {W_{3,t}}^2}$
- (almost) frictionless (delta-)hedging results in minimal super-replication cost of 0.68

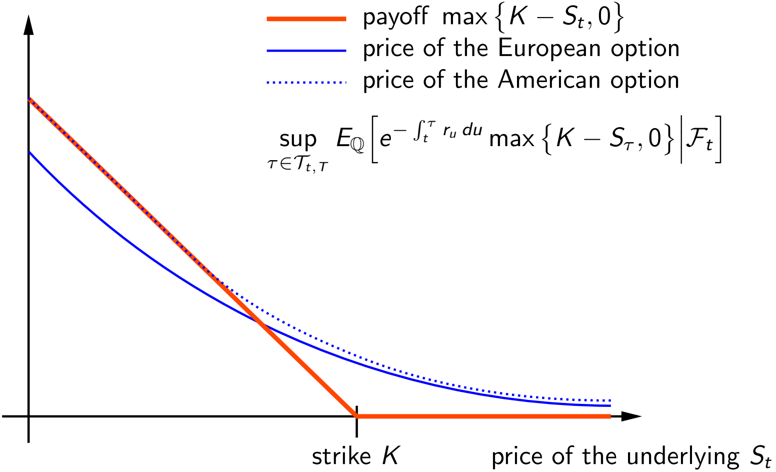
Valuation and (Over-)Hedging

Monte-Carlo



- random price process $S_t = \sqrt{{W_{1,t}}^2 + {W_{2,t}}^2 + {W_{3,t}}^2}$
- (almost) frictionless (delta-)hedging results in minimal super-replication cost of 0.68

