Financial modeling with applications of machine learning and explainable AI

Bartłomiej Małkus

Jagiellonian University

21.10.2021

Outline

- Financial modeling in general
- Financial instruments
- Financial models
- Model calibration
- Data sources
- Generating synthetic financial data

Financial modeling in general

Financial modeling may seem like a very broad term, and it is.

There's no one general definition for it - everybody understands it a bit differently and as having different scope.

Common definitions of financial modeling

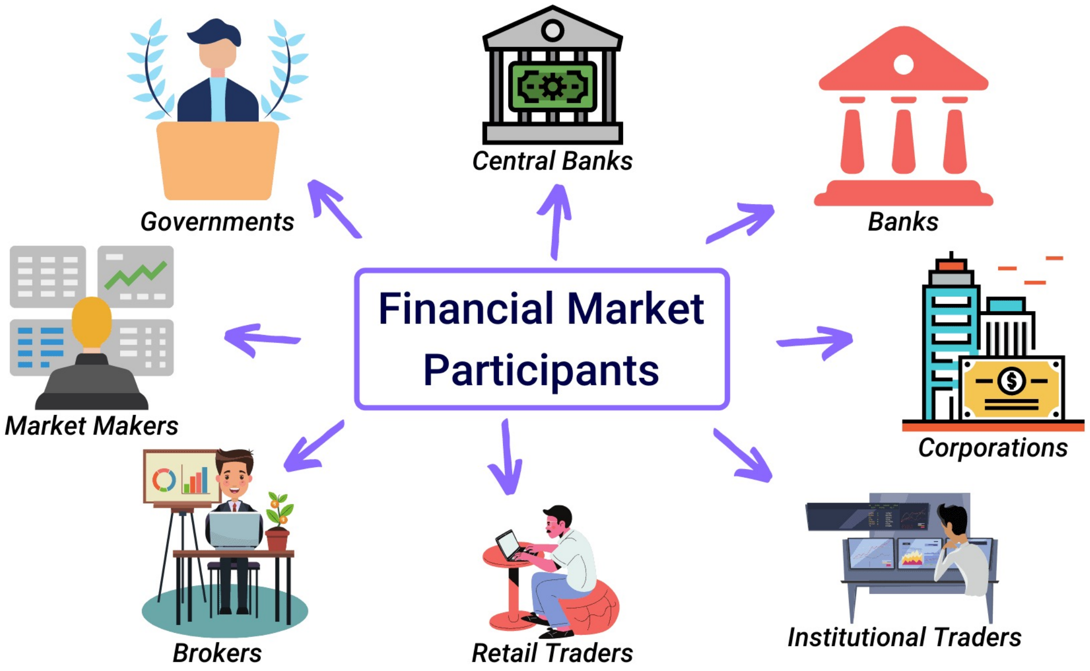
Wikipedia says it's "the task of building an abstract representation (a model) of a real world financial situation"

Investopedia says it's "the process of creating a summary of a company's expenses and earnings in the form of a spreadsheet that can be used to calculate the impact of a future event or decision"

Moneyterms defines financial model as "anything that is used to calculate, forecast or estimate financial numbers"

Where is financial modeling utilized?

- In financial entities, like:
  - banks
  - insurance companies
  - investment funds
  - rating agencies
- In the Government
- In non-financial entities, like corporations and regular companies



What is financial modeling utilized for?

- Banks – all kinds of risks assessments, like credit risk, liquidity risk, operational risk, market risk etc., credit scoring, calculation of reserves and norms, valuation of assets and liabilities
- Insurers – calculation of insurance premiums, financial reserves, valuation of subjects of insurance, etc.
- Investment funds – valuation of all types of financial instruments, assets, risk management, etc.
- Rating agencies – basically living off models, assigning trustworthiness ratings to entities, financial instruments, countries, etc.
- Regular companies – budget management and forecasting, valuation, capital allocation, etc.

