Quantitative Finance

Edoardo Vittori

July 4th, 2025

Today's Presentation

Financial Markets

- Market Making
- Execution
- Asset Managers and Hedge Funds
- Prop Traders
- Quants

Financial Engineering

- Stochastic processes
- Option pricing
- MC simulations
- Execution algorithms

Quant Toolkit

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Deep Learning
- Generative AI

Quant Research

- Classical systematic strategies
- Learning an FX trading strategy with RL
- Intraday trading with ML
- Cross sectional momentum with LLMs

Today’s Presentation

Financial Markets

- Market Making
- Execution
- Asset Managers and Hedge Funds
- Prop Traders
- Quants

Financial Engineering

- Stochastic processes
- Option pricing
- MC simulations
- Execution algorithms

Quant Toolkit

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Deep Learning
- Generative AI

Quant Research

- Classical systematic strategies
- Learning an FX trading strategy with RL
- Intraday trading with ML
- Cross sectional momentum with LLMs

Financial Markets

CIB | Capital Markets

	Market Making	Asset Management Hedge Funds	Execution	Prop Trading
Scope	• **Offering liquidity** to the markets by continuously pricing assets • Risk transfer and inventory management	• **Managing investments** on behalf of clients • Portfolio construction • Investment research	• **Executing trades** requested from clients or internal desks • Focus on minimizing market impact and achieving best execution	• Trading with the firm's capital • VaR limits • **Intraday investments** • Use of derivative instruments
Quant focus	• Auto pricing • Auto hedging • Fast pricing of complex instruments	• Portfolio optimization • Risk analytics • Alpha generation	• Short term market prediction • Low latency	• Returns prediction • Earnings prediction • Trading signals • Analytics

Execution desks

Given a trade, execution desks should minimize transaction costs and market impact

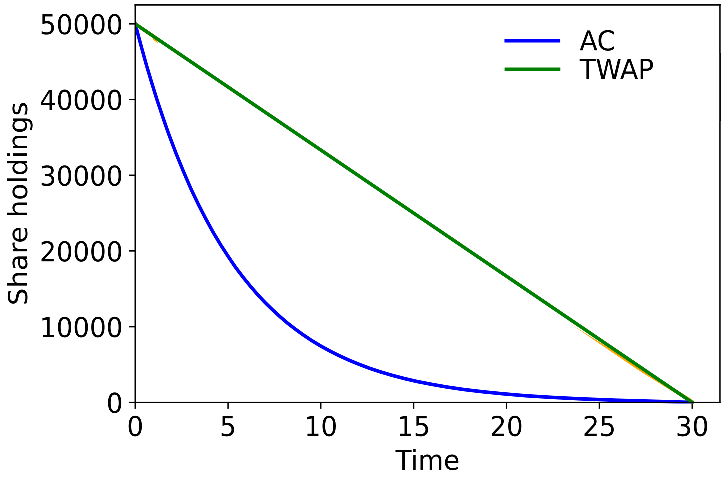
Definition of optimal execution

Given a trade to execute in a specified amount of time, minimize market impact and transaction costs

Order types

- **Market order** is an order to execute immediately at the best price possible
- **Limit order** is an order that specifies both the price and volume of a trade
- A limit order sits in the order book until it is either executed against a matching market order or canceled

Illustration of execution trajectories



Market making

Offering liquidity to the markets

Regulated market example

Last	Last Vol	Total Vol	Close	Daily Low	Daily High
4045.00	2	367267	4097.50	4033.50	4101.50

Implied

Bid		Offer	
Volume	Price	Price	Volume
136	4044.50	4045.00	62
327	4044.00	4045.50	293
348	4043.50	4046.00	427
620	4043.00	4046.50	426
358	4042.50	4047.00	463
330	4042.00	4047.50	348
325	4041.50	4048.00	327
318	4041.00	4048.50	294
305	4040.50	4049.00	281
512	4040.00	4049.50	288

Dealer market example - OTC

MARKIT ITRX EUR SNR FIN 06/26 | 92) Order Book | 91) RFS | 97) Settings
11:56:39 | 95) Buy | 96) Sell | BTFE | Filter By | All

PCS	Firm Name	CCP	Bid Spd / Ask Spd	BSz(MM) x ASz(MM)
CSDE	CREDIT SUISSE INTL	ICEE	54.6900 / 55.0100	50 x 50
CCGC	Citi CCGC	ICEE	54.7650 / 55.0350	50 x 50
GSMX	GS MINI	ICEE	54.7350 / 55.0350	15 x 15
JCTT	JP MORGAN	ICEE	54.7600 / 55.0400	100 x 100
BXCZ	Barclays Minis	ICEE	54.8400 / 55.0400	75 x 75
MSTI	MORGAN STANLEY MINI	ICEE	54.8000 / 55.0400	50 x 50
ABNP	BNP Paribas	ICEE	54.8000 / 55.0500	51 x 51
SGMI	SocGen Mini	ICEE	54.7380 / 55.0880	50 x 50
CSEO	CS iTraxx Mini	ICEE	54.610 / 55.090	100 x 100
BARX	Barclays	ICEE	54.7650 / 55.1150	250 x 250
CGCX	Citi CGCX	ICEE	54.6800 / 55.1200	100 x 100
EBNP	BNP Paribas	ICEE	54.7250 / 55.1250	101 x 101
SCDS	SocGen	ICEE	54.6890 / 55.1380	125 x 125
DBVD	DB Index-(DBDV)	ICEE	54.8500 / 55.1500	100 x 100
GSET	GOLDMAN SACHS	ICEE	54.6100 / 55.2100	75 x 75
CCGB	Citi CCGB	ICEE	54.5600 / 55.2400	200 x 200
JPOS	JP Morgan	ICEE	54.5600 / 55.2400	200 x 200
MSTT	MORGAN STANLEY MAXI	ICEE	54.5500 / 55.2900	100 x 100
CSXE	Credit Suisse EU	ICEE	54.406 / 55.294	200 x 200

RFQ Example

Client buys protection 200mln
Price: ____

Send

Asset Management and Hedge Funds

Portfolio managers

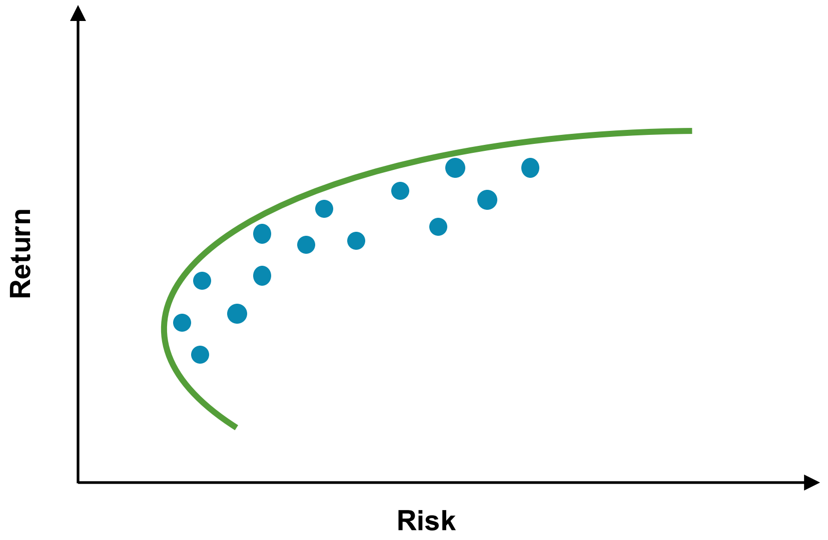
Objective

Manage investments to meet specific risk-return objectives for clients or proprietary funds. Usually comparing to a **benchmark**

Core Activities

- **Portfolio Construction**: apply techniques like Modern Portfolio, Risk Parity, and to construct diversified portfolios.
- **Strategy Allocation**: allocate capital across asset classes and strategies (e.g. long/short equity, macro, credit, multi-strategy) to:
    - Achieve a balanced exposure
    - Maximize risk-adjusted returns (Sharpe ratio, IR)
    - Maintain diversification and liquidity
- **Risk Management**: monitor volatility, VaR, drawdowns, and correlation

Illustration of efficient frontier



Prop Trading

Traders

Objective
Generate alpha by exploiting short to medium term market opportunities using the institution's own capital, within strict risk constraints (e.g., VaR limits)

Core Activities
- Alpha Generation: Develop and execute strategies across asset classes (e.g., equities, FX, rates, commodities, crypto) based on:
    - Technical indicators
    - Quantitative signals (statistical arbitrage, momentum)
    - Macro views or event-driven insights
- **Risk Budgeting:** operate within daily and aggregate Value-at-Risk (VaR) constraints
- **Monitoring:** monitor intraday P&L, position limits, and tail risk

Trading Working Desk Example

Quants and Financial engineering

The enabling component

Quants

Quantitative analysts/researchers/developers use advanced mathematics, statistics, and programming to **analyze financial data** and **develop models** that support trading, risk management, and investment decisions

Roles:

- Data analysis & signal generation for trading strategies
- Backtesting models and calibrating parameters
- Supporting portfolio optimization and asset allocation
- Implementing models in code (usually Python)

Financial Engineering

Application of mathematical models and financial theory to design and **price complex financial instruments** and develop innovative financial products

Roles:

- Quantitative modeling
- Risk management
- Derivatives pricing
- Designing execution algorithms
- Implementing pricing models in code (usually C++)

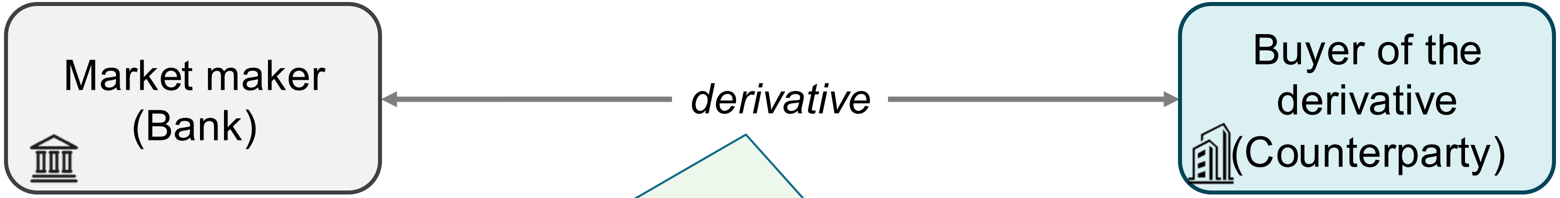
Asset classes and instruments

Each asset class has a range of instruments which can be used

Asset class	Instruments
Equities	Stocks, ETFs, derivatives
Fixed Income (rates)	Government bonds, ETFs, derivatives
Credit	Corporate bonds, ETFs, derivatives
FX	Spot, ETFs, derivatives
Commodities	ETFs, derivatives

Derivatives

Financial contracts whose value is derived from an underling asset

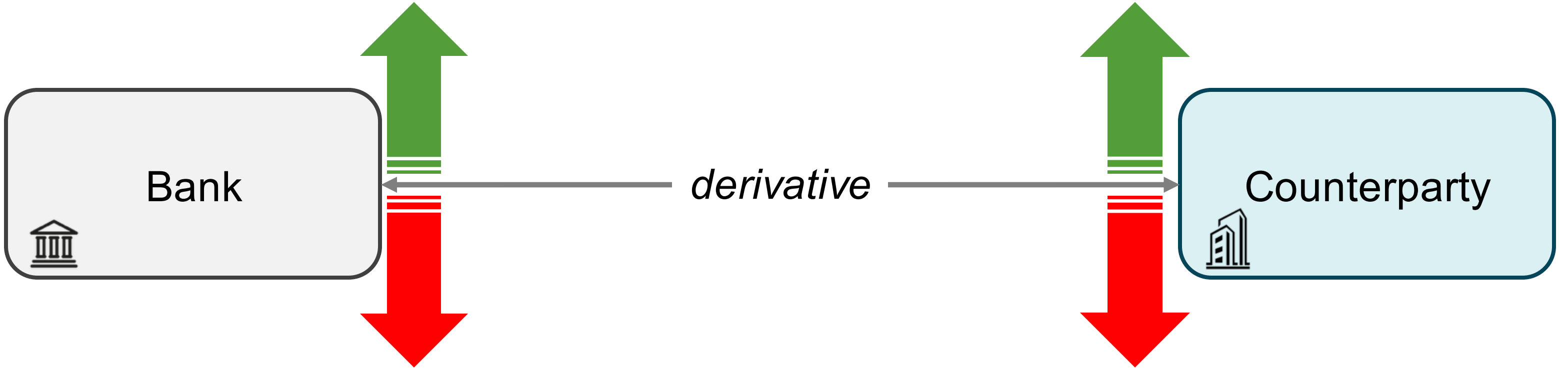


Typical derivative examples:

- FX forwards
- Futures
- Options
- Swaps (IRS, CCS, CDS, Commodity Swaps)
- Exotic derivatives

Derivatives

The derivative initially has a zero value, then its value fluctuates based on market conditions

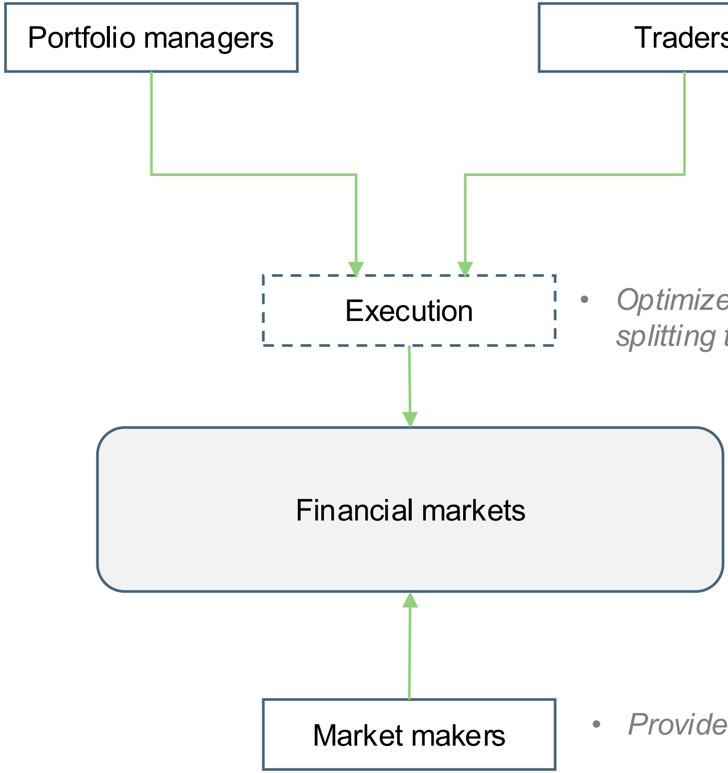


Hedging derivatives
Banks do not like being exposed to unpredictable market movements, so they **hedge** their position by opening an opposite trade with another bank or hedging with correlated but less expensive instruments