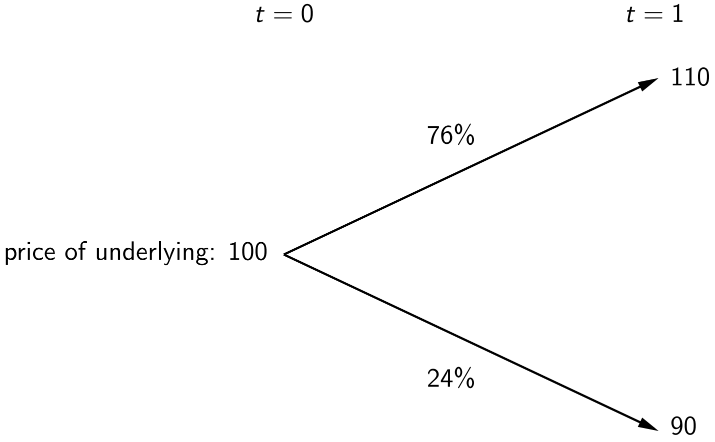
Option Pricing



Valuation and (Over-)Hedging

Dynamic Programming

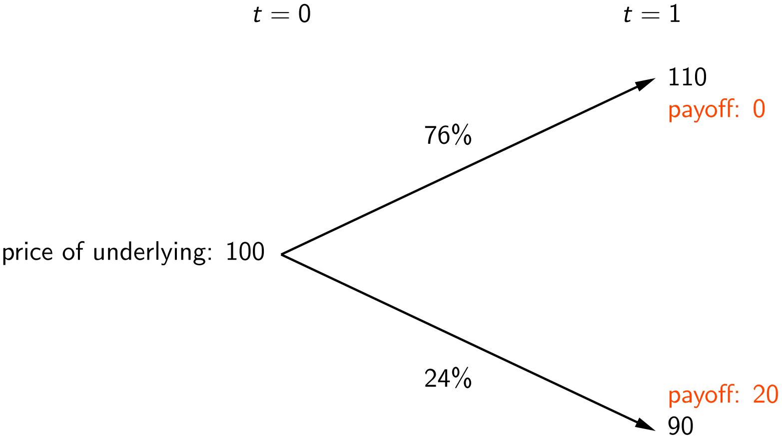
Discrete World: $K = 110$, $r = 5\%$



Valuation and (Over-)Hedging

Dynamic Programming

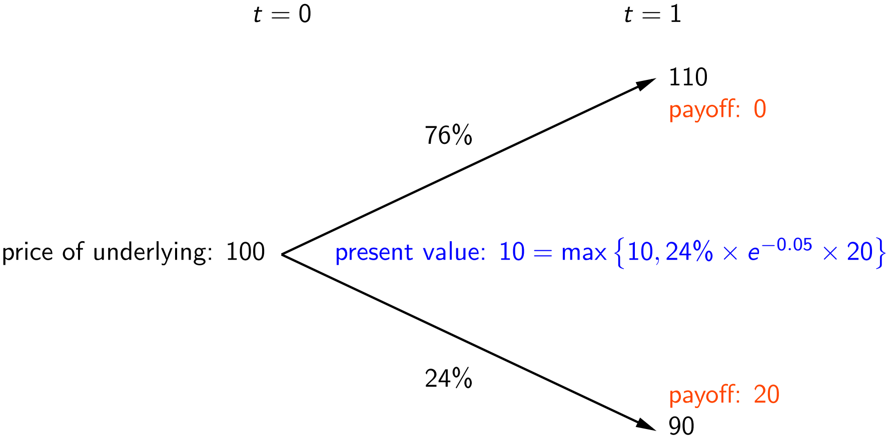
Discrete World: $K = 110,\ r = 5\%$



Valuation and (Over-)Hedging

Dynamic Programming

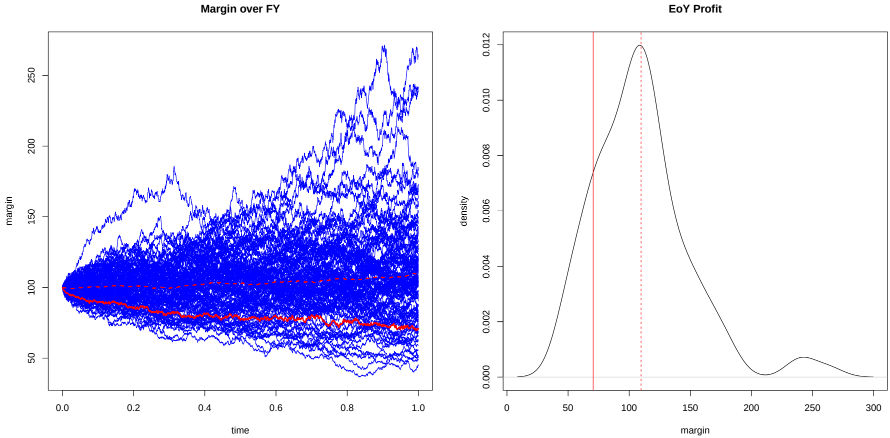
Discrete World: $K = 110$, $r = 5\%$



Valuation and (Over-)Hedging

Flaws of Classical Valuation Approaches

- Monte-Carlo-techniques or dynamic programming tend to be **computationally intensive**.
- The **level of sophistication** remains limited.



Valuation and (Over-)Hedging

The Curse of Dimension

No. of Underlyings	Discretisation of Space and Time	Runtime	Scale Unit
1	1 000	1	millisecond
2	1 000 000	1	second
3	1 000 000 000	17	minutes
4	10^{12}	12	days
5	10^{15}	32	years
6	10^{18}	317	centuries
⋮	⋮	⋮	⋮

- Longstaff-Schwartz (2001): 20 underlyings
- Becker-Cheridito-Jentzen (2018): 500 underlyings below 10 minutes with techniques inspired from **machine learning**

Valuation and (Over-)Hedging

Hierarchy of financial assets from the accounting and pricing viewpoint (according to FASB 157):

- **Level 1**: Quotes are readily observable in the market.
- **Level 2**: Prices can be inferred through models and observable quantities.
- **Level 3**: Valuations involve complex models and subjective assumptions.

A professional and well-calibrated valuation platform must meet the following requirements:

- The model reprices level 1 products.
- The model features generally observed market phenomena.
- The model accounts for the significant risk drivers in a realistic manner.

Valuation and (Over-)Hedging

Risk-Adjusted Valuation

What is a fair price π_0 of getting $h(S)$ at time $T > 0$ as seen from $t = 0$, where $S = (S_t)_{0 \leq t \leq T}$ is a d-dimensional underlying risk factor and h some payoff function?

- Finding **realistic dynamics** is almost impossible due to the statical uncertainty.
- The **(super-)replication strategy** is often not known explicitly.
- Trading off **complexity**, mathematical **tractability** and inherent **model risks** is very challenging.
- Analytically, it is very hard to deal with **transaction cost**.
- Maintaining and **automating** a suitable, efficient and well-calibrated valuation platform (e.g., stochastic local volatility models) for several thousand derivatives is tough.

The Game Has Changed

In 2017 a research group of DeepMind published the following results:

White	Black	Wins[3]	Draws	Losses
AlphaZero[1]	*Stockfish*	25	25	0
Stockfish[2]	*AlphaZero*	3	47	0

[1] AlphaZero is an algorithm that learns to play chess from scratch solely by **smart self-play**.

[2] Stockfish is a powerful open-source chess engine and TCEC world champion 2016.

[3] Outcome as seen from AlphaZero's perspective.

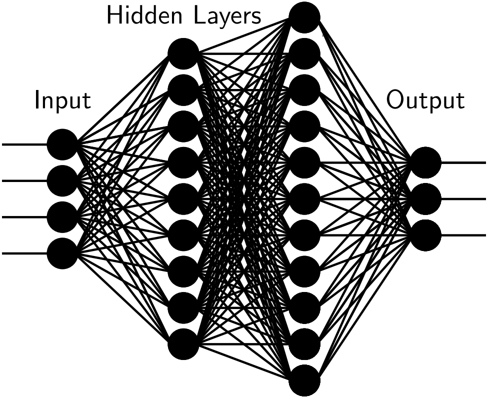
This results stimulates the imagination that quantitative methods for finance enter a new era.

Paradigm

Regarding the presented challenges, what would a **clever** financial agent with a lot of **experience** and a decent **risk appetite** do?

Outline

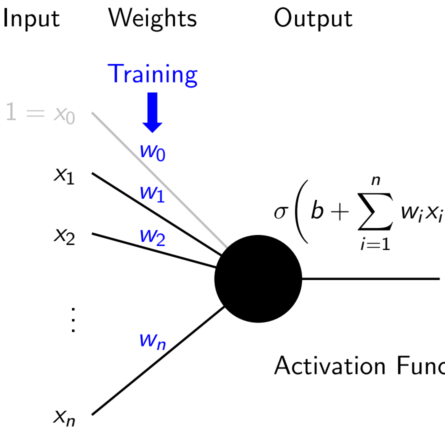
Neural Networks



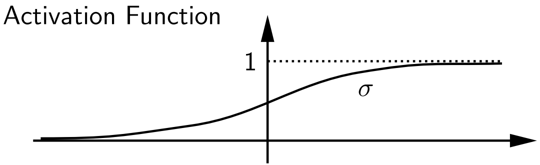
Machine Learning from the Mathematical Viewpoint

Simply put, it is the approximation of a high-dimensional non-linear function in terms of a (deep) neural network (DNN).