Financial instruments

Underlying instrument is a variable, e.g.:

- stock price
- stock index value
- bond yield
- interest rate

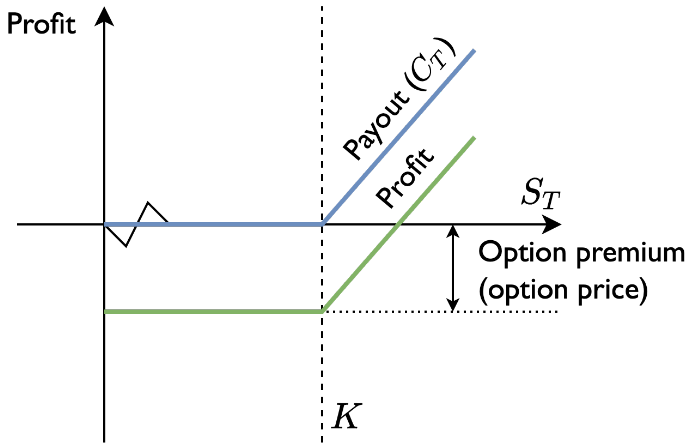
Derivative is an instrument, whose value depends on the underlying instrument:

- option
- future/forward contract
- swap

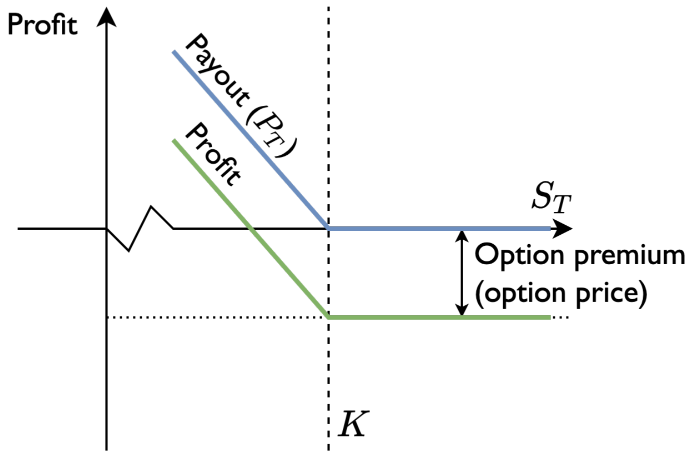
For the purposes of valuation of derivative instruments, the behavior of the underlying instruments is modeled

Options

- The options give the right to buy / sell the underlying instrument at a fixed price on a fixed date
- Options types (european / american / exotic)
- Why is option valuation important?
  - options are widely used in hedging a portfolio position
  - options are also used for speculation



