Q-Fitted Iteration in a Heston Simulation World for Option Pricing

Systematic Trading Division

Vienna, 12.06.2020

Team Overview

Systematic Trading

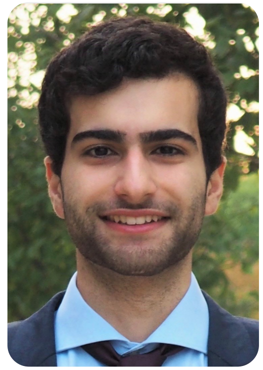
W U T I S

Michael
Dampf

Co-Head

- Methodology Concept
- Backtesting

- MSc. – 2nd Sem. (B&F)
- BSc. (WU)



Parham
Allboje

Co-Head

- RL Model Implementation

- MSc. – Incoming (ICL)
- BSc. (WU)

David
Hirnschall

Associate

- Heston Model Implementation

- PhD - Incoming (Stats)
- Msc. - (FinAcMath - TU)
- Bsc. - (FinAcMath - TU)

Dmitriy
Klimov

Analyst

- δ-Hedge & Backtest Implementation

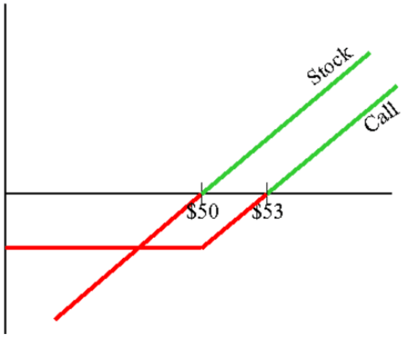
- MSc. – 2nd Sem. (B&F)
- BSc. (B&F)

Option Valuation

What it is and why we care

What makes options unique?

- Options have a non-linear pay-off
- The holder of an option has unlimited profit potential, but limited downside potential – for this favorable position, he must pay a premium to the seller
- The question is: how big should that premium be?



How they are priced – The Black & Scholes Formula

- Introduced by a paper in 1973 by Fischer Black, Myron Scholes and Robert Merton
- They received a noble price in 1997
- Still used nowadays

The Black & Scholes Formula

Basic Formula:

$$C(S,t) = S\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2)$$

With:

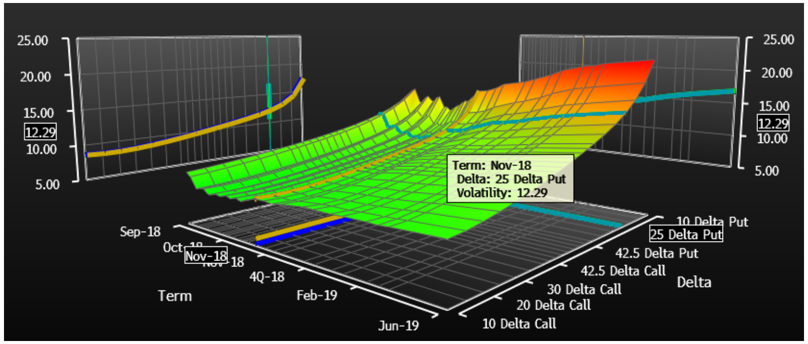
$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) \mathrm{d}z$$

$$d_1 = \frac{\ln(S/K) + (r + \sigma^2/2)(T-t)}{\sigma\sqrt{T-t}}$$

$$d_2 = \frac{\ln(S/K) + (r - \sigma^2/2)(T-t)}{\sigma\sqrt{T-t}} = d_1 - \sigma\sqrt{T-t}$$

The flaws of Black & Scholes

- Black & Scholes assumed that an options implied volatility is constant
- Volatility however is "skewed" – it differs across deltas and expirations. This results in the volatility surface, which reflects everything that B&S fail to capture in their model
- Therefore: room for improvement



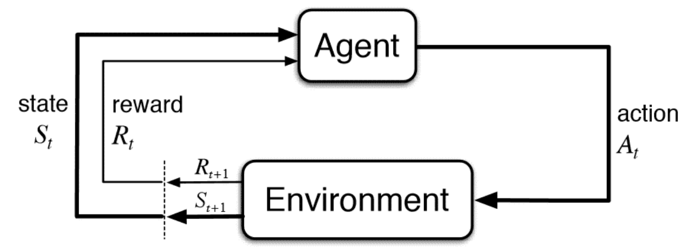
Reinforcement Learning

Optimizing Optimal Policies in order to maximize rewards in a Markov Decision Process

What is Reinforcement Learning?