Schematic Overview of Financial Markets

Focus on the most influential actors

- *Decide the investment strategy*
- *Low frequency, large sizes*
- *Invest client liquidity*



Traders:

- *Decide trading strategy*
- *Higher frequencies, smaller sizes*
- *Invest own liquidity*

Execution:

- *Optimizes execution by splitting the order in time*

Financial markets can be:

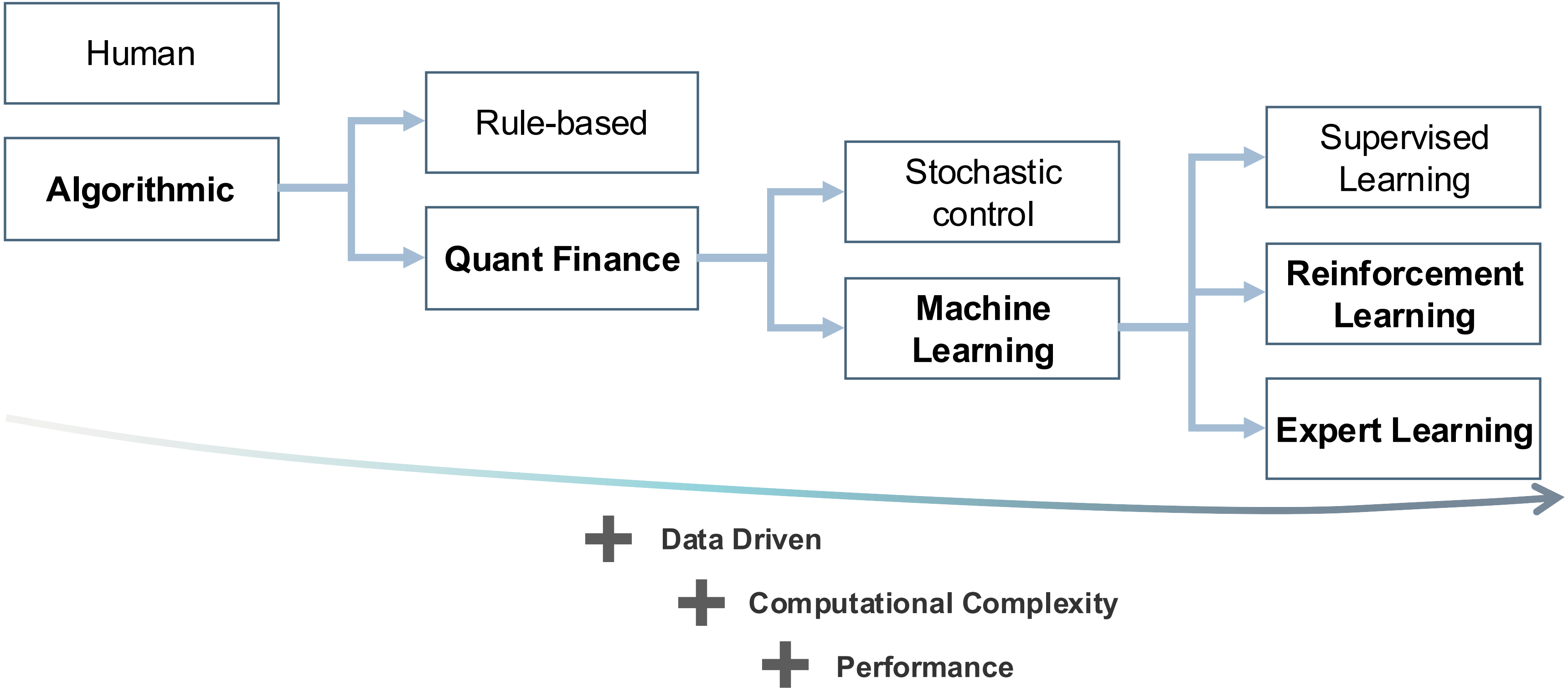
- *Regulated exchanges such as: NYSE, Nasdaq, LSE, Euronext*
- *Multi-lateral trade facilities*
- *Dark pools*

Market makers:

- *Provide liquidity to the financial markets*

Algorithmic Trading Technologies

Classification by technology type, with a focus on today's topics



Today’s Presentation

Financial Markets

- Market Making
- Execution
- Asset Managers and Hedge Funds
- Prop Traders
- Quants

Financial Engineering

- Stochastic processes
- Option pricing
- MC simulations
- Execution algorithms

Quant Toolkit

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Deep Learning
- Generative AI

Quant Research

- Classical systematic strategies
- Learning an FX trading strategy with RL
- Intraday trading with ML
- Cross sectional momentum with LLMs

Introduction to stochastic processes

Modeling uncertainty in finance

A **stochastic process** is a collection of random variables $\{X_t\}_{t>0}$ indexed by time.

Brownian motion $\{W_t\}_{t>0}$ is a key stochastic process used in finance:

- $W_0 = 0$ and $W_{t+\Delta t} - W_t \sim N(0, \Delta t)$
- W_t has independent increments and continuous paths

Geometric Brownian motion

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

Ito's Lemma

- Suppose X_t follows $dX_t = a(X_t, t)dt + b(X_t, t)dW_t$ and $f(X_t, t)$ is a twice-differentiable function. Then:

$$df = \left(\frac{\partial f}{\partial t} + a\frac{\partial f}{\partial x} + \frac{1}{2}b^2\frac{\partial^2 f}{\partial x^2}\right)dt + b\frac{\partial f}{\partial x}dW_t$$

Illustration of Brownian motion

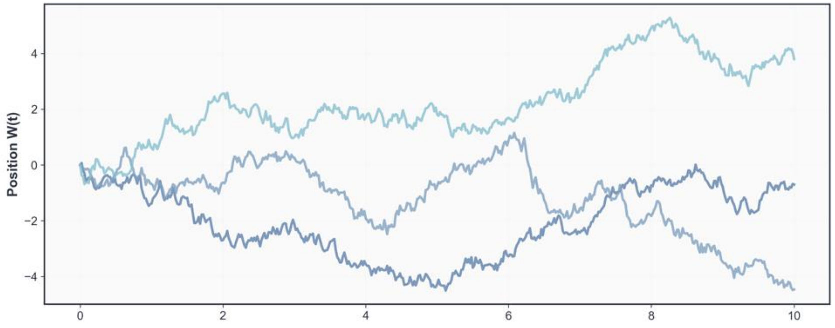
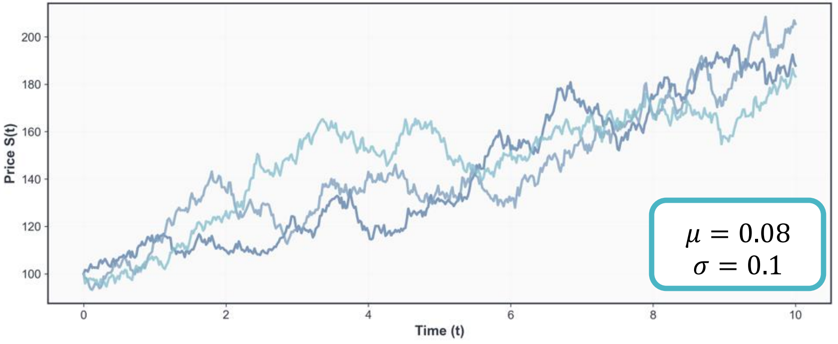


Illustration of Geometric Brownian motion



Black and Scholes model

For options pricing

The **Black-Scholes model**:

- is a **no-arbitrage framework** used to **price European-style options**
- assumes underlying is Brownian motion, trading in **continuous time** and have **no trading costs**
- is a **closed-form formula** to determine the fair value of a call or put option based on K, S, σ, r, T

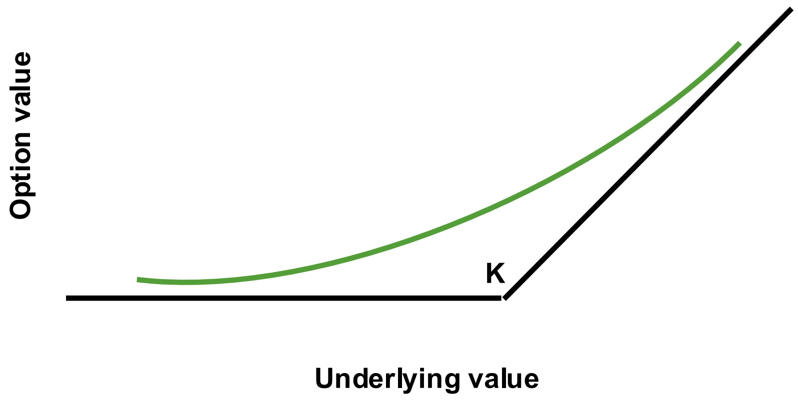
$$C(S_t, t) = N(d_1)S_t - N(d_2)Ke^{-rt}$$

$$d_1 = \frac{1}{\sigma\sqrt{T-t}}\left[\ln\frac{S_t}{K} + \left(r + \frac{\sigma^2}{2}\right)(\mathrm{T} - \mathrm{t})\right]$$

$$d_2 = d_1 - \sigma\sqrt{T-t}$$

Option **Greeks** are the partial derivatives of the price with respect to the parameter values

Call Option Price Diagram



Monte Carlo simulations

One of the most important tools in financial engineering

Monte Carlo simulation is a **numerical technique** used to **estimate expected values** by simulating many scenarios, it is used to **price derivatives** when closed-form solutions are not available

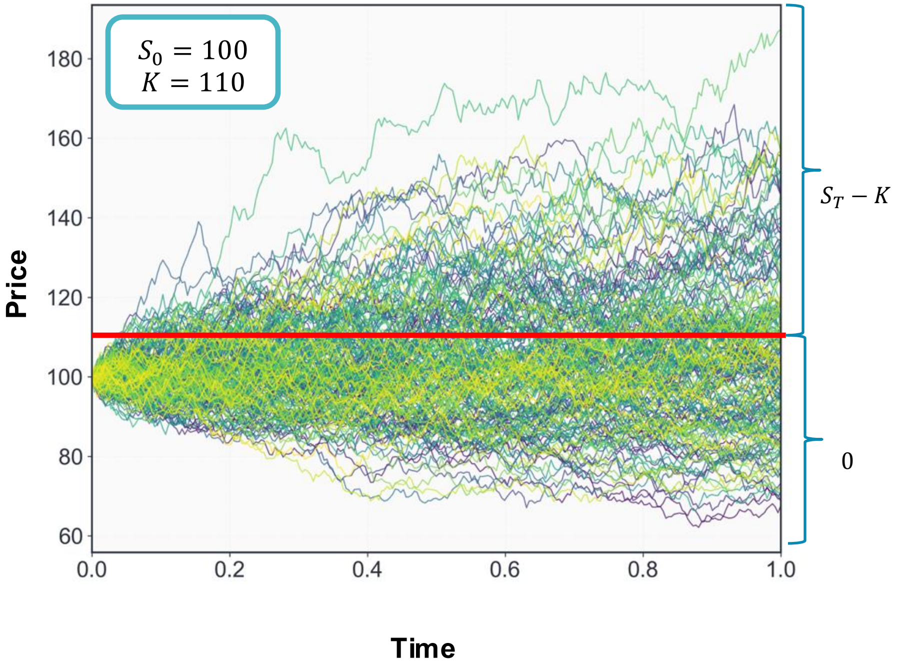
Applications

- Pricing of exotic **options**, credit **derivatives**, and structured products
- **Risk assessment**: VaR, Conditional VaR
- Stress testing and **scenario analysis**

Methodology

1. Calibrate stochastic process to the asset of interest
2. Simulate many price paths for the asset
3. Evaluate the payoff for each simulated path
4. Compute the average payoff and discount it to present value

Call option pricing example with MC simulations



Avellaneda Stoikov model

Stochastic control framework for market making in a LOB

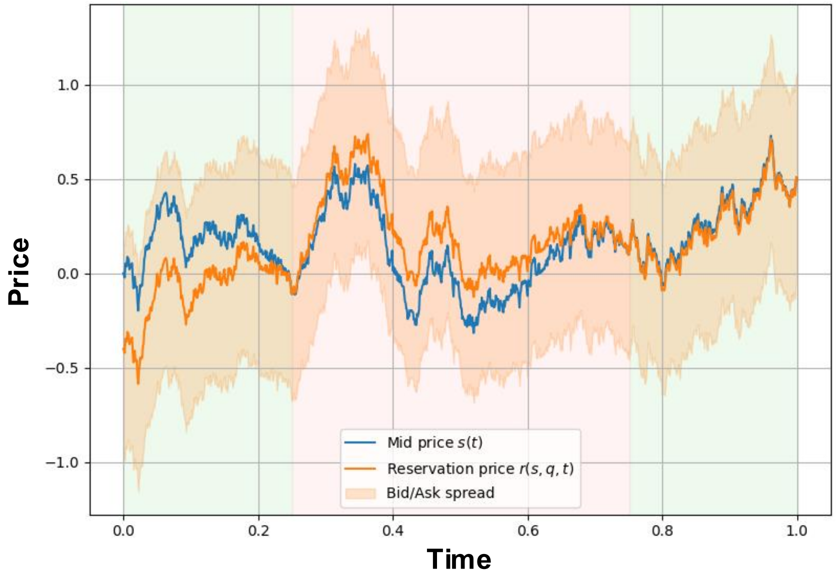
Assumptions:

- The mid-price follows a **Brownian motion**
- Market maker has **exponential utility function**: $U(W_T) = -e^{-\gamma W_T}$ where W_T is terminal wealth and γ is risk aversion
- Arrival rates of buy/sell market orders are **exponentially decreasing functions** of the distance between quoted prices and the mid-price: $Ae^{-k(q-m)}$

The market maker is continuously quoting a bid and ask prices calculated as a reservation price and plus the spread where:

- The reservation price is an adjustment to the mid-price, which accounts for the inventory held by the agent: $r = s - q\gamma\sigma^2(T-t)$
- The bid ask spread around the reservation price is: $\text{spread} = \gamma\sigma^2(T-t) + \frac{2}{\gamma}\ln(1+\frac{\gamma}{\mathrm{k}})$

Simulation of reservation price and spread



Almgren and Chriss

An optimal execution algorithm

Almgren and Chriss is a model which defines the optimal execution strategy given the following assumptions on the price impact:

- $P_k = P_{k-1} + \theta v_{k-1} + \eta_{k-1}$ where $\eta \sim \mathrm{IID}(0, \sigma)$, where θv_{k-1} is a **linear market impact** term
- $\bar{P} = P_k + \rho v_k + sign(v_k)\frac{S}{2}$ where S is the bid ask spread and ρv_k is a **linear temporary market impact** term

The **execution costs** are defined as:

- $C(v) = \sum_{k=0}^{N-1} v_k \bar{P} - X P_0$

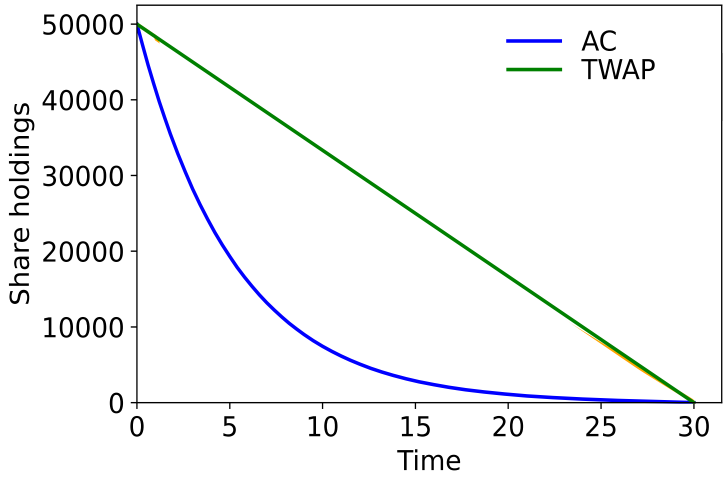
The objective function considers the trader's **risk aversion** λ and is defined as:

- $argmin_v \, \mathbb{E}[C(v)] - \lambda \mathrm{Var}(C(v))$

This problem has an analytical solution:

- $v_k = A \cosh(\sqrt{\frac{\lambda \sigma^2}{\rho}}(T - t_k))$

Illustration of Almgren Chriss execution trajectory



Today’s Presentation

Financial Markets

- Market Making
- Execution
- Asset Managers and Hedge Funds
- Prop Traders
- Quants

Financial Engineering

- Stochastic processes
- Option pricing
- MC simulations
- Execution algorithms

Quant Toolkit

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Deep Learning
- Generative AI

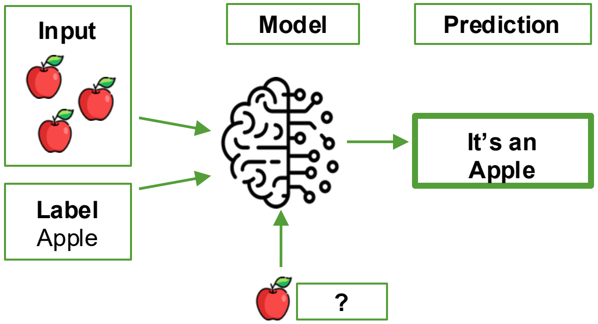
Quant Research

- Classical systematic strategies
- Learning an FX trading strategy with RL
- Intraday trading with ML
- Cross sectional momentum with LLMs

Introduction to Machine Learning

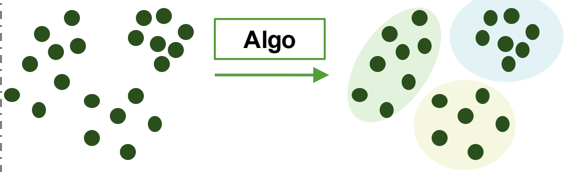
There are three main paradigms in ML

Supervised Learning



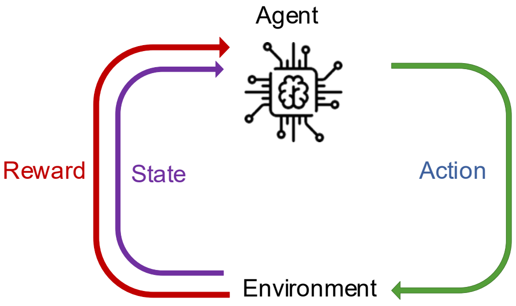
Usage
- Regression
- Classification

Unsupervised Learning



Usage
- Clustering
- Dimensionality reduction
- Feature extraction

Reinforcement Learning



Usage
- Optimal Control
- Game playing

Linear regression

Supervised learning

Definition

Method to model the **linear relationship** between one or more input variables (features) $\{x_i\}_{i \in I}$ and a **continuous** output (target) $\{y_i\}_{i \in I}$

Objective function

Minimize mean squared error: $\min \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2$

Evaluation metrics

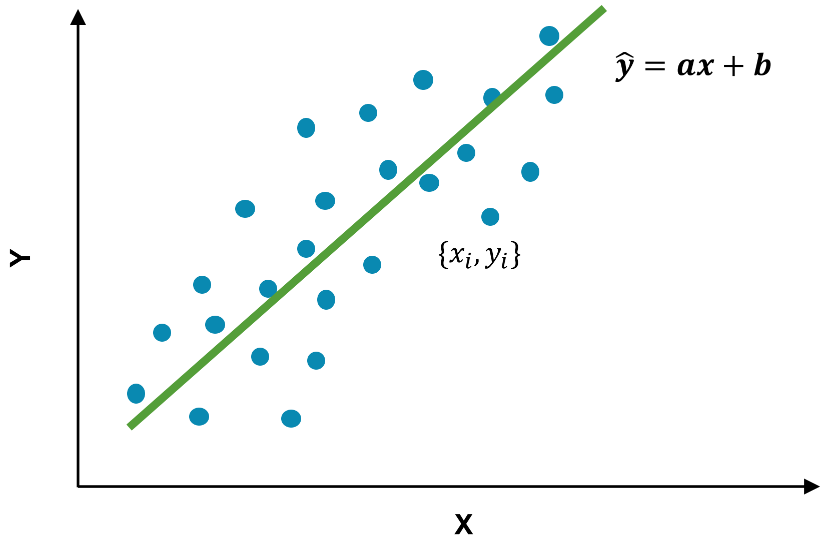
- Mean squared error
- $R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$

The linear regression has a closed form solution

Assumptions

- Independence of observations
- Constant variance
- Gaussian errors

Illustration of Linear Regression



Non-linear regression

Supervised learning

Definition

Method to model a **non-linear** relationship between one or more input variables (features) $\{x_i\}_{i \in I}$ and a continuous output (target) $\{y_i\}_{i \in I}$

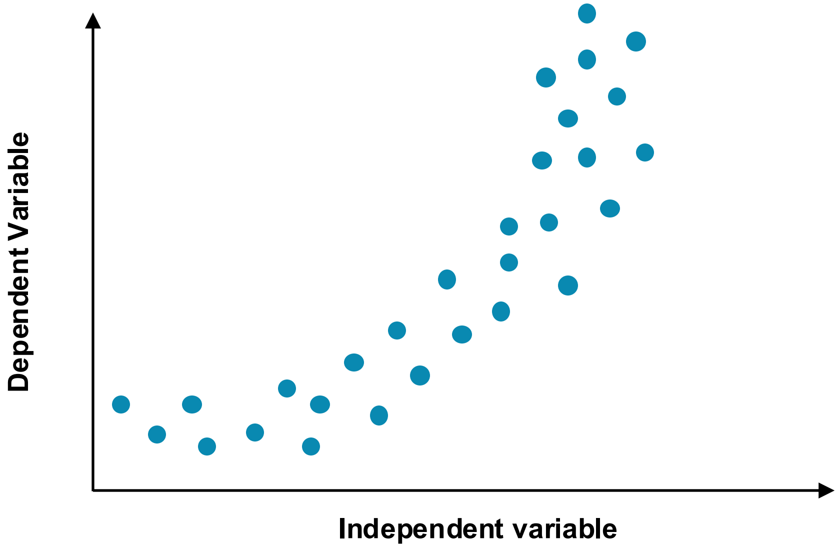
Models

- Polynomial regression
- Trigonometric regression
- Exponential regression
- Logarithmic regression
- Piecewise regression
- Tree based models
- Neural networks

Objective function

- Minimize mean squared error $\min \frac{1}{n} \sum_i (y_i - \hat{y})^2$

Illustration of non-linear regression



Classification

Supervised learning

Definition

Method used to predict to **which class** $\{0,1\}$ an input $\{y_i\}_{i \in I}$ belongs to

Models

- Logistic regression
- k-nearest neighbors
- Support vector machines
- Tree based methods
- Neural networks

Objective function

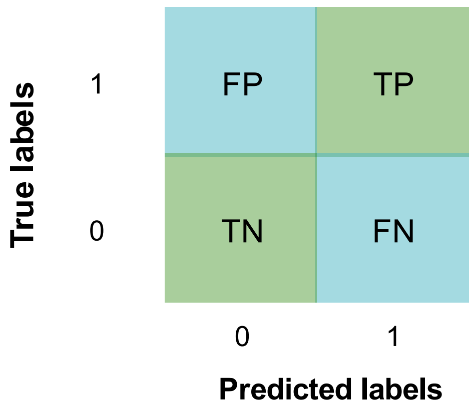
- Cross entropy: $\min \; -[y \log(\hat{y}) + (1-y)\log(1-\hat{y})]$

  Where $y \in \{0,1\}$ and $\hat{y} \in (0,1)$

Evaluation Metrics

- **Accuracy** = correct predictions / total
- **Precision, Recall**

Example of confusion matrix



Unsupervised learning

Definition

Method used to discover patterns, structure or groupings in the data. Useful when:

- You don't have labels
- You want to explore structure in data
- You're doing customer segmentation, anomaly detection, or data compression

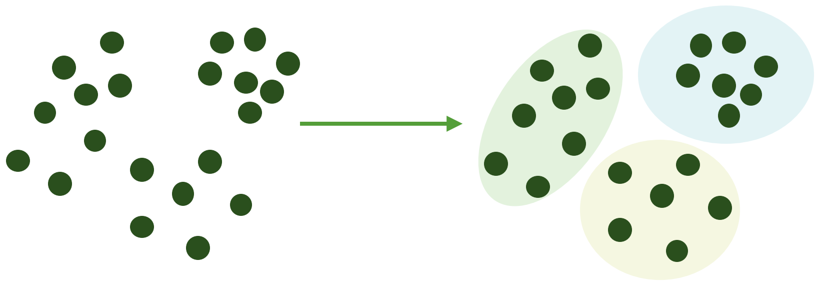
Methods

- **k-Means**: groups data into *k* clusters
- **PCA (Principal Component Analysis)**: projects data to fewer dimensions while preserving variance

Objective Function

- k-Means minimizes the distance between points and their cluster centroids
- PCA maximizes variance along new axes (principal components)

Illustration of clustering



Reinforcement Learning

Markov Decision Process: process which describes interaction between agent and environment

Definition

Reinforcement Learning (RL) is a paradigm where an agent learns to make decisions by interacting with an environment to maximize a long-term reward

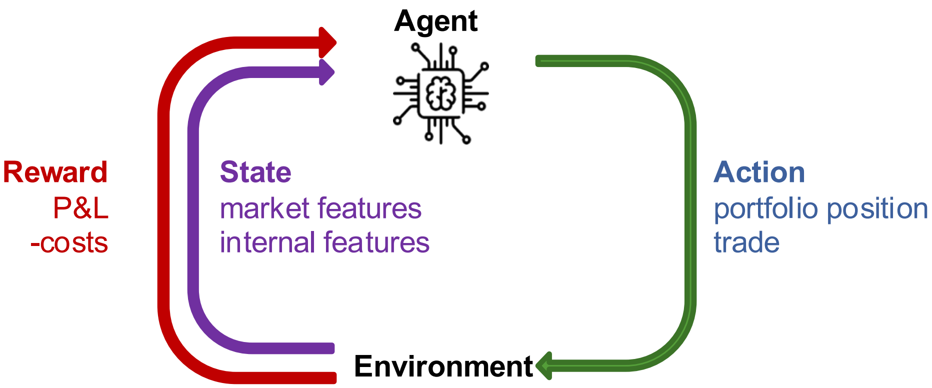
The agent is usually a regressor like a neural network or a tree based model

Markov Decision Process (MDP): process which describes interaction between agent and environment

Objective Function

- The objective is finding the policy π which maximizes the discounted sum of the rewards
- $J_\pi = \mathbb{E}_\pi[\sum \gamma^t R_t]$

Illustration of MDP



Q-function and Policy Search

RL algorithms enable the learning of the policy π

Q-learning

- Q-function

  $$Q_\pi = \mathbb{E}_\pi[\sum \gamma^t R_t | s_0, a_0]$$

- Bellman Equation

  $$Q_\pi = r(s,a) + \gamma \mathbb{E}_{s',a'}[Q_\pi(s',a')]$$

- Q-learning algorithm

  $$Q_t(s,a) = r(s,a) + \gamma \max_{a'} Q_t(s',a')$$

- Q-learning is a tabular algorithm which can be generalized using function approximators such as XGBoost