$$C_T = \max(S_T - K; 0)$$



$$P_T = \max(K - S_T; 0)$$

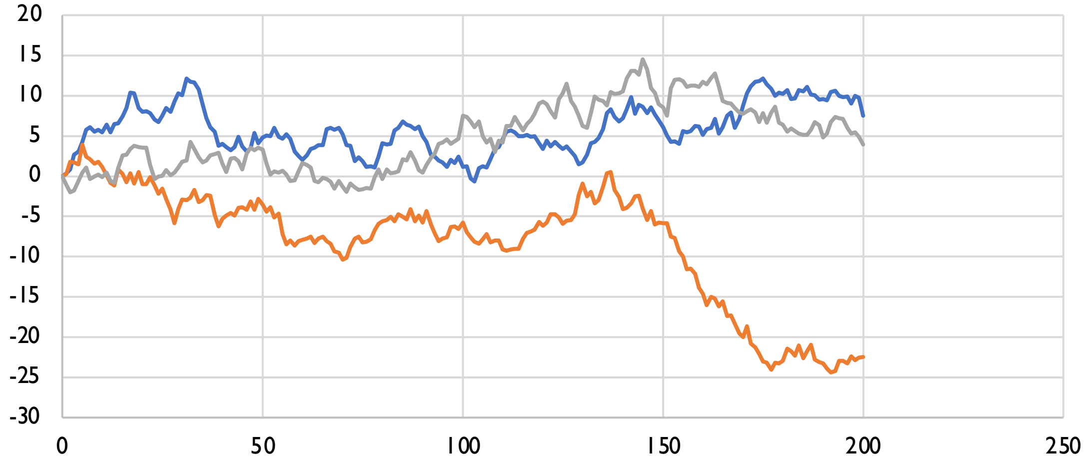
Black-Scholes model

Wiener process

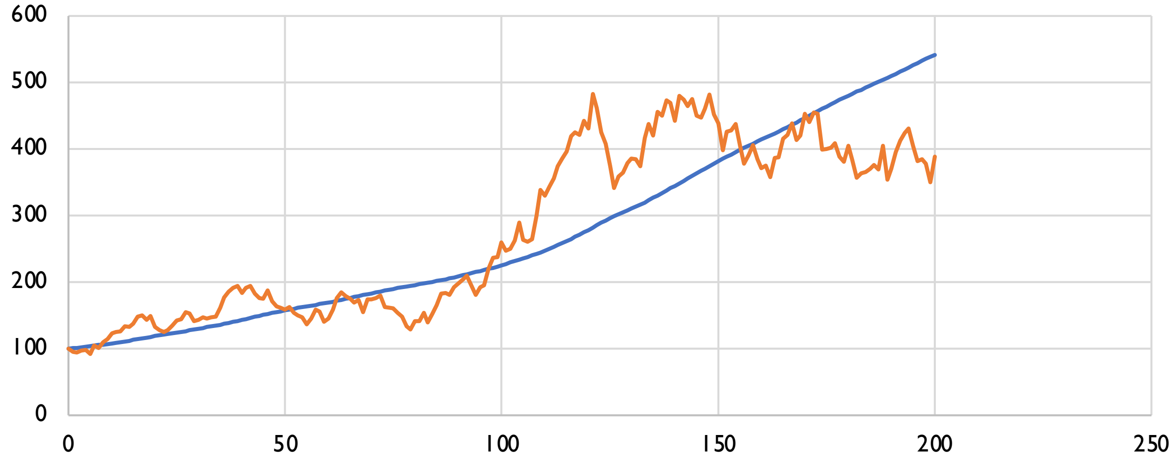
$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

long-term trend

random fluctuations



Example implementations of the Wiener process



Example implementation of the geometric Brownian motion

Black-Scholes formula (for call option price)

option won't be exercised — option will be exercised

$$C_0 = S_0 N(d_1) + K e^{-rT} N(d_2)$$

$$d_1 = \frac{\ln \frac{S_0}{K} + \left(r + \frac{\sigma^2}{2}\right) T}{\sigma \sqrt{T}} \qquad d_2 = d_1 - \sigma\sqrt{T}$$

where:

S_0 - (known) current share price

K - (known) option strike price

T - (known) option expiry time

r - (known) risk-free rate

σ - (unknown) standard deviation of the logarithmic returns (volatility)

Black-Scholes model extensions

The Black-Scholes model has some assumptions that are not necessarily met (such as the fact that the volatility of $\boldsymbol{\sigma}$ is constant over time) therefore there are many extensions to it. One of them is the Heston model.

$$dS_t = rS_t dt + \sqrt{\nu_t} S_t dW_t^s, \qquad \mathrm{S}_{\mathrm{t}_0} = \mathrm{S}_0$$

$$d\nu_t = \kappa(\bar{\nu} - \nu_t)dt + \gamma\sqrt{\nu_t}dW_t^\nu, \qquad \nu_{t_0} = \nu_0$$

$$dW_t^s dW_t^\nu = \rho dt$$

New parameters appear in the model.

Black-Scholes model extensions

The Bates model extends the Heston model with random jumps in the prices of the underlying instrument. Next parameters appear in the model.

$$dS_t = rS_t dt + \sqrt{\nu_t} S_t dW_t^s + \left(e^{\alpha + \delta\epsilon} - 1\right) S_t dq, \qquad S_{t_0} = S_0$$

$$d\nu_t = \kappa(\bar{\nu} - \nu_t)dt + \gamma\sqrt{\nu_t}\, dW_t^\nu, \qquad \nu_{t_0} = \nu_0$$

$$dW_t^s dW_t^\nu = \rho dt$$

Why is model calibration important?

- In financial institutions, models are one of the key elements in investment decision-making
- Models are recalibrated multiple times during the day
- Calibration is resource intensive (due to the time needed to evaluate models and the use of mainly non-gradient methods)

Calibration problem

We define quotation (stock price or implied volatility) resulting from the model:

$$Q(\tau; \theta)$$

where τ denotes the features of the given instrument, and $\theta \in \mathbb{R}^n$ denotes the model parameters (n is the number of these parameters).

And market quotation:

$$Q^{mkt}(\tau)$$

We want to determine the parameters θ minimizing the cost function:

$$\arg\min_{\theta \in \mathbb{R}^n} \sum_{i=1}^{N} \omega_i \left(Q(\tau_i; \theta) - Q^{mkt}(\tau_i) \right)^2$$

Stock price and implied volatility

Knowing the call option price (market or model based) V, implied volatility σ^* can be calculated by solving following equation:

$$BS(\sigma^*; S, K, \tau, r) = V$$

In explicit form:

$$\sigma^*(m, \tau) = BS^{-1}(V; m, \tau, r)$$

where $m = \frac{S}{K}$ and $\tau = T - t$

There is no analytical solution to the above equation. Numerical methods are used to solve it.