Perceptron



$$\sigma\left(b + \sum_{i=1}^{n} w_i x_i\right) = \sigma\left(\sum_{i=0}^{n} w_i x_i\right)$$



Neural Networks

Mathematical Properties

- **Universal Approximation Theorems**: Provided that they are sufficiently large, neural networks can approximate complex functions arbitrarily close.
- Computing the derivative of the network output with respect to the weights is straightforward. Therefore, an incremental **learning process** becomes feasible.

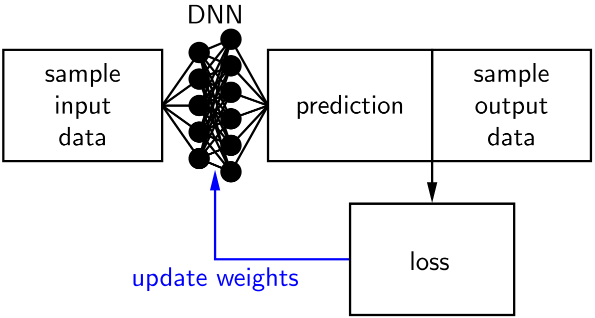
Outline

1. Challenges
   - Asset Management
   - Pricing and (Over-)Hedging
2. Neural Networks
3. Machine Learning
   - Supervised Learning
   - Re-inforcement Learning
4. Applications

Machine Learning

Supervised Learning

Training: Minimise a Loss Function



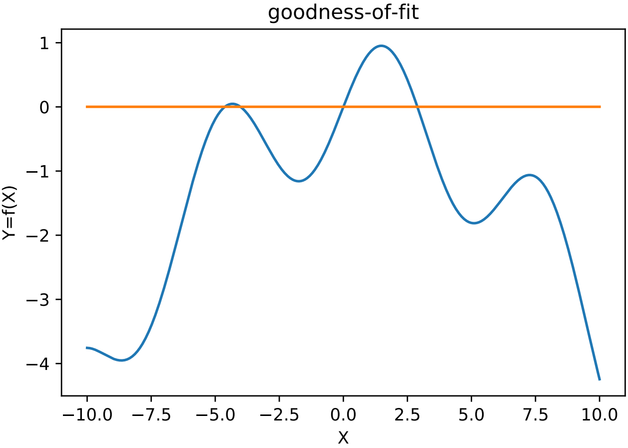
Validation: Check Accuracy of Prediction on Concealed Data

Machine Learning

Supervised Learning

Number of Nodes: 1–30–30–10–10–1

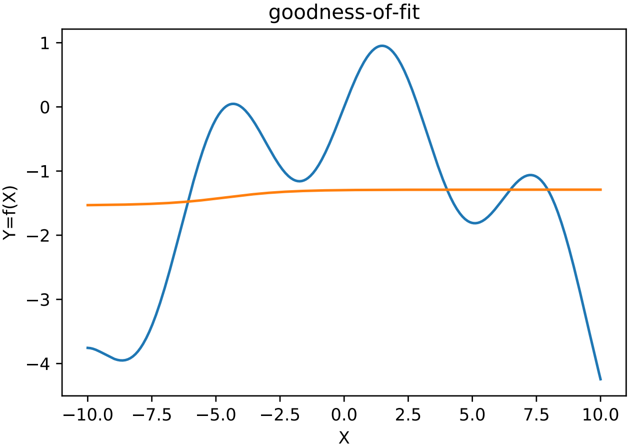
Number of Epochs: 0



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

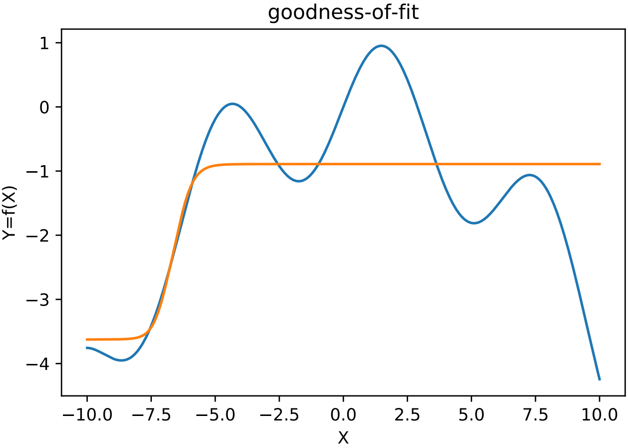
Number of Epochs: 1 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