- In Reinforcement Learning (RL) the goal is to maximize rewards
- An agent performs an action to transition from on state to a next one and is given a **reward** in that next state.
- Q-Learning is a RL algorithm where the goal is to learn the **optimal policy**. A policy is a set of rules tot tell the agent what action to take in given state
- Here the agent choses an action, observes a reward and enters a new state, updating Q, the “quality” of the action take at each time t

Agent-Environment Interaction in RL (Markov Decision Process)



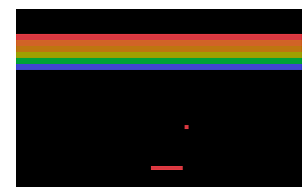
RL vs Other AI and Machine Learning algorithms

	AI Planning	SL	UL	RL	IL
Optimization	x			x	x
Learns from experience		x	x	x	x
Generalization	x	x	x	x	x
Delayed Consequences	x			x	x
Exploration				x	

- SL = Supervised Learning; UL = Unsupervised Learning; RL = Reinforcement Learning; IL = Imitation Learning
- Reinforcement Learning is provided with censored labels (SL -> correct labels; UL ->no labels; IL reduces RL to SL)

Successful Applications of Reinforcement Learning

Self Landing Rockets



Atari Games

The Q-Learning Black Scholes pricing model

Reducing the problem of option pricing to rebalancing of a dynamic replicating portfolio

QLBS Reward Function created from replicated portfolio

- The QLBS model starts with a discrete-time version of BS. To hedge the option, the seller replicates portfolio ∏ made of stock S and deposit B:

$$\Pi_t = a_t S_t + B_t$$

- The optimal value function is expressed through the optimal option hedging and pricing formulated as a Stochastic Optimal Control (SOC) problem:

$$Q_t^{\star}(X_t, a_t^{\star}) = \gamma \mathbb{E}_t \left[Q_{t+1}^{\star}(X_{t+1}, a_{t+1}^{\star}) - \lambda\gamma \hat{\Pi}_{t+1}^2 + \lambda\gamma \left(a_t^{\star}\left(X_t\right)\right)^2 \left(\Delta \hat{S}_t\right)^2 \right]$$

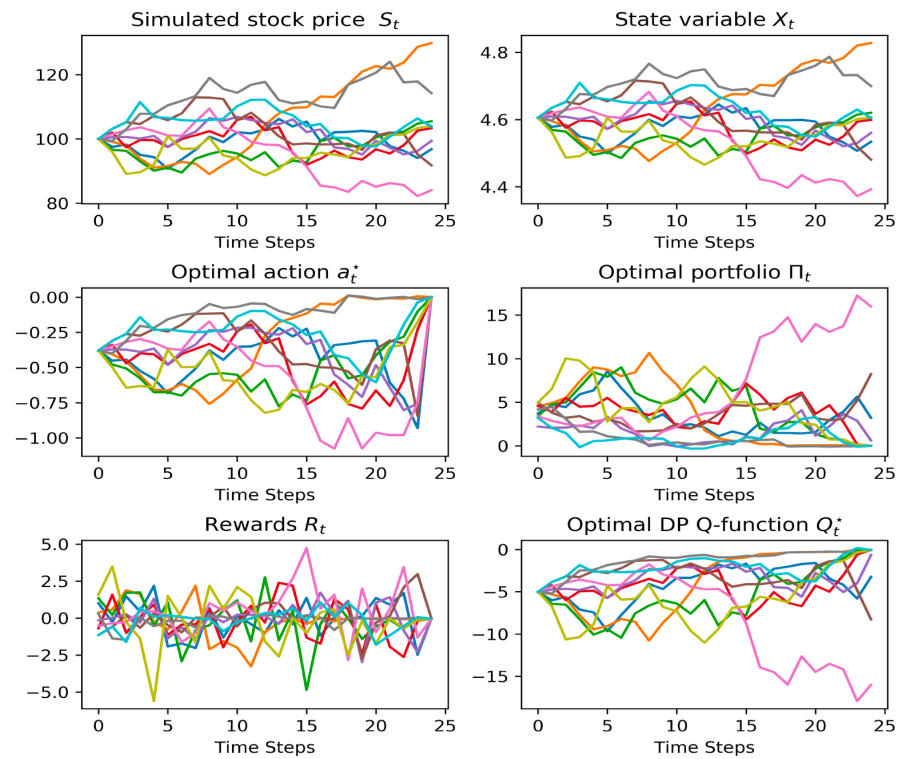
Solving the recursive problem statement through simulation