Algorithm examples

- DQN
- DDQN
- FQI

Policy Search

- Policy update

  $$\theta_{t+1} = \theta_t + \alpha \nabla_\theta J_{\pi_\theta}$$

- Policy gradient theorem

  $$\nabla_\theta J_{\pi_\theta} = \mathbb{E}[\nabla \log \pi_\theta(a|s) Q_{\pi_\theta}(s,a)]$$

- The policy is a parametric and differentiable function, usually a neural network

Algorithm examples

- REINFORCE
- TRPO
- PPO

Deep Learning

Advanced Supervised Learning

Definition

Deep Learning is a type of machine learning that uses **neural networks with many layers** to learn complex patterns in data

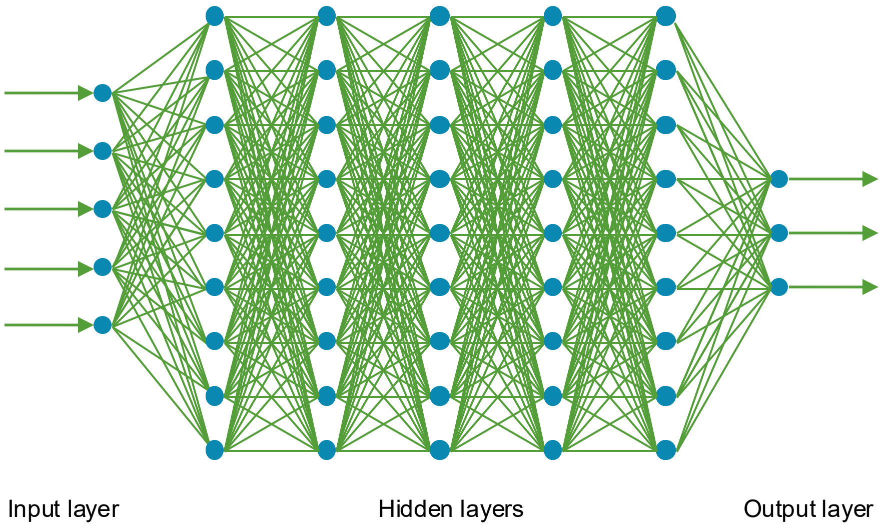
Common architectures

- Feed Forward
- LSTM (speech recognition)
- CNN (image recognition)
- Transformers (language)

Objective function

- MSE if regression task
- Categorical cross entropy if classification task

Illustration of Deep Neural Network



Generative AI

Deep learning taken to the extreme

Definition

Generative AI refers to models that can **generate new content** like text, images, music, or code that resembles the data they were trained on

Methods

- **Large Language Models:** deep neural networks trained on massive text datasets (often with billions of parameters). Next token prediction.

  Typically formulated as a **token-level classification problem**, predicting the next word or symbol in a sequence.

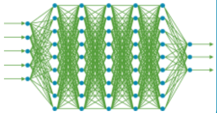
- **Variational Autoencoders:** neural networks that learn to **encode and decode data**, used to generate **new but similar** examples (commonly images).
- **Generative Adversarial Networks (GANs)**: Consist of a **generator** and a **discriminator** trained in competition. Used to create **high-quality synthetic data**, especially in computer vision.

Illustration of LLM workflow

Prompt: Give me an analysis on Nvidia

↓

LLM



↓

Nvidia is leading the AI surge with its Blackwell platform and new GPUs powering sovereign-AI initiatives in Europe—analysts see sovereign…

From data to generalization

Avoiding overfitting in ML

Dataset

A dataset is a collection of **examples** with:

- **Input data (features)**: image, text, numbers
- **Labels (targets)**: the correct answer (class or value)

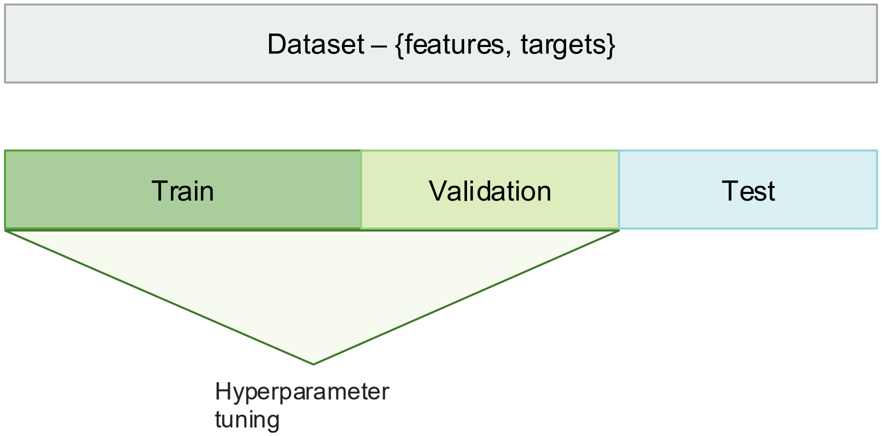
Data Processing

Normalize, clean, encode, or scale input features, then split the data into training set, validation set, testing set.

Key concepts

- **Overfitting**: When the model learns **too much from the training data**, and performs poorly on test data
- **Regularization** are techniques to **reduce overfitting** by simplifying the model such as adding an **L2 / L1 penalty, using dropout or early stopping**
- **Hyperparameter tuning** is the process of selecting the best set of hyperparameters for a ML model

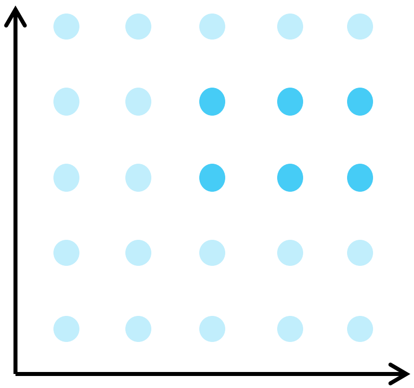
Dataset splitting illustration



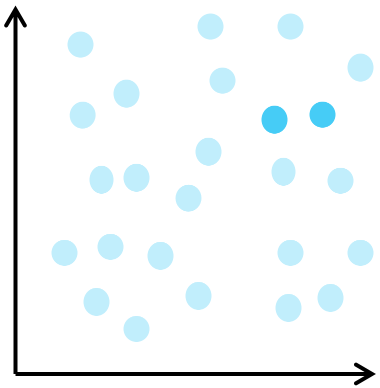
Hyperparameter Tuning

Training these models make take a large amount of time and compute power

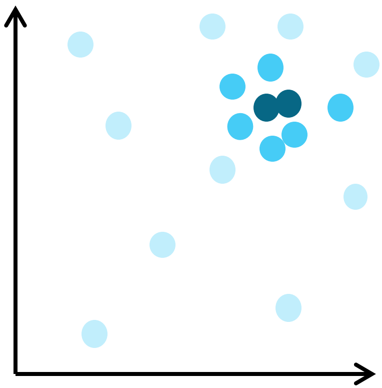
Grid search



Random search



Bayesian search



References

- Akiba, Takuya, et al. "Optuna: A next-generation hyperparameter optimization framework." *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019
- Bergstra, James, Dan Yamins, and David D. Cox. "Hyperopt: A Python Library for Optimizing the Hyperparameters of Machine Learning Algorithms." *SciPy* 13 (2013): 20.
- Hutter, Frank, Holger H. Hoos, and Kevin Leyton-Brown. "SMBO: Sequential model-based optimization for general algorithm configuration." *Learning and Intelligent Optimization: 5th International Conference*, 2011.

Today’s Presentation

Financial Markets

- Market Making
- Execution
- Asset Managers and Hedge Funds
- Prop Traders
- Quants

Financial Engineering

- Stochastic processes
- Option pricing
- MC simulations
- Execution algorithms

Quant Toolkit

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Deep Learning
- Generative AI

Quant Research

- Classical systematic strategies
- Learning an FX trading strategy with RL
- Intraday trading with ML
- Cross sectional momentum with LLMs

AGENDA

- **Classic systematic strategies**
- Learning an FX trading strategy with RL
- Trading futures with ML
- Cross sectional momentum with LLMs

Introduction to Quantitative Research

Defining and building a quantitative trading strategy

Quantitative trading definition

Quantitative trading uses mathematical and statistical models to identify trading opportunities

Common quantitative trading strategies

Focus next

- Momentum
- Mean-reversion
- Seasonality
- Statistical arbitrage
- Market making
- Alternative data

Building a quant trading strategy

A Objective	Financial assets, frequency, style
B Data	Price, LOB, sentiment, fundamental, economic
C Strategy	Define trading rules for the strategy
D Testing	Performance evaluation on historical data (backtesting)
Production	Connect to market via APIs, deploy strategy on a server

Momentum strategy (1/2)

Are there any clearly visible patterns in the training set?

Strategy description

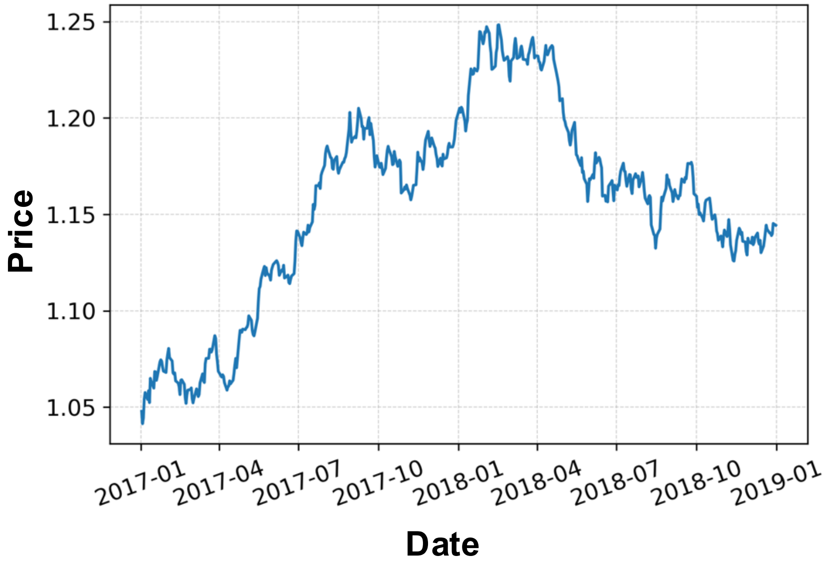
A Objective

- **Asset**: FX currency pair
- **Frequency**: daily
- **Style**: long/short

B Data

- **Data:** prices

Price process of asset on training set



Momentum strategy (1/2)

Are there any clearly visible patterns in the training set?

Strategy description

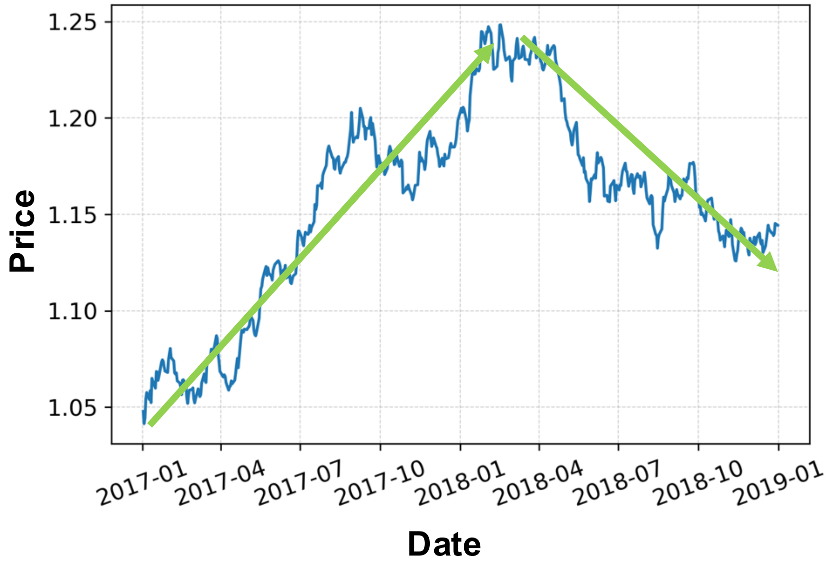
A Objective

- **Asset**: FX currency pair
- **Frequency**: daily
- **Style**: long/short

B Data

- **Data:** prices

Price process of asset on training set



Momentum strategy (2/2)

Choose parameters on training set and analyze P&L on testing set

Strategy description

A Objective

- **Asset**: FX currency pair
- **Frequency**: daily
- **Style**: long/short

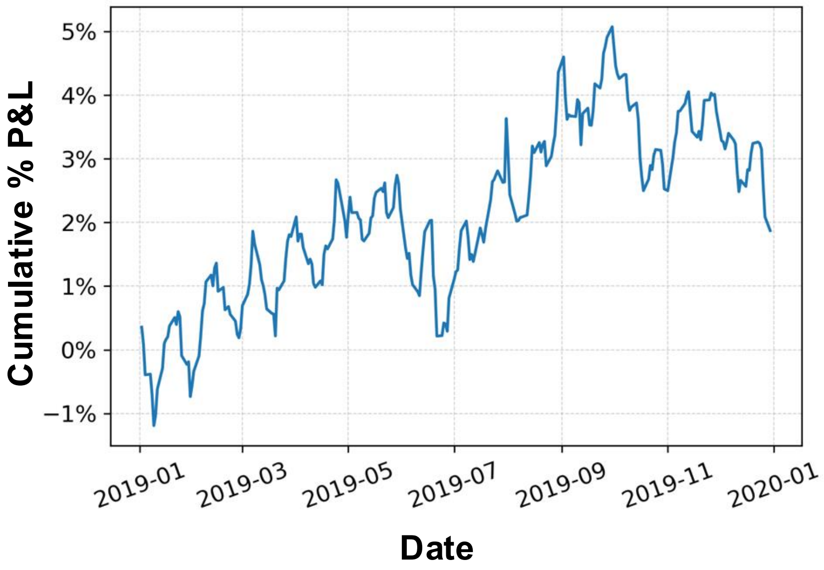
B Data

- **Data:** prices

C Strategy

+1 if $\mathrm{MA}_{\mathrm{short}} > \mathrm{MA}_{\mathrm{long}}$
-1 if $\mathrm{MA}_{\mathrm{short}} < \mathrm{MA}_{\mathrm{long}}$
where MA is the moving average