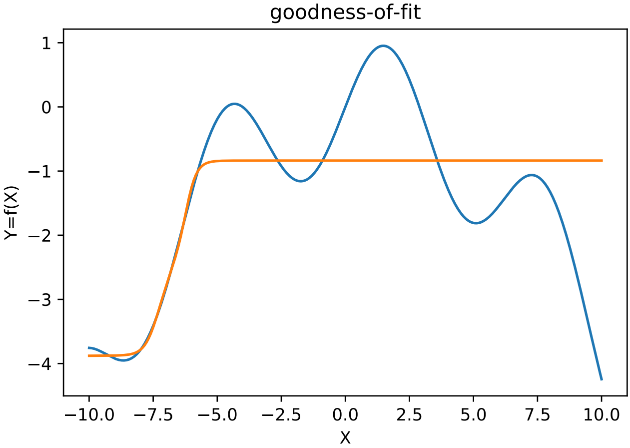
Number of Epochs: 2 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

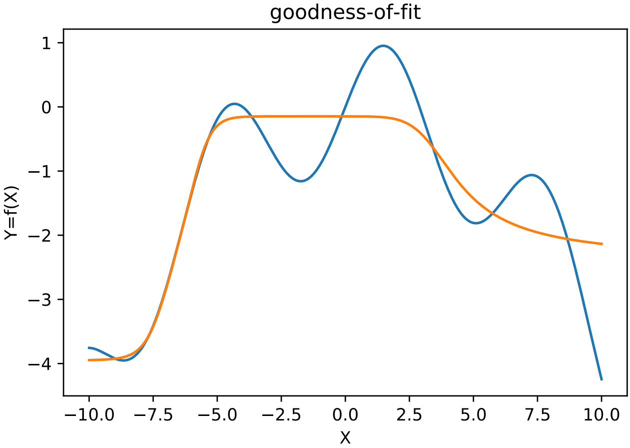
Number of Epochs: 3 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

Number of Epochs: 4 000

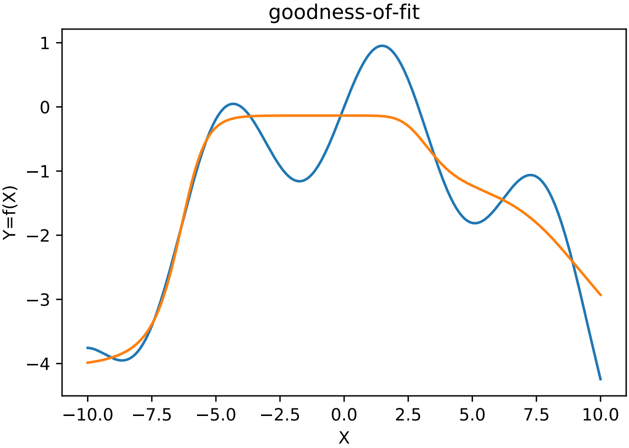


Machine Learning

Supervised Learning

Number of Nodes: 1–30–30–10–10–1

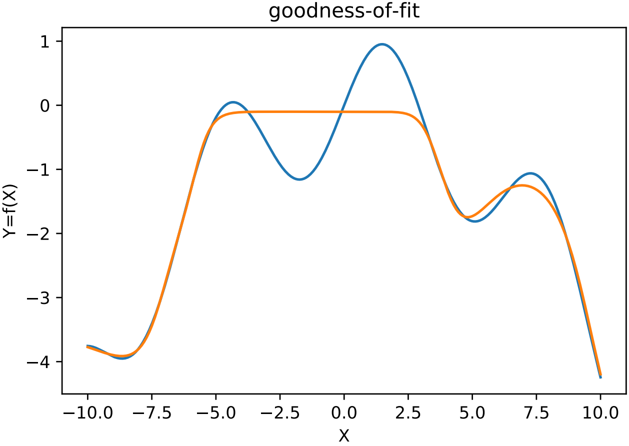
Number of Epochs: 5 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

Number of Epochs: 6 000

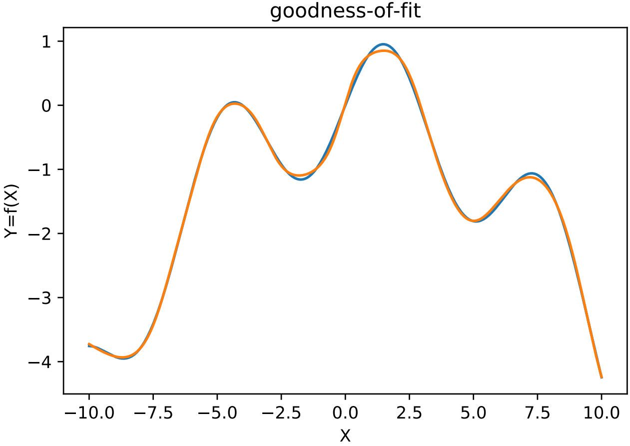


Machine Learning

Supervised Learning

Number of Nodes: 1–30–30–10–10–1

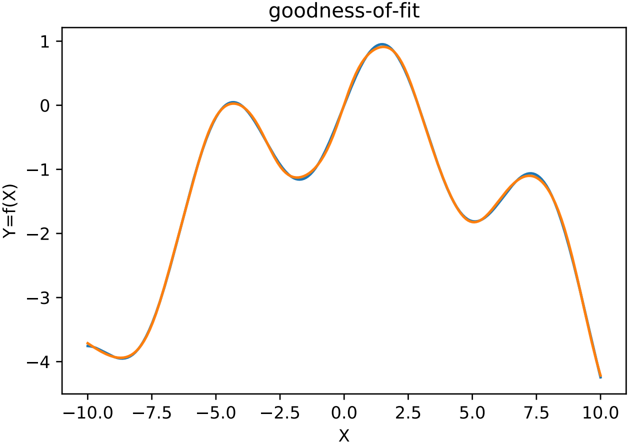
Number of Epochs: 7 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