Stock price and implied volatility

Instead of using numerical methods, a neural network can be used, mapping the parameters $\{V, m, \tau, r\}$ to volatility σ^*.

Benefits of such approach:

- network can be trained on synthethic data
- network is faster than numerical methods

A problem appears (which seems to be already solved) – when S jest strongly different than K (i.e., when $m < 0.5$ or $m > 2$) model ceases to be sensitive to σ^* changes, large gradients appear in the inverse function, which negatively affects the network performance.

Stock price and implied volatility

IV-ANN	Parameters	Range	Unit
Input	Stock price (S_0/K)	[0.5, 1.4]	-
	Time to maturity (τ)	[0.05, 1.0]	year
	Risk-free rate (r)	[0.0, 0.1]	-
	Scaled time value ($\log(\tilde{V}/K)$)	[−16.12, −0.94]	-
Output	Volatility (σ)	(0.05, 1.0)	-

(for 20.000 european options)

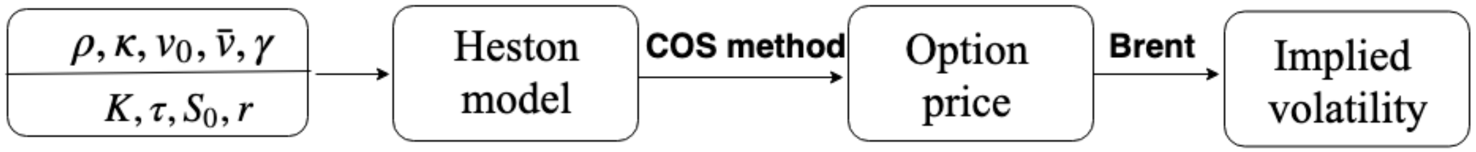
Method	GPU (seconds)	CPU (seconds)	Robustness
Newton-Raphson	19.68	23.06	No
Brent	52.08	60.67	Yes
Secant	88.73	103.76	No
Bi-section	337.94	390.91	Yes
IV-ANN	0.20	1.90	Yes

IV-ANN	MSE	MAE	MAPE	R^2
Input: $m, \tau, r, V/K$ Output: σ^*	6.36×10^{-4}	1.24×10^{-2}	7.67×10^{-2}	0.97510
Input: $m, \tau, r, log(\tilde{V}/K)$ Output: σ^*	1.55×10^{-8}	9.73×10^{-5}	2.11×10^{-3}	0.9999998

Source: Liu, 2019

Supporting model calibration with neural networks

Let's take the Heston model into consideration. Classic valuation using this model looks as follows:



The calibration of this model is performed with the use of non-gradient heuristics such as Differential Evolution or Particle Swarm Optimization.

Two bottlenecks arise here - the time needed to evaluate the model in the calibration process and the pace of the calibration itself.

Supporting model calibration with neural networks

To eliminate the first bottleneck, neural network can be used again, mapping the model parameters $\{\rho, \kappa, v_0, \bar{v}, \gamma; K, \tau, S_0, r\}$ to the price of an option V. The valuation process will be as follows:

where IV-ANN is the neural network described earlier.

The advantages of using such a network are similar to the previous case: the possibility of training on synthetic data and speed up of the operation.

Supporting model calibration with neural networks

ANN	Parameters	Range	Method
Input	Moneyness, $m = S_0/K$	(0.6, 1.4)	LHS
	Time to maturity, τ	(0.1, 1.4)(year)	LHS
	Risk free rate, r	(0.0%, 10%)	LHS
	Correlation, ρ	$(-0.95, 0.0)$	LHS
	Reversion speed, κ	(0.0, 2.0)	LHS
	Long average variance, $\bar{v}$	(0.0, 0.5)	LHS
	Volatility of volatility, γ	(0.0, 0.5)	LHS
	Initial variance, v_0	(0.05, 0.5)	LHS
Output	European call price, V	(0, 0.67)	COS