D Testing Cumulative % P&L on 2019



Performance metrics

The basic performance metrics to evaluate a strategy

Performance metrics

- **Annualized P&L:**

$$\text{mean}\left(P\&L_{daily}\right) * 252$$

- **Annualized vol:**

$$\text{std}\left(P\&L_{daily}\right) * \sqrt{252}$$

- **Sharpe ratio:**

$$\frac{P\&L}{vol}$$

- **Maximum drawdown:**

$$Trough \ - Peak$$

Momentum strategy metrics on 2019

- **Annualized P&L:** 1.9%
- **Annualized vol:** 4.6%
- **Sharpe ratio:** 0.41
- **Maximum drawdown**: -3.3%

Mean reverting strategy (1/4)

Observe the average daily price movement with confidence intervals

Strategy description

A Objective

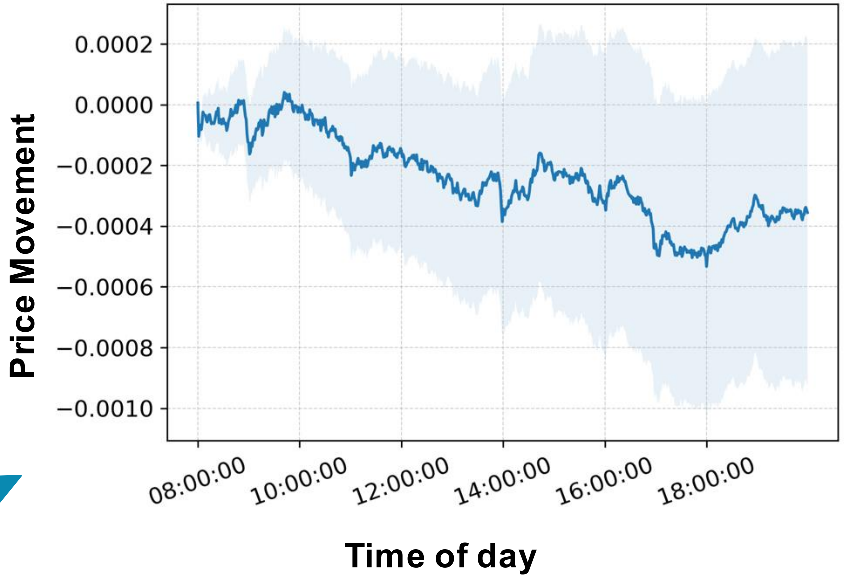
- **Asset**: FX currency pair
- **Frequency**: intraday
- **Style**: long/short

B Data

- **Data:** prices

The wide confidence intervals around the average asset movement suggest a mean reverting approach

Average asset movement



Mean reverting strategy (2/4)

The strategy achieves a surprisingly good result in the testing set

Strategy description

A Objective

- **Asset**: FX currency pair
- **Frequency**: intraday
- **Style**: long/short

B Data

- **Data:** prices

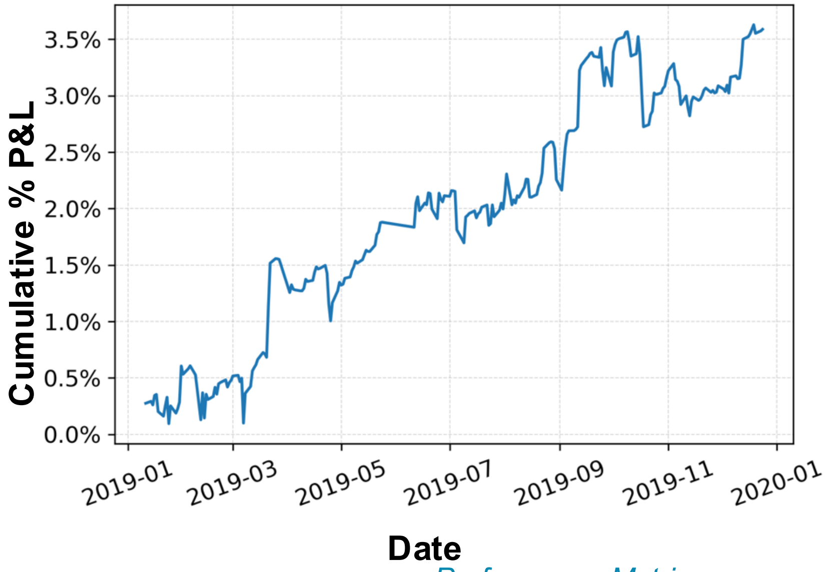
C Strategy

- Positioning: $-\sum_{i=0}^{T-2} f(i) R_{t-i}$

Legend

T = time horizon in minutes
R = returns

D Testing Cumulative % P&L on 2019



Performance Metrics

- Annualized return: 3.8%
- Sharpe: 2.03
- Max drawdown: -0.84%

Mean reverting strategy (3/4)

Each trades generates a cost proportional to the trade size

Defining Transaction Costs

- $\text{mid price} = \frac{1}{2}(\text{best ask} + \text{best bid})$
  - 4044.75
- $\text{spread} = (\text{best ask} - \text{best bid})$
  - 0.5
- $\text{transaction costs} = \text{trade size} * \frac{1}{2}\text{spread}$
- $\text{step P\&L} = \text{position} * \text{market movement} - \text{transaction costs}$

Example of LOB

BID		ASK	
Volume	Price	Price	Volume
136	4044.50	4045.00	62
327	4044.00	4045.50	293
348	4043.50	4046.00	427
620	4043.00	4046.50	426
358	4042.50	4047.00	463
330	4042.00	4047.50	348
325	4041.50	4048.00	327
318	4041.00	4048.50	294
305	4040.50	4049.00	281
512	4040.00	4049.50	288

Mean reverting strategy (4/4)

Adding trading costs causes the performance to degrade

Strategy description

A Objective

- **Asset**: FX currency pair
- **Frequency**: intraday
- **Style**: long/short

B Data

- **Data:** prices

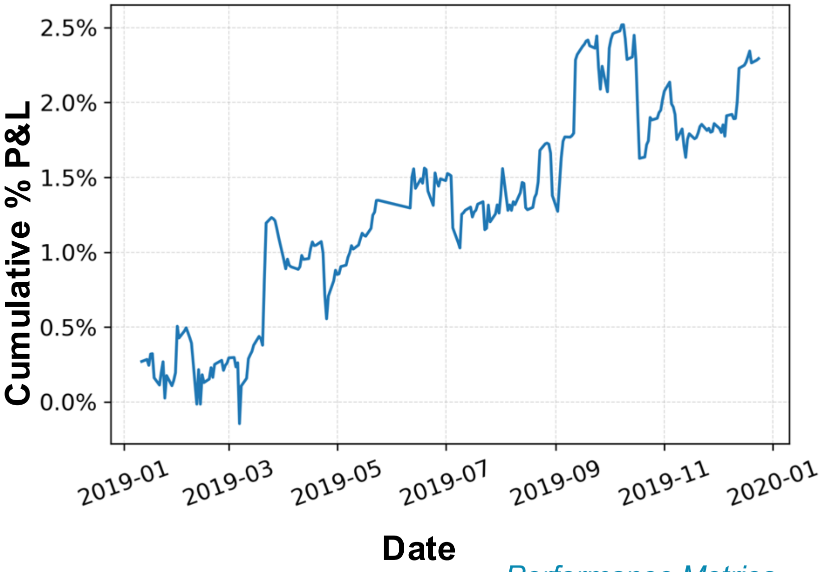
C Strategy

- Positioning: $-\sum_{i=0}^{T-2} f(i) R_{t-i}$

Legend

T = time horizon in minutes
R = returns

D Testing Cumulative % P&L on 2019

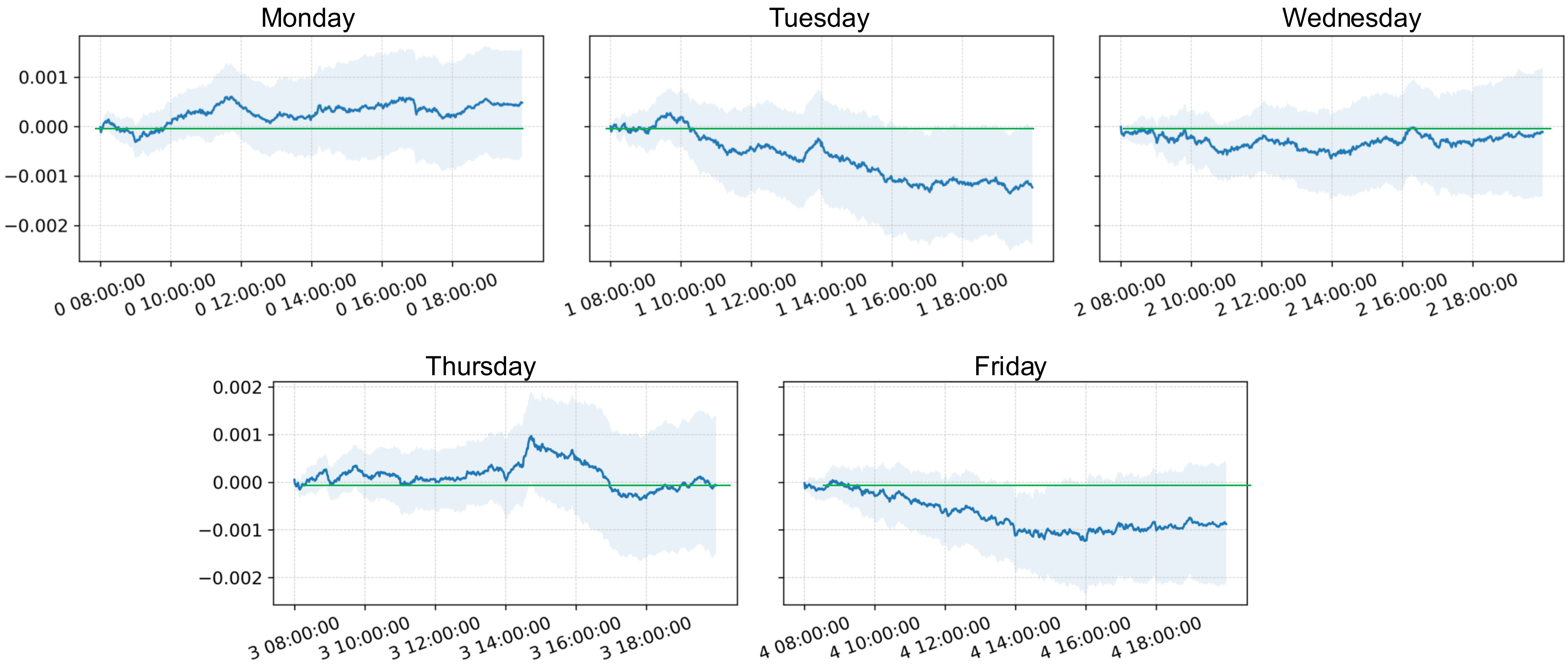


Performance Metrics

- Annualized return: 2.4%
- Sharpe: 1.2
- Max drawdown: -0.89%

Intraday seasonality strategy (1/2)

Analyze the average behavior on each day of the week



Intraday seasonality strategy (2/2)

Finding intervals where the cumulated intraday price is significantly different from zero

Strategy description

A Objective

- **Asset**: FX currency pair
- **Frequency**: intraday
- **Style**: long/short

B Data

- **Data:** prices

C Strategy

If on the training set the return is > 0 with statistical significance: long, elif it is < 0: go short

D Testing Cumulative % P&L on 2019

Performance Metrics

- Annualized return: 3.0%
- Sharpe: 1.06
- Max drawdown: -1.64%

Combining strategies (1/3)

By combining uncorrelated strategies, it is possible to improve the Sharpe ratio

Sharpe of sum of two strategies

- Sharpe$(S_1) = \frac{\mu_1}{\sigma_1}$, Sharpe$(S_2) = \frac{\mu_2}{\sigma_2}$
- Sharpe$(S_1 + S_2) = \frac{\mu_1+\mu_2}{\sqrt{\sigma_1^2+\sigma_2^2+2\rho\sigma_1\sigma_2}}$

Correlation: ρ	Sharpe(S1+S2)
1	$\frac{\mu_1 + \mu_2}{\sigma_1 + \sigma_2}$
0	$\frac{\mu_1 + \mu_2}{\sqrt{\sigma_1^2 + \sigma_2^2}}$
-1	$\frac{\mu_1 + \mu_2}{\lvert\sigma_1 - \sigma_2\rvert}$

Since $(\sigma_1 + \sigma_2)^2 > \sigma_1^2 + \sigma_2^2$, uncorrelated strategies are preferrable

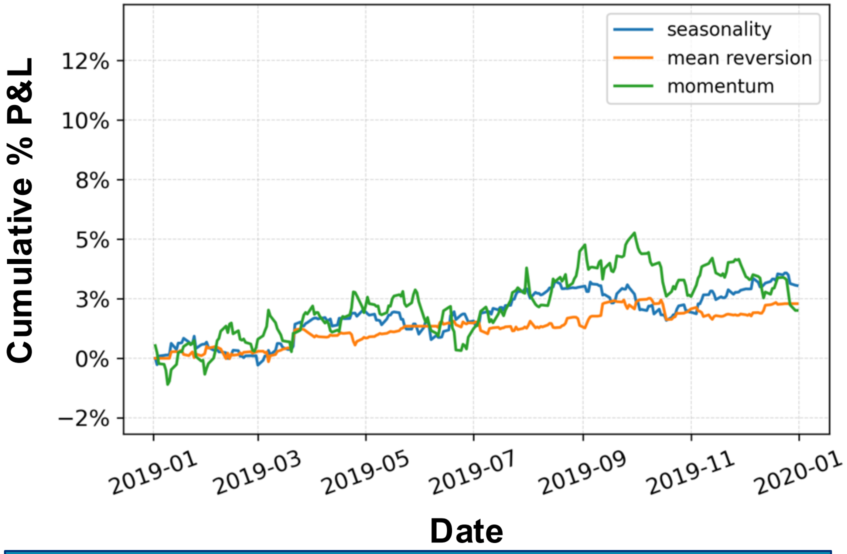
Correlation of our strategies

	Momentum	Mean reverting	Seasonality
Momentum	1.00	-0.18	0.25
Mean reverting	-0.18	1.00	-0.10
Seasonality	0.25	-0.10	1.00