Number of Epochs: 8 000

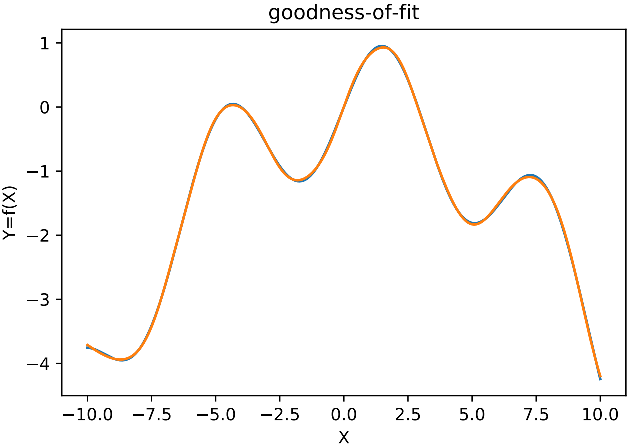


Machine Learning

Supervised Learning

Number of Nodes: 1–30–30–10–10–1

Number of Epochs: 9 000

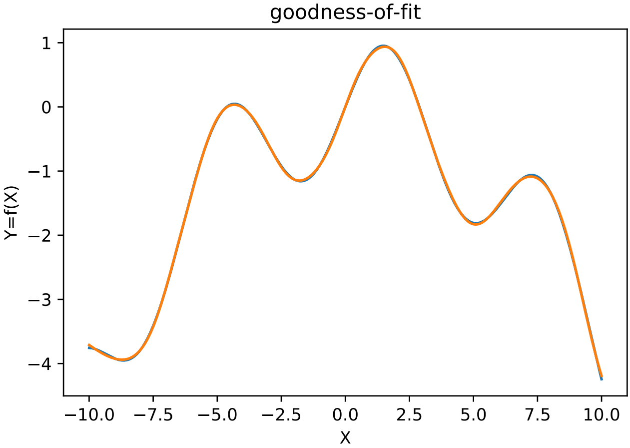


Machine Learning

Supervised Learning

Number of Nodes: 1–30–30–10–10–1

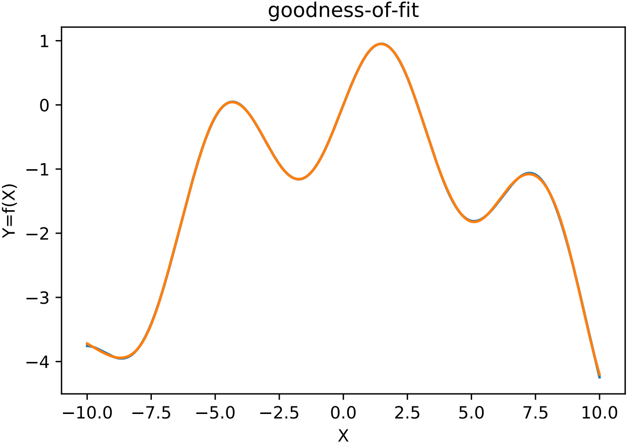
Number of Epochs: 10 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

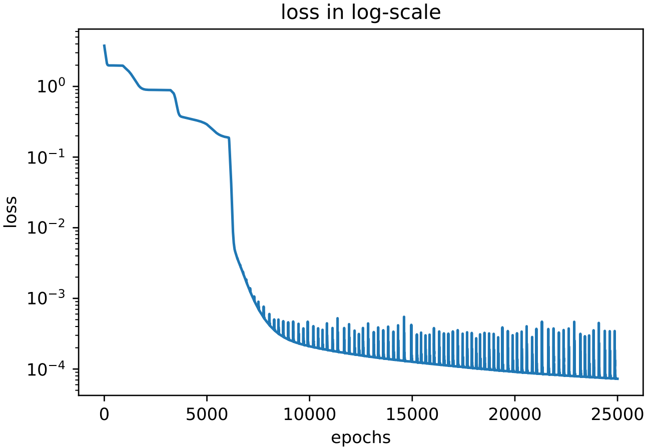
Number of Epochs: 25 000



Supervised Learning

Number of Nodes: 1–30–30–10–10–1

Number of Epochs: 25 000



Machine Learning

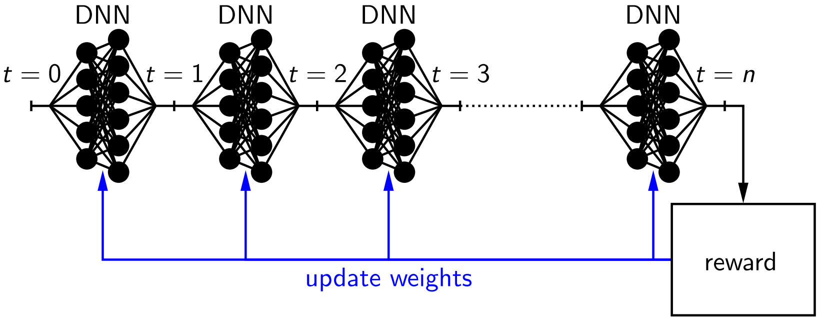
Observations

- The learning process evolves in small and random steps.
- The update of the weights results from the **backpropagation algorithm**. It can be seen as a very smart way of combining Monte-Carlo techniques and dynamic programming.
- Choosing suitable **hyperparameters** for the learning process might be tricky.
- Computing power is crucial.
- Neural networks can be evaluated efficiently by using pertinent software libraries, e.g., **TensorFlow**.
- Storing neural networks requires comparatively little storage space.