- In practice, the below stated recursion problem is solved in a Monte Carlo Setting, where we simulate N paths for the state variable Xt.
- For our project we used Geometric Brownian Motion and the Heston Model
- From the simulation we compute a terminal pay-off, which is the dollar amount an investor receives from the option strategy

$$H_T(S_T) = \max(K - S_T, 0)$$

- Using this terminal value the compute a portfolio and the optimal hedge and backwards update our parameters to converge to the optimal Q function which results from the optimal action that yields the optimal Reward

Running 100k simulations per day in order to get the optimal Q-function



- The QLBS option price is given by :

$$C_t^{(QLBS)}(S_t, ask) = -Q_t(S_t, a_t^{\star})$$

Heston Model

What it is and why do we use it

Key Facts

- Introduced by Steven Heston in 1993
- Volatility is modeled as time-dependent, stochastic process
- Approximates whole volatility surface
- Computationally intensive

Closed Form Solution

Basic Formula:

$$C_0 = S_0 . \Pi_1 - e^{-rT} K . \Pi_2$$

With:

$$\Pi_1 = \frac{1}{2} + \frac{1}{\pi} \int_0^{\infty} Re \left[\frac{e^{-i.w.ln(K)} . \Psi_{lnS_T}(w-i)}{i.w.\Psi_{lnS_T}(-i)} \right] dw$$

$$\Pi_2 = \frac{1}{2} + \frac{1}{\pi} \int_0^{\infty} Re \left[\frac{e^{-i.w.ln(K)} . \Psi_{lnS_T}(w)}{i.w} \right] dw$$

Black Scholes vs. Heston

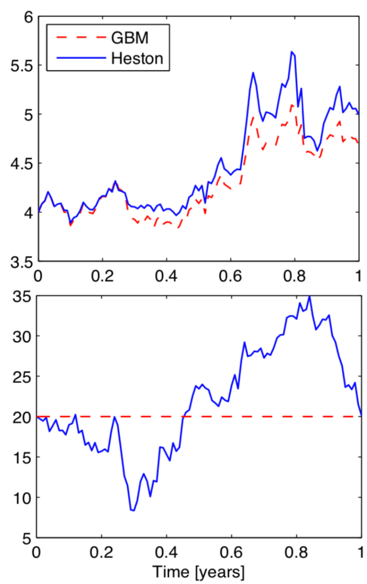
Black Scholes Model Definition:

$$dS_t = \mu S_t + \sigma S_t dW_t$$
$$\sigma = constant$$

Heston Model Definition:

$$dS_t = \mu S_t + \sqrt{\sigma_t} S_t dW_t^1$$
$$d\sigma_t = \kappa(\theta - \sigma_t)dt + \nu\sqrt{\sigma_t} dW_t^2$$

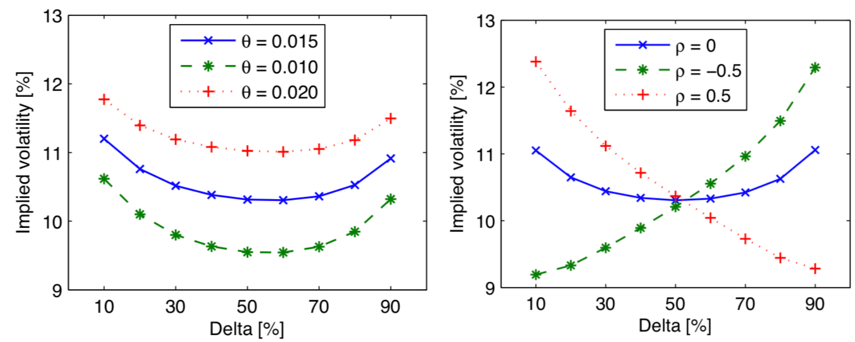
where W^1 and W^2 are Brownians Motions with correlation ρ