Heston-ANN	MSE	MAE	MAPE	R^2
Training	1.34×10^{-8}	8.92×10^{-5}	5.66×10^{-4}	0.9999994
Testing	1.65×10^{-8}	9.51×10^{-5}	6.27×10^{-4}	0.9999993

Heston-ANN & IV-ANN	RMSE	MAE	MAPE	R^2
Case 1: $\tau \in [0.3, 1.1], m \in [0.7, 1.3]$	7.12×10^{-4}	4.19×10^{-4}	1.46×10^{-3}	0.999966
Case 2: $\tau \in [0.4, 1.0], m \in [0.75, 1.25]$	5.53×10^{-4}	3.89×10^{-4}	1.14×10^{-3}	0.999980

Source: Liu, 2019

Supporting model calibration with neural networks

After training the network, calibration of the model comes down to determining the following network input parameters θ:

$$\{\theta; K, \tau, S_0, r\} \mapsto V(\theta; K, \tau, S_0, r)$$

To minimize the cost function:

$$\arg\min_{\theta \in \mathbb{R}^n} \sum_{i=1}^{N} \omega_i \left(V(\theta; K, \tau, S_0, r) - V^{mkt}(K, \tau, S_0, r)\right)^2$$

Supporting model calibration with neural networks

Problem: a trained network approximates the same function, so gradient methods still won't be able to optimize it – we have to use the same calibration methods as before.

This problem remains open.

We already know gradients in the network, so maybe they can support current methods?

In the literature, there are solutions that replace heuristics with neural networks, but these are rather theoretical considerations.

Models assumptions

Models which are approximated using neural networks have certain assumptions that must be satisfied by that neural network (such as assumption that there is no arbitrage in the market), which can be written in the form of requirements imposed on derivatives:

$$\frac{\partial V}{\partial T} > 0, \qquad \frac{\partial V}{\partial K} < 0, \qquad \frac{\partial^2 V}{\partial K^2} > 0$$

Data sources

Macroeconomic data is available from government sites for particular economies.

OTC data:

- *finance.yahoo.com* – free, mostly US equites, bonds, FX, commodities
- *stooq.pl* – free, mainly polish quotes and bonds, but also main foreign, indexes, FX
- *data.nasdaq.com* (formerly *quandl.com*) – paid/free, various equities, FX, macro data, packaged into groups, most of packages are affordable
- *historicaloptiondata.com* – US equity options, paid, but affordable, data is available with high granularity

Why do we need synthetic financial data?

There are couple of reasons:

- privacy (of data subjects)
- data use restrictions
- small amount of historical data for particular events (crashes on the market, recessions, recoveries, etc.)
- too little data to train more advanced models

Generating synthetic financial data

Let's split financial data into two categories:

- retail banking data (e.g. customer data, including age, profession, income, marital status, gender)
- market microstructure data (time series, e.g., stock price or implied volatility over time)

Generating synthetic financial data - methods

- Modeling real data with particular models (AR, GARCH, Black-Scholes, Heston for options, etc.)
- Neural networks – QuantGANs, CGANs
- “Inverting” decision tree classifiers, SVM
- Agent-based models (more advanced, mostly for real-time data simulation)

Thank you for your attention

Questions?

Sources

Samonas, M. (2015). *Financial forecasting, analysis, and modelling: A framework for long-term forecasting.* John Wiley & Sons.

Hernandez, A. (2016). Model calibration with neural networks. *SSRN Electronic Journal*.

Liu, S., Borovykh, A., Grzelak, L. A., & Oosterlee, C. W. (2019). A neural network-based framework for financial model calibration. *Journal of Mathematics in Industry*, *9*(1), 9.

Büchel, P., Kratochwil, M., Nagl, M., & Roesch, D. (2020). Deep calibration of financial models: Turning theory into practice. *SSRN Electronic Journal*.

Itkin, A. (2019). Deep learning calibration of option pricing models: Some pitfalls and solutions. *arXiv:1906.03507*

Wiese, M., Knobloch, R., Korn, R., & Kretschmer, P. (2020). Quant GANs: Deep generation of financial time series. *Quantitative Finance*, *20*(9), 1419–1440.

Assefa, S. A., Dervovic, D., Mahfouz, M., Tillman, R. E., Reddy, P., & Veloso, M. (2020). Generating synthetic data in finance: Opportunities, challenges and pitfalls. *Proceedings of the First ACM International Conference on AI in Finance*, 1–8.