Machine Learning

Re-inforcement Learning

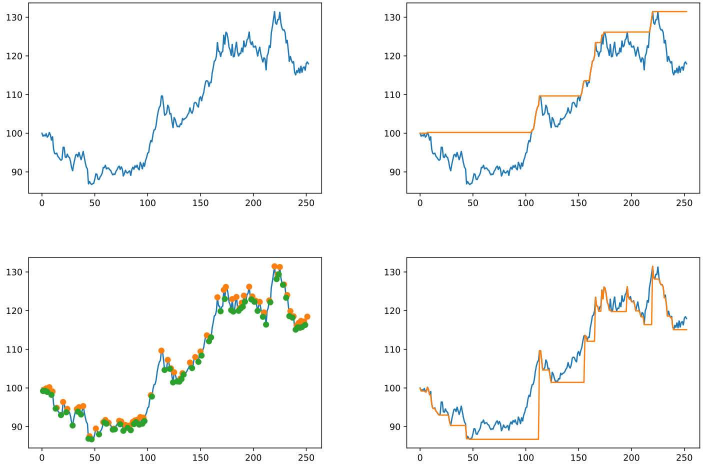
Training: Maximise a Reward Function



Validation: Check Performance of Decisions on New Scenarios

Re-inforcement Learning

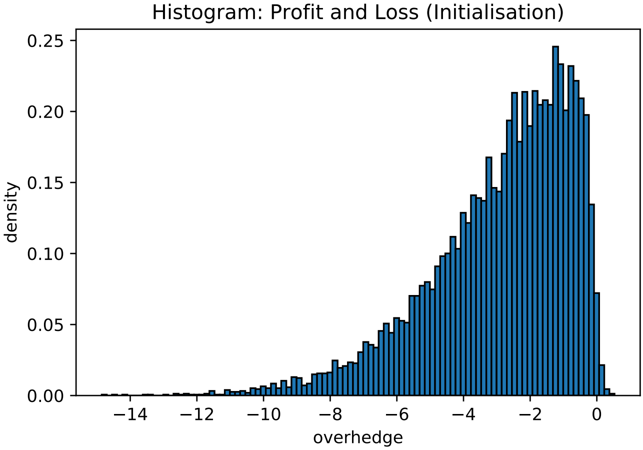
Scenarios, Features and States



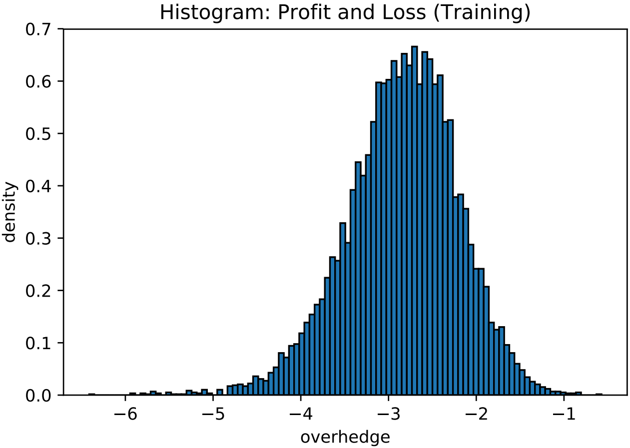
Experiment on Deep Hedging

- Exposure: We issue a call option with payoff $\max\{S_T - K, 0\}$, strike $K = 100$ and maturity $T = 30d$.
- Market Environment:
  - bank account
  - underlying
- Rules:
  - Investment strategies must be **self-financing**.
  - Re-allocations are possible once a day and may involve proportional **transaction cost**.
- Objective: We aim to minimise the quadratic discrepancy between the due payoff and the value of the hedge.
- Training: 10 000 scenarios

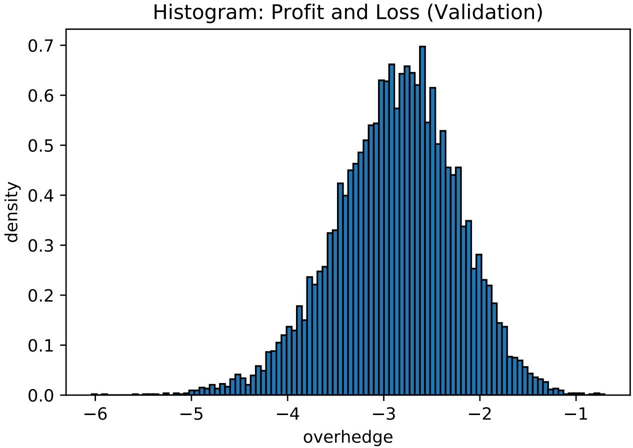
Deep Hedging (without Transaction Cost)