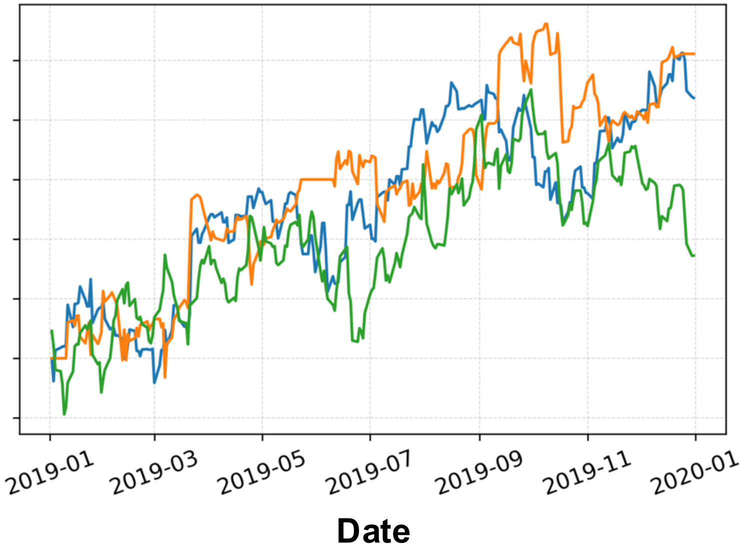
Combining strategies (2/3)

By changing the size of the strategies, we can modify the relative volatility

Strategies with same size



Strategies with equal vol



The size multiplier is $\boldsymbol{m_i} = \frac{\sigma}{\sigma_i}$ where σ_i is the volatility of strategy I and σ the target volatility. Choose a level of volatility you are comfortable with, 0.10 in the example

	Momentum	Mean reversion	Seasonality
m_i	2.1	5.6	3.6

Combining strategies (3/3)

A naive approach is to give the same weight to each strategy

Cumulative % P&L on 2019

Performance metrics

	Sharpe	Vol
Aggregated	1.6	0.06
Momentum	0.4	0.10
Variance	1.2	0.10
Seasonality	1.1	0.10

We can see that by averaging the strategies, the Sharpe improves as volatility has decreased. Since we are comfortable with a 0.10 vol, we can increase the size of the strategy.

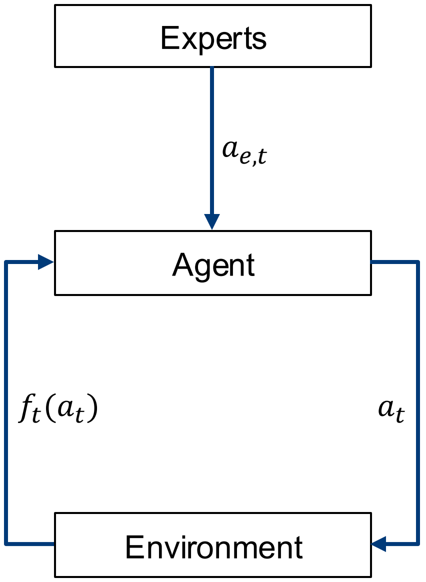
Expert Learning (1/2)

A dynamic, data driven approach to portfolio optimization

Characteristics

- While classical portfolio optimization models usually necessitate an estimation of the covariance matrix and the returns, expert learning algorithms are data driven
- Field of research close to reinforcement learning
- Objective is to learn **sequential decision processes**
- Online algorithms with **no training phase**
- Expert learning algorithms choose at each timestep which experts to follow
- Regret guarantees: finding the **best expert in sub-linear time**
- **Regret** $R_T = \sum_{t=1}^{T} f_t(a_t, y_t) - \inf_{e \in E} \sum_{t=1}^{T} f_t(a_{e,t}, y_t)$

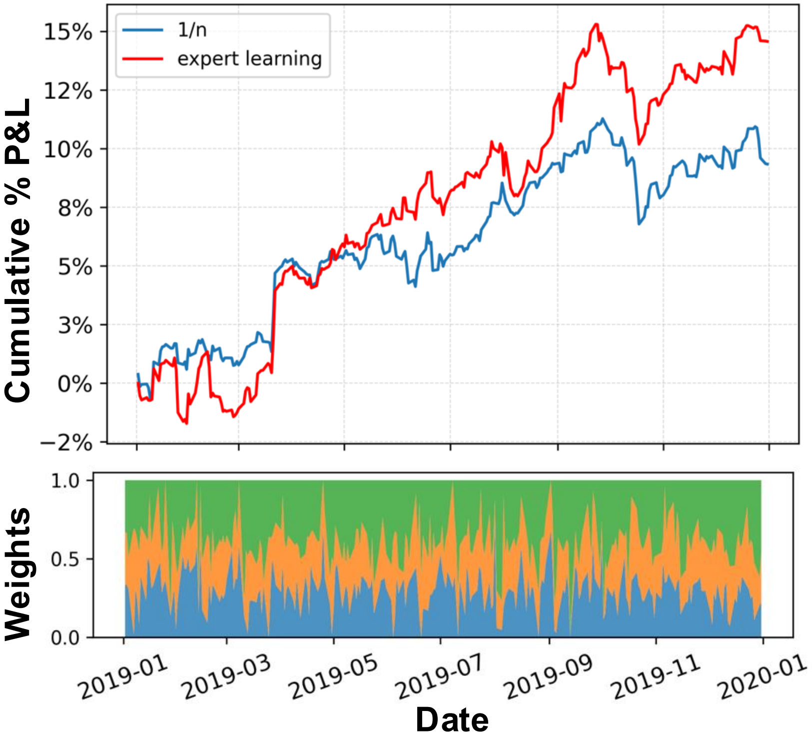
Expert interaction scheme



Expert Learning (2/2)

Combining strategies using Expert Learning

Cumulative % P&L on 2019



Performance metrics

	Sharpe	Vol
Expert learning	2.0	0.06
1/n	1.6	0.06

Comparing with the average helps to understand whether there is value in applying expert learning for dynamic rebalancing. Also in the expert learning case, volatility is reduced meaning we can increase the size.

Industry standards of quant trading strategies

Higher frequencies mean higher infrastructure costs

The focus of the following slides is at this frequency

Frequency →	Monthly	Daily	Intraday	Second	Nanosecond
Capacity ←	Billions	100s millions	10s millions	Few millions	Small size
Sharpe →	~0.5	<=1	>=1	>=2	>8
Data availability →	Company earnings	Price Sentiment Macro	Price	LOB TAQ	Tick by tick order flow

Reinforcement Learning requires hundreds of thousands of datapoints to learn a trading strategy, which requires an intraday frequency.
High frequency trading requires extremely expensive infrastructure. It is necessary to find a compromise

How can we improve?

Recap the current approach

Devise uncorrelated strategies

Daily momentum strategy

Intraday mean reverting

Intraday seasonality

Normalize

Normalize the trading strategies

Combine

Combine using expert learning

How can we use machine learning to improve the underlying strategies?

- consider costs when generating the strategy
- move on from a strictly defined trading rule

AGENDA

- Classic systematic strategies
- **Learning an FX trading strategy with RL**
- Trading futures with ML
- Cross sectional momentum with LLMs

Reinforcement Learning for Quantitative Trading

Problem description and MDP definition

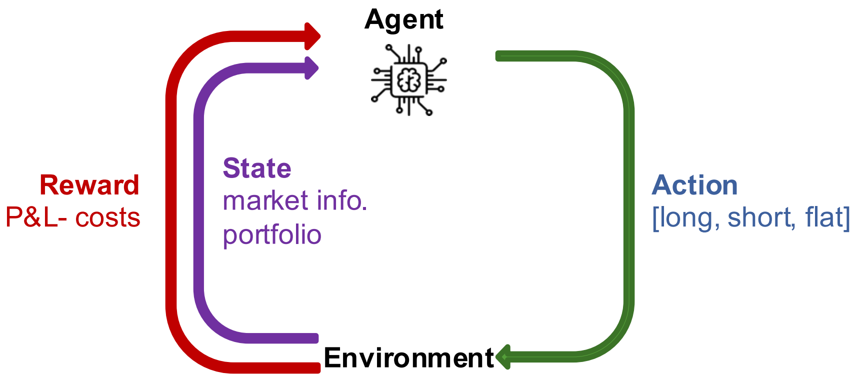
Definition

- At each timestep, decide whether to go long, short or flat to maximize gains

Markov Decision Process (MDP)

- **State:** price window, bid-ask spread, current portfolio, date/time
- **Action:** long, short, flat
- **Reward:** P&L – transaction costs

MDP Graphic



- The objective is finding the policy π which maximizes the discounted sum of the rewards
- $J_\pi = \mathbb{E}_\pi[\sum \gamma^t R_t]$

Reinforcement Learning for FX trading

Experimental results - performance

Strategy description

A Objective

- **Asset**: FX currency pair
- **Frequency**: intraday
- **Style**: long/short

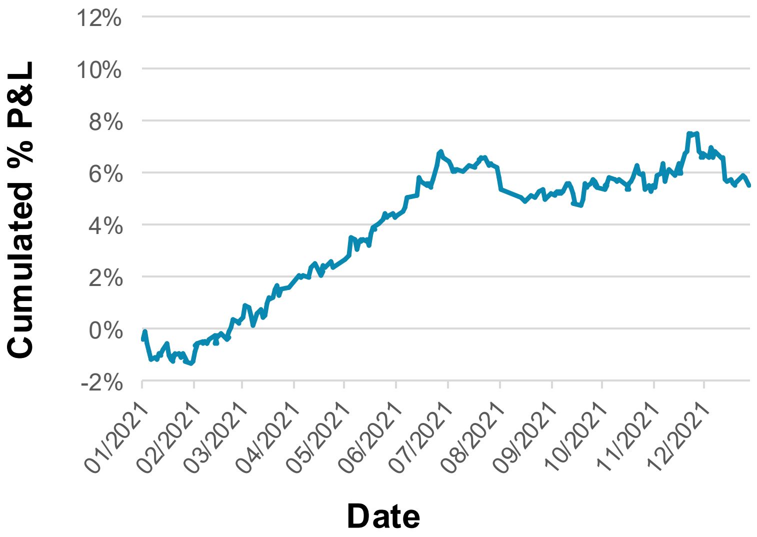
B Data

- **Data:** prices

C Strategy

Training with reinforcement learning on 2018-2019, validation on 2020

D Testing Cumulative % P&L on 2021



Performance Metrics

- Annualized return: 6.0%
- Sharpe: 1.7
- Max drawdown: -2.0%

Riva, Antonio, et al. "Learning FX trading strategies with FQI and persistent actions." *ICAIF* 2021.

Reinforcement and Expert Learning for FX trading

Experimental results - performance

P&L of backtest of RL strategies on 2021

Performance metrics

Strategy	Return	Sharpe
Expert 1	6.8%	2.2
Expert 2	8.4%	2.1
Agent	4.3%	1.9
Expert 3	7.0%	1.8
Expert 4	3.3%	1.5
Expert 5	4.4%	1.3
Expert 6	3.6%	1.1
Expert 7	0.8%	0.7
Expert 8	1.7%	0.5
Expert 9	-6.4%	-2.5

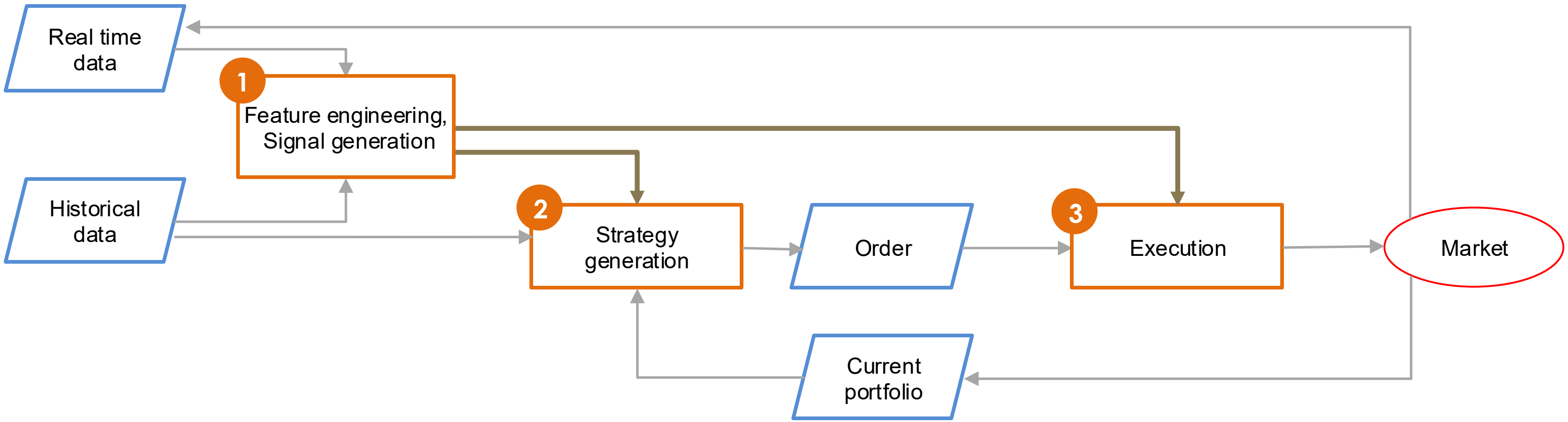
Riva, Antonio, et al. "Addressing non-stationarity in FX trading with online model selection of offline RL experts." *ICAIF* 2022.