Model Calibration

Find best values for: $\kappa, \theta, \nu, \rho$ and σ_0:

$$\sum_{i=1}^{N} (C(T,\Delta)^{market} - C(T,\Delta)^{Heston})^2 \rightarrow min \qquad \forall T, \Delta$$



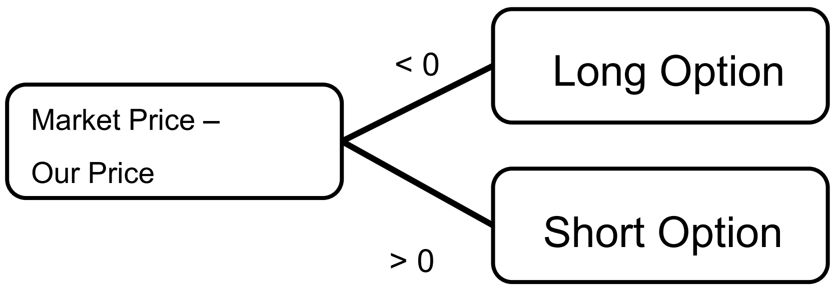
https://demonstrations.wolfram.com/VolatilitySurfaceInTheHestonModel/

Backtest

Methodology

Basis for Algorithm

- Every day at close we compare the market price with the q-learning derived price
- Based on this comparison, we will decide if the option is over or underpriced and position ourselves accordingly



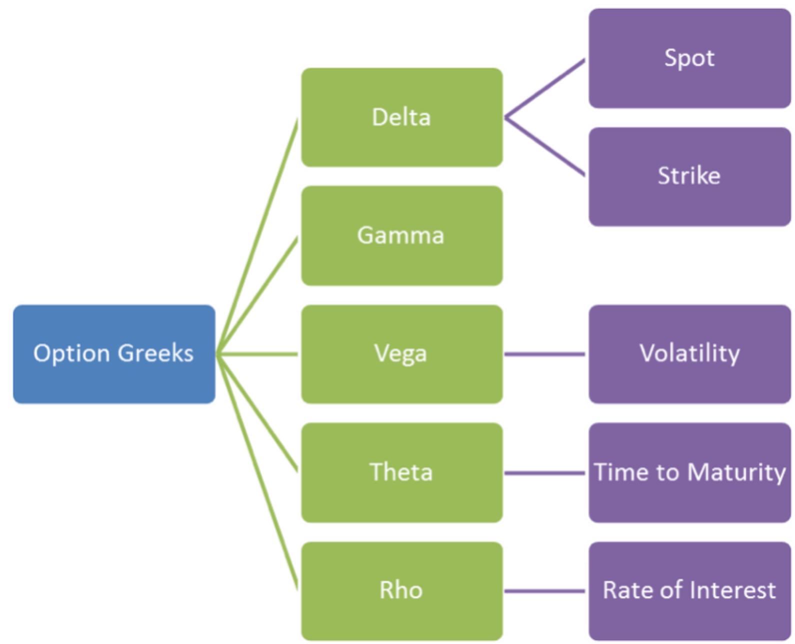
Delta hedging

- The aim of our model is to derive a better price for the option – not to predict the market
- A naked short on the wrong day however might drag down the return of our portfolio purely due to bad luck
- To avoid this scenario, we "delta hedge" our portfolio – i.e. we eliminate the impact of the underlying on our PnL

t	Underlying	Delta	p&l
1	10.00	0.50	
2	5.00	0.25	2.50
3	10.00	0.50	-1.25

Risk factors of an option

- There are several factors which have an impact on the option price – they are called the "Greeks"
- Delta refers to the sensitivity of an option to its underlying – i.e. if the underlying goes up by 1$ tomorrow, how does the price of the option change?



Results

And further applications

Main Stats

- Total return: 1.55 %
- Max drawdown: -0.47 %
- Sharpe ratio : 1.05

Important findings

- Q-learning option pricing give us better prices for put options
- The model performs well in high volatility

Further applications

- Optimization of the simulation approach
- Approximation the Q-tale using deep learning
- Adjusting of the model from the q fitted iteration to inverse reinforcement learning

Performance overview