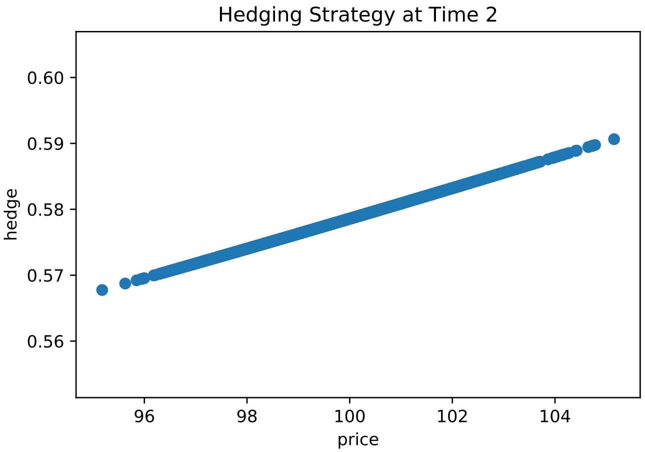
Deep Hedging (without Transaction Cost)



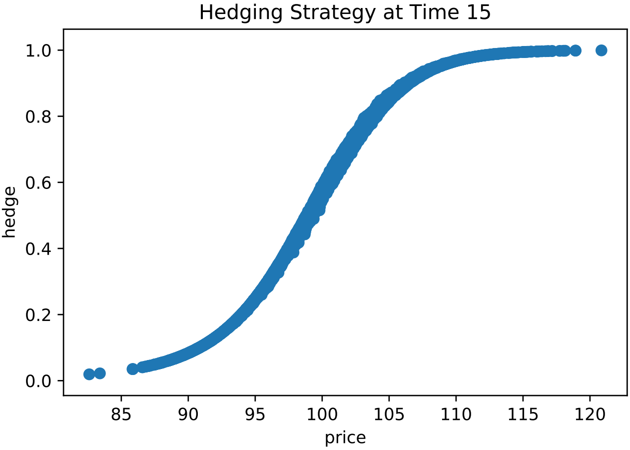
Deep Hedging (without Transaction Cost)



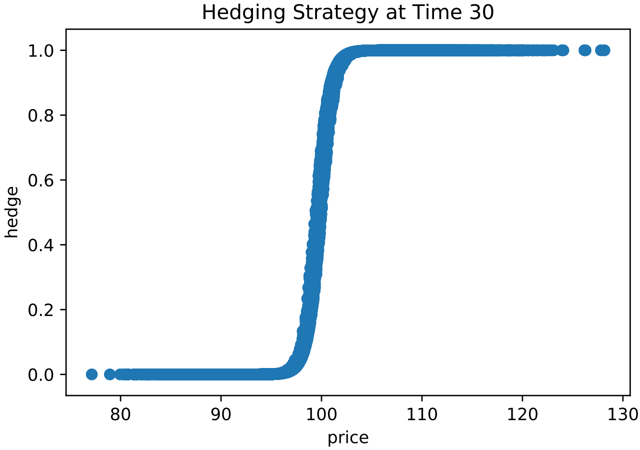
Deep Hedging (without Transaction Cost)



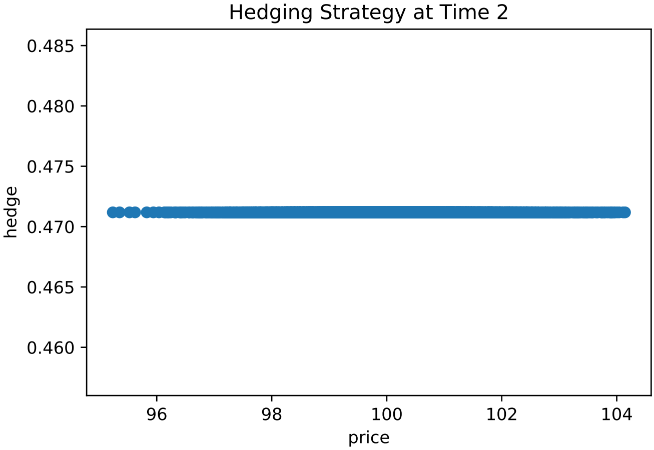
Deep Hedging (without Transaction Cost)



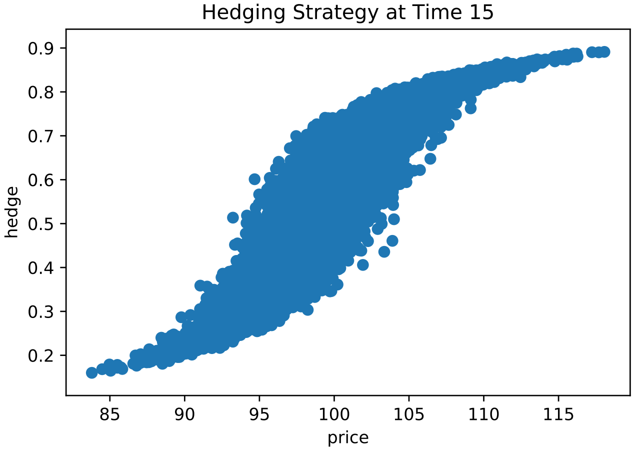
Deep Hedging (without Transaction Cost)