AGENDA

- Classic systematic strategies
- Learning an FX trading strategy with RL
- **Trading futures with ML**
- Cross sectional momentum with LLMs

Intraday Trading with Machine Learning

End-to-end workflow to build an intraday trading strategy



Automatic trading algorithms are comprised of 3 main components:

1. The **feature engineering** and **signal generation** component extracts **features** and **alpha signals** from the LOB
2. The **second component** takes the alpha signals and compressed LOB information to generate a **trading strategy**
3. The **execution algorithm** has the objective of minimizing transaction costs and controlling market impact

Reinforcement Learning for FX trading

Experimental results - performance

Strategy description

A Objective

- **Asset**: Futures
- **Frequency**: intraday/seconds
- **Style**: long/short

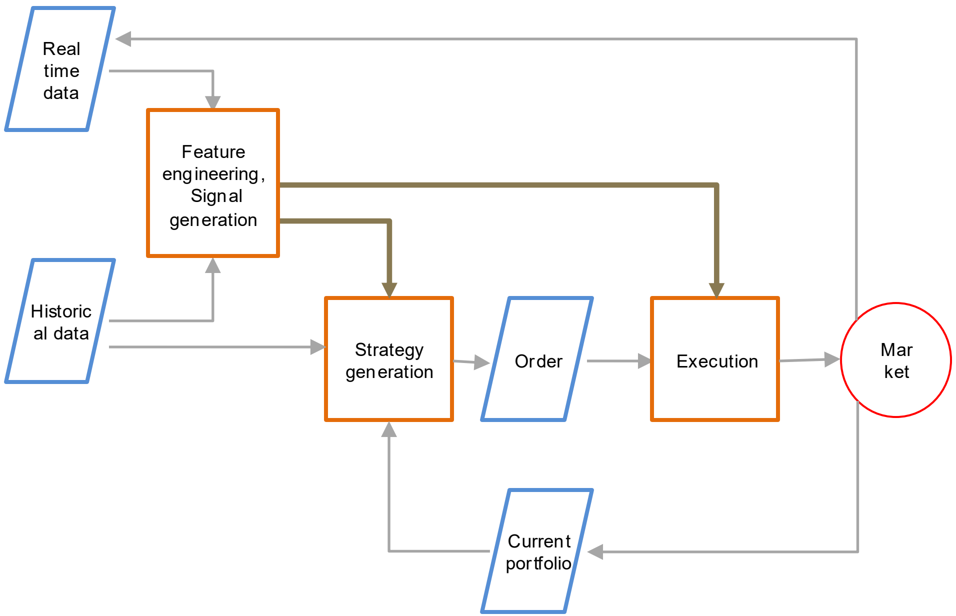
B Data

- **Data:** LOB, TAQ

C Strategy

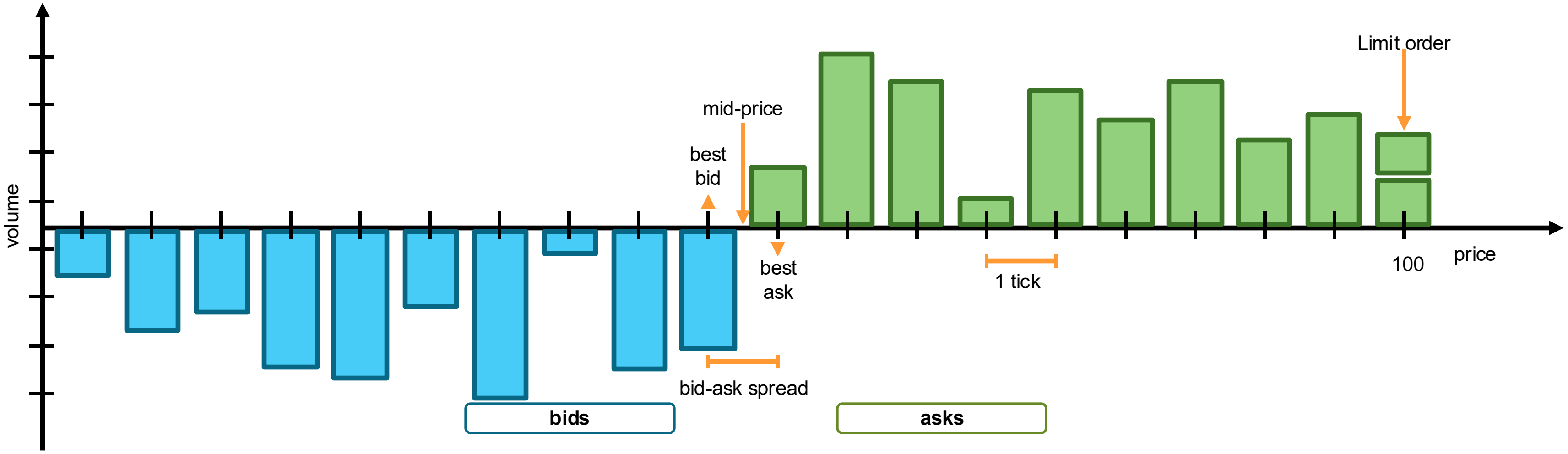
Combine deep learning and reinforcement learning to create a trading strategy

Workflow



1 Limit Order Book (LOB) Data

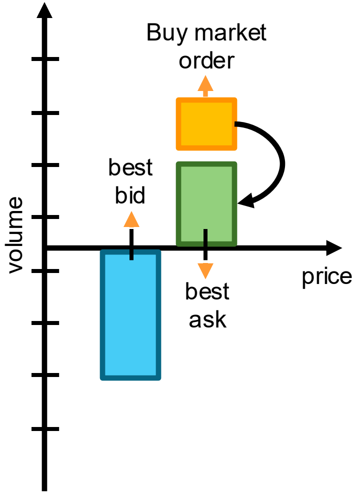
LOB data contains 10 price levels on the bid and on the ask



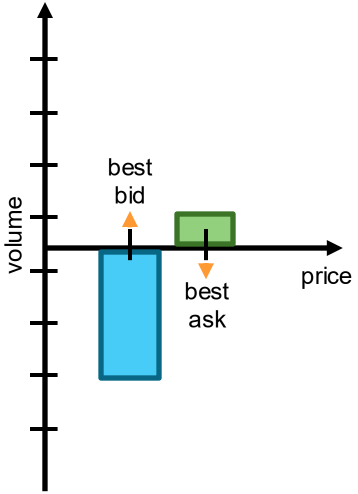
1 Trades and Quotes (TAQ) Data

TAQ data contains only best bid, best ask and executed orders

TAQ dataset time T_0



TAQ dataset time T_1



1 Processing the Raw Data

LOB and TAQ data have timestamps at nanosecond (10^{-9}) precision

Data Quality

- Handle nans/gaps (forward fill?)
- Merge the LOB and TAQ datasets
- Be careful with datetime features
- Handle roll dates

Working frequency

- At this step it is necessary to decide the frequency you want to work with:
    - Tick by tick
    - Downsample every x ticks
    - Downsample every x seconds

1 Extracting Features from the LOB

We can use a classical approach, a machine learning approach or a combined one

Classical approach

Use hand crafted features such as:

- Autocovariance of the price
- Order Flow Imbalance*

$$OF_{b,t} \begin{cases} v_{b,t} \text{ if } p_{b,t} > p_{b,t-1} \\ v_{b,t} - v_{b,t-1} \text{ if } p_{b,t} = p_{b,t-1} \\ -v_{b,t} \text{ if } p_{b,t} < p_{b,t-1} \end{cases}$$

$$OFI_t = OF_{b,t} - OF_{a,t}$$

- Volume imbalance

$$\frac{v_b - v_a}{v_b + v_a}$$

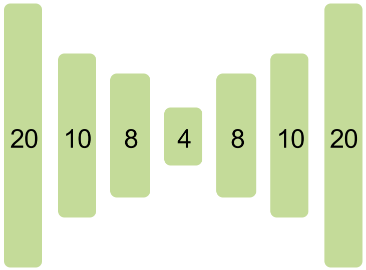
- Trade imbalance

$$\sum_{N(t_{k-1})}^{N(t_k)} b_n - \sum_{N(t_{k-1})}^{N(t_k)} s_n$$

Machine Learning approach

- Use convolutional neural networks to extract features†
- Compress the information with autoencoders

Autoencoder example



*Cont, Rama, Arseniy Kukanov, and Sasha Stoikov. "The price impact of order book events." *Journal of financial econometrics* 12.1 (2014): 47-88.
† Zhang, Zihao, Stefan Zohren, and Stephen Roberts. "Deeplob: Deep convolutional neural networks for limit order books." (2018). *IEEE Transactions on Signal Processing* 67.11

1 Generating Trading Signals

The objective is to accurately predict the direction of the price movement

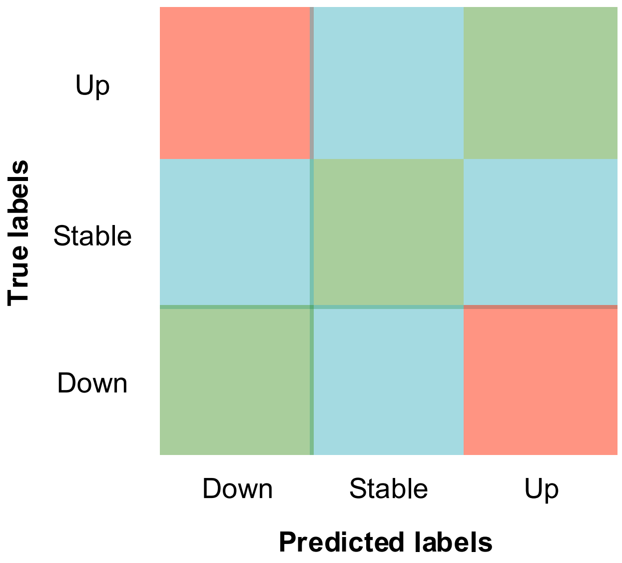
Defining the target

- target $\begin{cases} (\text{mid}_{t+x\text{sec}} - \text{mid}_t) < -\theta \rightarrow \text{Down} \\ -\theta < (\text{mid}_{t+x\text{sec}} - \text{mid}_t) < \theta \rightarrow \text{Stable} \\ (\text{mid}_{t+x\text{sec}} - \text{mid}_t) > \theta \rightarrow \text{Up} \end{cases}$
- $\text{mid}_{t+x\text{ticks}} - \text{mid}_t$
- Should I consider an average price to smooth out the noise?
- Should I look at bid and ask prices instead of mid

Classifier choice

- DeepLOB, LSTM, MLP
- Xgboost, Light GBM, Extra Trees
- Ensemble of predictions
- Hyperparameter tuning

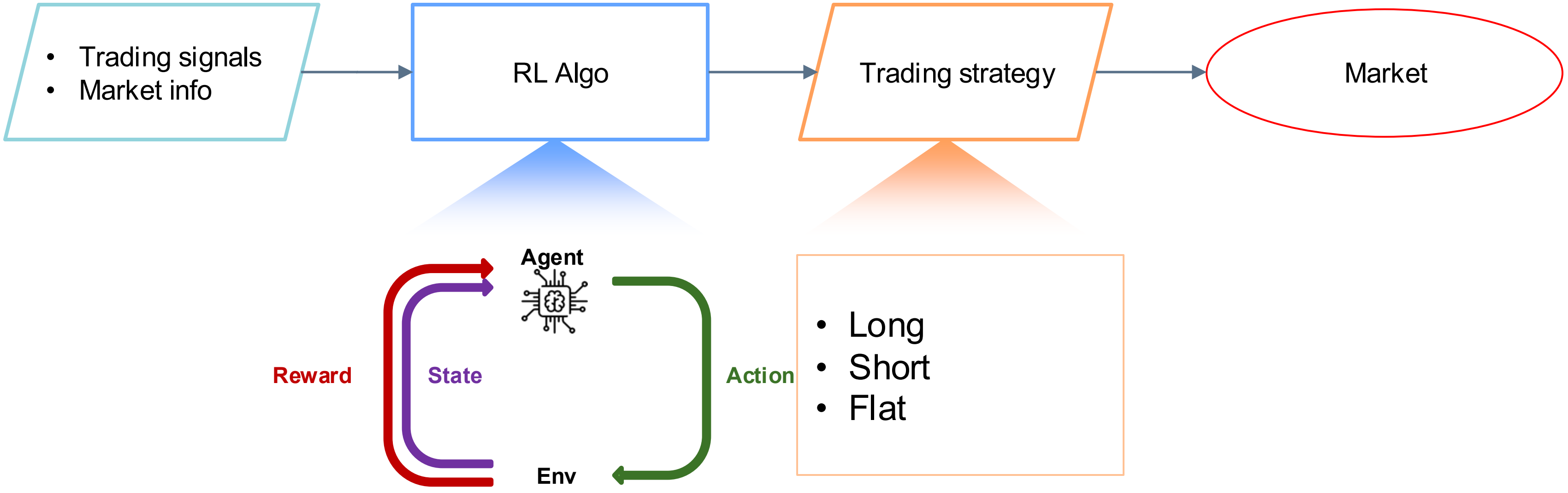
Example of confusion matrix



2 Defining the Trading Strategy Using RL

RL has the task of optimizing the trading strategies taking into account transaction costs

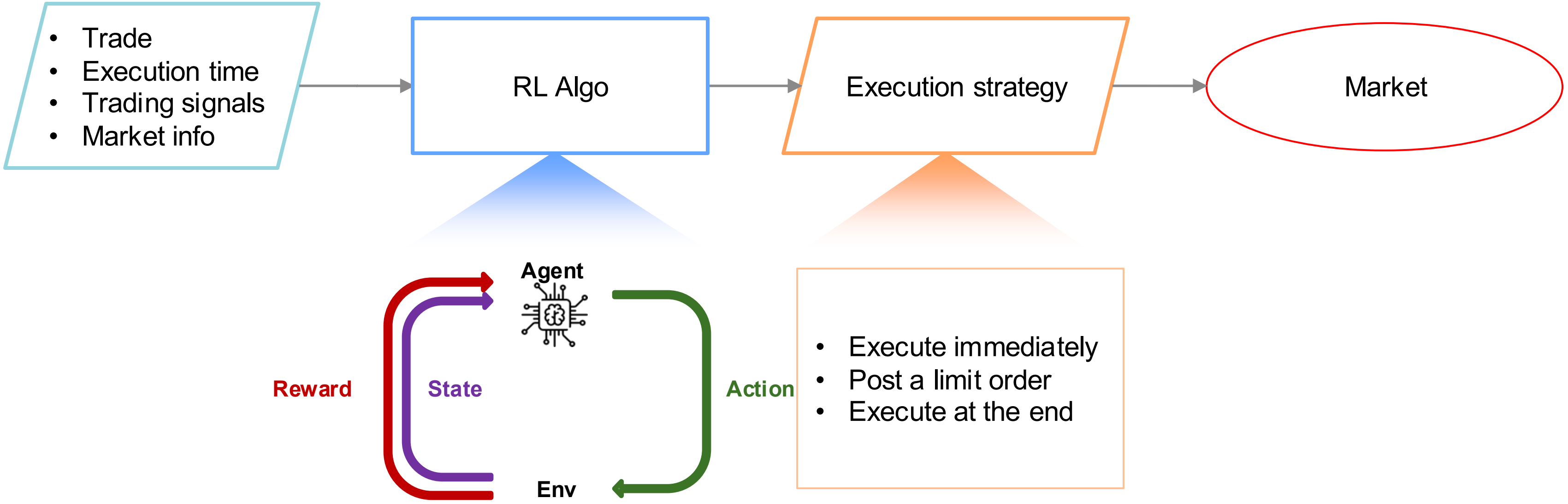
State: trading signals, market information and current position
Action: can be the portfolio position or the trade
Reward: p&l – transaction costs