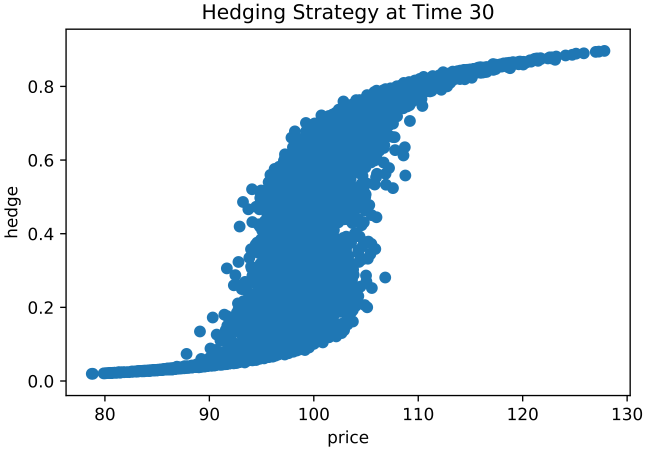
Deep Hedging (with Transaction Cost)



Deep Hedging (with Transaction Cost)



Deep Hedging (with Transaction Cost)



Machine Learning

Summary

Traditional Programming:

$$\text{data} \; + \; \text{program} \; \longrightarrow \; \text{output}$$

Supervised Learning:

$$\text{data} \; + \; \text{output} \; \longrightarrow \; \text{program}$$

Re-inforcement Learning:

$$\text{rules} \; + \; \text{scenarios} \; \longrightarrow \; \text{convincing strategy}$$

Hypothesis

Techniques inspired from re-inforcement learning pave the way for a new era in quantitative risk management from various viewpoints.

1. It is a **disruptive** technology; **high-dimensional** optimisation problems of this kind were not accessible until only recently.
2. It is a very **efficient** and **powerful** technology with
   - super fast requests-on-demand,
   - instantaneous validation (**model risk management**).
3. It is a very **flexible** technology. In a few lines of code, one easily accounts for
   - arbitrary path-dependent payoffs,
   - complex stochastic environments,
   - liquidity squeezes/transaction cost/price impacts,
   - regulatory constraints,
   - risk appetite,

     ⋮

Outline

1. Challenges
   - Asset Management
   - Pricing and (Over-)Hedging
2. Neural Networks
3. Machine Learning
   - Supervised Learning
   - Re-inforcement Learning
4. Applications

Applications

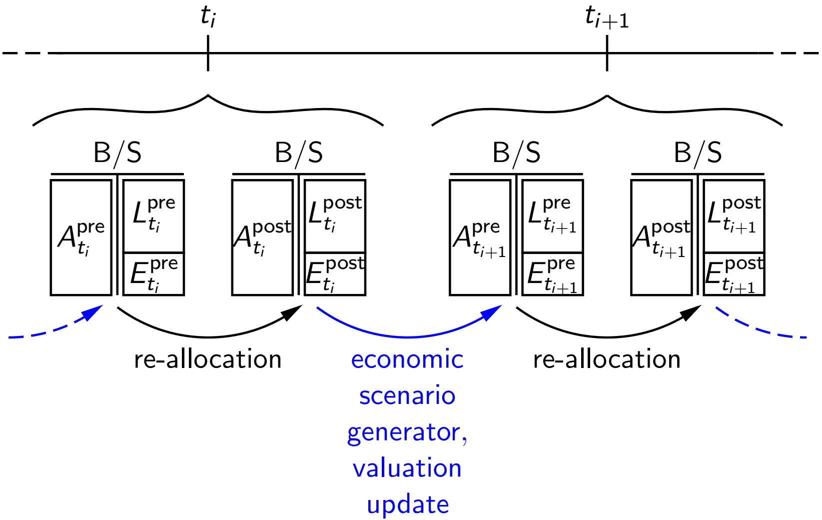
Balance Sheet of an Enterprise in the Financial Industry

Assets	**Liabilities**
Investmentportfolio	Debts
	Equity

Objective

Maximise the expected utility of the **return-on-equity** over different time instances while not exceeding a certain **draw-down** and while guaranteeing the **regulatory constraints** with a high probability.

Balance Sheet Roll-Forward



Applications

Model Ingredients for Re-inforcement Learning

- **economic scenario generator**
  - yield curves
  - credit migrations
  - stock prices
  - client behaviour

    ⋮
- parameterisation of the **states**
- **rule book**
  - constraints
  - eligible re-allocations
  - frictions
- **objective**

Applications

Deep Asset-Liability-Management

Simply put, one solves **high-dimensional** hedging problems with **constraints** in the presence of **frictions** by means of re-inforcement learning techniques.

- **retail bank**: replicating portfolio
- **insurance company**: strategic asset allocation that accounts for the necessary returns and institutional liquidity, optimised re-insurance programme
- **commodity trading**: optimal procurement in the presence of uncertainty, pricing impacts and storage cost
- **pump-storage hydropower plant**: optimised production plan, pricing and hedging in an illiquid environment

Applications

Further Research

- Reach a suitable level of **complexity**.
- Deal with **uncertainty** of model assumptions.
- Model choices and regulisations that promote **robust solutions**.
- Corroborate that sophisticated approach and additional complexity is **profitable**.

References

Becker, S., Cheridito, P., Jentzen, A.
Deep optimal stopping (2018).
arXiv:1804.05394.

Bühler, H., Gonon, L., Teichmann, J., Wood B.
Deep Hedging (2018).
arXiv:1802.03042.

Longstaff, F. A., Schwartz, E. S.
Valuing American Options by Simulation: A Simple Least-Squares Approach (2001).
The Review of Financial Studies.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., Hassabis, D.
Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm (2017).
arXiv:1712.01815v1.

Credits

thomas.krabichler@hslu.ch