3 Optimizing Strategy Execution

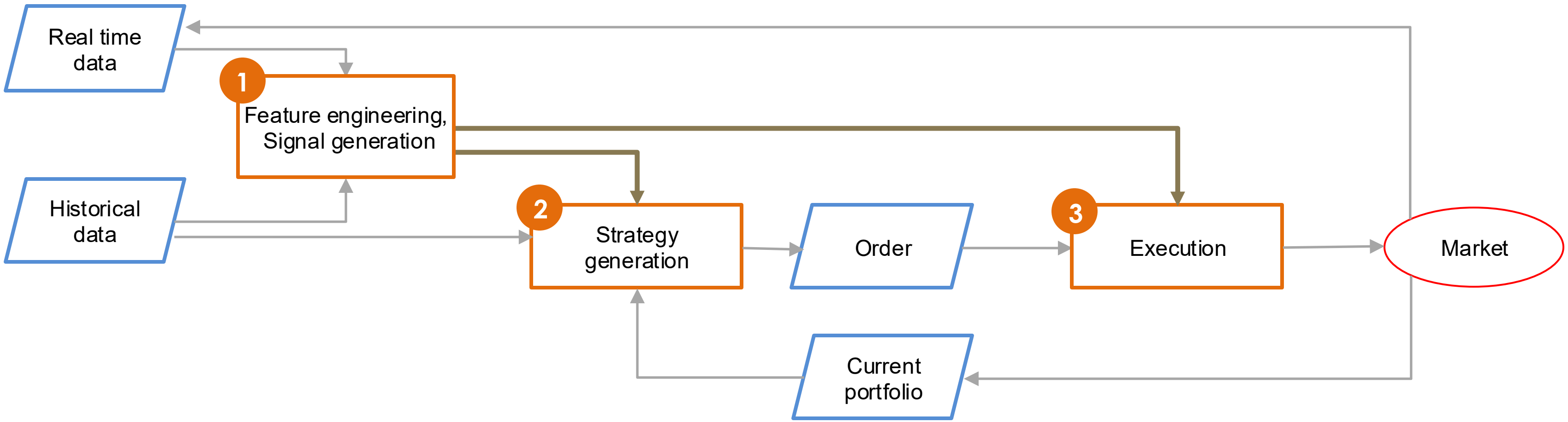
Optimizing execution can reduce transaction costs and market impact

Opportunity cost: execute immediately with high market impact or execute in time with the risk of a market movement?
Trading signal: shorter term compared to the one used in the trading strategy



Intraday Trading with Machine Learning

A generic workflow to generate a trading strategy with any listed asset



Automatic trading algorithms are comprised of 3 main components:

1. The **feature engineering** and **signal generation** component extracts **features** and **alpha signals** from the LOB
2. The **second component** takes the alpha signals and compressed LOB information to generate a **trading strategy**
3. The **execution algorithm** has the objective of minimizing transaction costs and controlling market impact

AGENDA

– Classic systematic strategies
– Learning an FX trading strategy with RL
– Trading futures with ML
– **Cross sectional momentum with LLMs**

Cross-sectional momentum strategy

Choose parameters on training set and analyze P&L on testing set

Strategy description

A Objective

- **Asset**: Single stocks, S&P500 universe
- **Frequency**: monthly
- **Style**: cross sectional, market neutral

B Data

- **Data:** prices, news, macro releases

C Strategy

Use **LLMs** to enhance a cross-sectional market neutral strategy by **including news, articles and macro data**

Standard cross-sectional momentum

- At the end of each month, you calculate **each stock's total return** over the past 6 or 12 months
- You then **rank all stocks in the universe from best to worst performers**
- Go **long** the 10% of top-ranked stocks
- Go **short** the 10% of bottom-ranked stocks
- Construct **dollar neutral** or **beta neutral** portfolio

Leveraging LLMs for equity trading

System design, data requirements, and implementation considerations

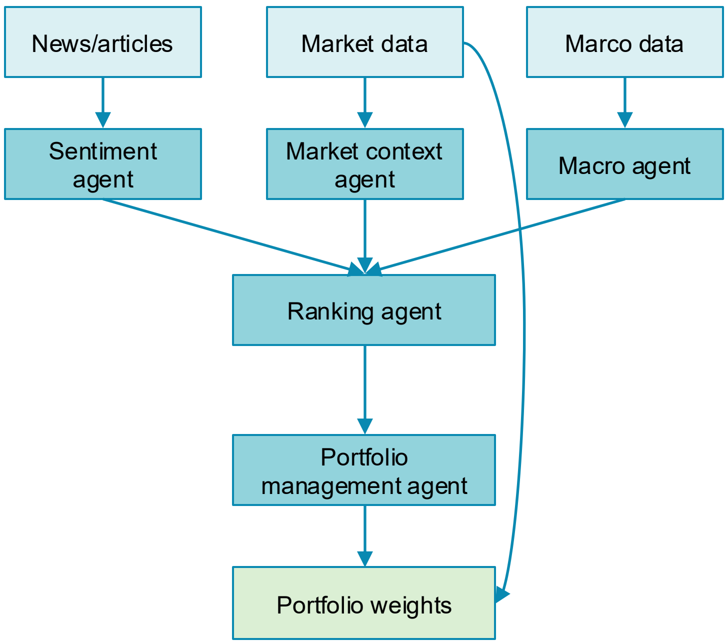
System design choices

- Choice of LLM and context length limitations
- Prompt engineering for financial understanding
- Fine-tuning vs. zero-shot vs. few-shot use
- Avoiding forward-looking bias (LLMs trained beyond back-test window)

Dataset

- **News/articles:** Company news, financial blogs, press releases
- **Market data:** Price history, volume, volatility, technical indicators
- **Macro data:** Interest rates, inflation, GDP, employment stats

Diagram of agent configuration



Q&A

Edoardo Vittori
edov@me.com

References LOB Forecasting

Several research groups are working on short term forecasting from the LOB

- Briola, Antonio, Silvia Bartolucci, and Tomaso Aste. "Deep Limit Order Book Forecasting." *arXiv preprint arXiv:2403.09267* (2024).
- Lucchese, Lorenzo, Mikko S. Pakkanen, and Almut ED Veraart. "The short-term predictability of returns in order book markets: a deep learning perspective." *International Journal of Forecasting* (2024).
- Kolm, Petter N., Jeremy Turiel, and Nicholas Westray. "Deep order flow imbalance: Extracting alpha at multiple horizons from the limit order book." Mathematical Finance 33.4 (2023): 1044-1081.
- Aït-Sahalia, Yacine, et al. *How and When are High-Frequency Stock Returns Predictable?*. No. w30366. National Bureau of Economic Research, 2022.
- Briola, Antonio, et al. "Deep reinforcement learning for active high frequency trading." *arXiv preprint arXiv:2101.07107*(2021).
- Sirignano, Justin, and Rama Cont. "Universal features of price formation in financial markets: perspectives from deep learning." *Machine Learning and AI in Finance*. Routledge, 2021. 5-15.
- Zhang, Zihao, Stefan Zohren, and Stephen Roberts. "Deeplob: Deep convolutional neural networks for limit order books." *IEEE Transactions on Signal Processing* 67.2020
- Cont, Rama, Arseniy Kukanov, and Sasha Stoikov. "The price impact of order book events." *Journal of financial econometrics* 12.1 (2014): 47-88.

References on Expert Learning

- Wang, Yibo, et al. *Non-stationary projection-free online learning with dynamic and adaptive regret guarantees.* Proceedings of the AAAI Conference on Artificial Intelligence. 2024.
- Lu, Zhou, and Elad Hazan. "On the computational efficiency of adaptive and dynamic regret minimization." arXiv. 2022.
- Zhang, Lijun, Shiyin Lu, and Tianbao Yang. *Minimizing dynamic regret and adaptive regret simultaneously." International Conference on Artificial Intelligence and Statistic*. PMLR, 2020.
- Zhang, Lijun, Shiyin Lu, and Zhi-Hua Zhou. *Adaptive online learning in dynamic environments.* Advances in neural information processing systems. 2018.
- Luo, Haipeng, and Robert E. Schapire. *Achieving all with no parameters: Adanormalhedge.* Conference on Learning Theory. PMLR, 2015.
- Li, Bin, and Steven CH Hoi. *Online portfolio selection: A survey.* ACM Computing Surveys. 2014. Cesa-Bianchi, Nicolo, and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.
- Agarwal, A., Hazan, E., Kale, S., and Schapire, R. *Algorithms for portfolio management based on the newton method*. ICML, 2006.
- Zinkevich, Martin. *Online convex programming and generalized infinitesimal gradient ascent*. Proceedings of the 20th international conference on machine learning. 2003.
- Helmbold, David, et al. *Online portfolio selection using multiplicative updates*. Mathematical Finance, 1998.
- Cover, T. and Ordentlich, E. *Universal portfolios with side information*. IEEE Transactions on Information Theory, 1996.

References for LOB forecasting

- Briola, Antonio, Silvia Bartolucci, and Tomaso Aste. "*Deep Limit Order Book Forecasting*." *arXiv.* 2024
- Lucchese, Lorenzo, Mikko S. Pakkanen, and Almut ED Veraart. "*The short-term predictability of returns in order book markets: a deep learning perspective*." International Journal of Forecasting*.* 2024
- Kolm, Petter N., Jeremy Turiel, and Nicholas Westray. "*Deep order flow imbalance: Extracting alpha at multiple horizons from the limit order book*." Mathematical Finance. 2023
- Aït-Sahalia, Yacine, et al. *How and When are High-Frequency Stock Returns Predictable?*. National Bureau of Economic Research. 2022
- Briola, Antonio, et al. "*Deep reinforcement learning for active high frequency trading*." arXiv*. 2021*
- Sirignano, Justin, and Rama Cont. "*Universal features of price formation in financial markets: perspectives from deep learning*." Machine Learning and AI in Finance 2021.
- Zhang, Zihao, Stefan Zohren, and Stephen Roberts. "*Deeplob: Deep convolutional neural networks for limit order books*." *IEEE Transactions on Signal Processing.* 2020
- Cont, Rama, Arseniy Kukanov, and Sasha Stoikov. "*The price impact of order book events*." *Journal of financial econometrics.* 2014

References on trading using RL

- Sun, Shuo, et al. "*TradeMaster: a holistic quantitative trading platform empowered by reinforcement learning*." Advances in Neural Information Processing Systems. 2024
- Sun, Shuo, Rundong Wang, and Bo An. "*Reinforcement learning for quantitative trading*." ACM Transactions on Intelligent Systems and Technology. 2023
- Brini, Alessio, and Daniele Tantari. "*Deep reinforcement trading with predictable returns*." Physica A: Statistical Mechanics and its Applications. 2023
- Riva, Antonio, et al. "*Addressing non-stationarity in FX trading with online model selection of offline rl experts*." ICAIF. 2022.
- Riva, Antonio, et al. "*Learning FX trading strategies with FQI and persistent actions.*" ICAIF 2021.
- Briola, Antonio, et al. "*Deep reinforcement learning for active high frequency trading*." arXiv. 2021
- Théate, Thibaut, and Damien Ernst. "*An application of deep reinforcement learning to algorithmic trading*." Expert Systems with Applications. 2021
- Carapuço, João, Rui Neves, and Nuno Horta. "*Reinforcement learning applied to Forex trading*." Applied Soft Computing. 2018
- Deng, Yue, et al. "*Deep direct reinforcement learning for financial signal representation and trading*." IEEE transactions on neural networks and learning systems. 2016
- Dempster, Michael AH, and Vasco Leemans. "*An automated FX trading system using adaptive reinforcement learning*." Expert systems with applications. 2006

References on optimal execution

- Tommaso Mariotti, Fabrizio Lillo, and Giacomo Toscano. *From zero-intelligence to queue-reactive: limit-order-book modeling for high-frequency volatility estimation and optimal execution*. Quantitative Finance. 2023
- Marina Giacinto, Claudio Tebaldi, and Tai-Ho Wang. 2022. *Optimal order execution under price impact: a hybrid model. Annals of Operations Research*. 2022
- Siyu Lin and Peter A. Beling. *An End-to-End Optimal Trade Execution Framework based on Proximal Policy Optimization.* In IJCAI. 2020
- Michaël Karpe, Jin Fang, Zhongyao Ma, and Chen Wang. 2020. *Multi-agent reinforcement learning in a realistic limit order book market simulation*. ICAIF. 2020
- Brian Ning, Franco Ho Ting Lin, and Sebastian Jaimungal. 2018*. Double deep q-learning for optimal execution*. arXiv. 2018
- Álvaro Cartea, Sebastian Jaimungal, and Jason Ricci. 2018. *Algorithmic Trading, Stochastic Control, and Mutually Exciting Processes*. SIAM. 2018
- Dieter Hendricks and Diane Wilcox. 2014. *A reinforcement learning extension to the Almgren-Chriss framework for optimal trade execution*. IEEE Conference on Computational Intelligence for Financial Engineering & Economics. 2014
- Robert Almgren. *Optimal Trading with Stochastic Liquidity and Volatility*. SIAM Journal on Financial Mathematics 3. 2012