Systematic Trading

Lukas Gonon and Johannes Muhle-Karbe

Imperial College London, June 2024

IMPERIAL

Outline

Quantitative Portfolio Management

1. Introduction
2. Ingredients for Systematic Trading
3. Examples of Trading Strategies
4. Constructing Trading Signals
5. Risk Models
6. Portfolio Construction

1. Introduction

Bio: Lecturers

Johannes Muhle-Karbe (`j.muhle-karbe@imperial.ac.uk`)

- At Imperial College London since 2019.
- Head of the Mathematical Finance Section.
- Director of the CFM-Imperial Institute, a joint venture with large quantitative hedge fund.
- Previously: Carnegie Mellon, Michigan, and ETH Zurich.
- Research on Stochastic Processes, Optimal Control, and applications to Economics and Finance.
- For more details: `Webpage`

1. Introduction

Bio: Lecturers

Lukas Gonon (l.gonon@imperial.ac.uk)

- At Imperial College London since 2022.
- Senior Lecturer (Associate Professor) in the Mathematical Finance Section / Imperial-X.
- Collaborations with banks and hedge funds.
- Research on machine learning and its applications to finance.

- Previously: University of Munich, University of St. Gallen, ETH Zurich.
- For more details: Webpage

IMPERIAL

1. Introduction

Bio: Mentors

Guangyi He (g.he23@imperial.ac.uk)

- At Imperial since 2023, .
- PhD in Mathematical Finance Section, Imperial College London.
- Research on mathematics, finance and machine learning.
- Previously: Master of Mathematics at University of Oxford.
- For more details: LinkedIn

IMPERIAL

1. Introduction

Introductions

- Can you each say a few words about yourself?
- For example:
  - What's your background?
  - Why are you taking this course and what do you expect to learn from it?
- We look forward to getting to know you better while working together this week!
- Important: don't hesitate to ask if you have any questions!

IMPERIAL

1. Introduction

Systematic Investing

- What is this course about?
- How to invest in financial markets?
- Lots of conflicting approaches.
  - Pick the best stock!
  - Just buy and hold the market!
  - Look at past prices and identify trends!
  - Apply fancy machine learning techniques to all kinds of "alternative data"!
- How to make sense of this?

1. Introduction

Efficient Markets?

- Classical finance: *efficient market hypothesis*.
  - Market prices reflect *all* relevant information at all times.
- Consequences:
  - No point in trying to “beat the market”.
  - Makes no sense to pay the high fees charged by active asset managers.
  - Better to invest in a mix of (broadly diversified) passive index funds and (almost) risk-free government bonds.
- But *why* should markets be efficient?

1. Introduction

Efficient Markets?

- Why should markets be efficient?
  - Suppose there was an usually good investment opportunity: high profit for little risk.
  - Then, attentive investors would immediately trade to exploit it until it disappears.
- But now suppose everyone believes markets are fully efficient.
- Then, everyone gives up on finding and exploiting mispricings.
- But who then makes markets efficient in the first place?
  - "Impossibility of efficient markets".

1. Introduction

Inefficient Markets?

- Other extreme view: markets are *inefficient*.
  - (Some) investors are irrational and have systematic biases.
  - Over-/underreaction to news, herding behavior, etc.
- Consequences:
  - Market prices bounce around with little link to fundamentals.
  - Many highly profitable active investment strategies exist.
  - Beating the market should be easy.
- But financial markets are highly competitive.
- Even the most sophisticated investment managers often fail to consistently beat the market.
- Reality lies between *efficient* and *inefficient* markets.

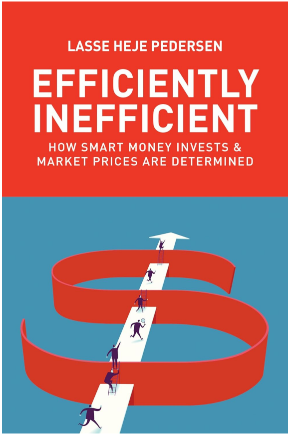
1. Introduction

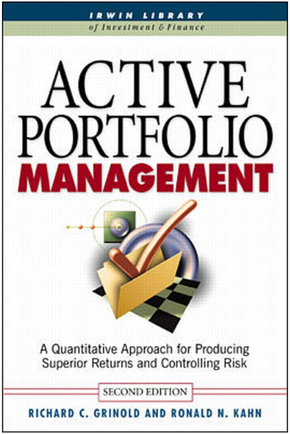
Efficiently Inefficient Markets

- Real markets are "*efficiently inefficient*".
  - Some inefficiencies exist and can be exploited for a profit.
  - Some room for active managers with comparative advantages.
  - But the potential extra returns need to be earned by costly efforts or taking on certain risks.
- In this course, we will discuss some of the basic tools used in this context:
  - What *signals* can be used to trigger trades?
  - How to measure and manage the corresponding *risk*?
  - How to implement the trades when faced with *trading costs*?

1. Introduction

Efficiently Inefficient Markets: Literature





1. Introduction

Comparing Prices?

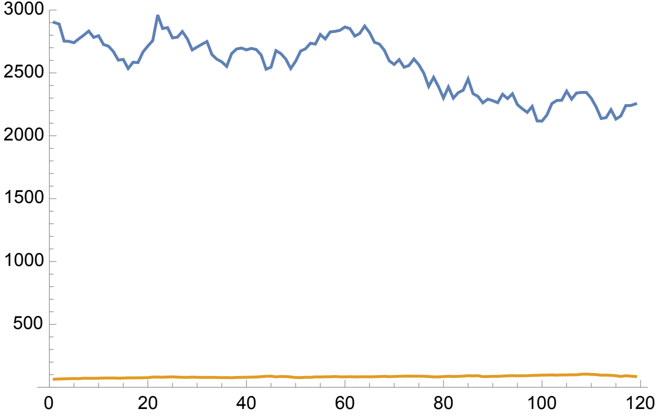
- Asset prices have very different levels:
  - Google June 24, 2022: \$2244.
  - Exxon Mobil, June 24, 2022: \$85.
- Standard deviations of daily price changes also very different.
  - Google 2022: \$230.
  - Exxon Mobil 2022: \$8.
- How to compare across different assets?
- Same problem over long time horizons.
  - S&P 500 today vs. S&P500 50 years ago?
- Ultimately, what matters are not asset prices themselves but their contribution to a *portfolio*!
  - Fewer shares of asset with higher price yield same dollar investment.

IMPERIAL

1. Introduction

Comparing Prices? ct'd

- Prices of Google (blue) and Exxon (orange) in 2022:



- Obviously the wrong scale for a meaningful comparison!

1. Introduction

Comparing Prices? ct'd

- Long S&500 time series:

- Comparing price changes in the '50's to today makes no sense.

1. Introduction

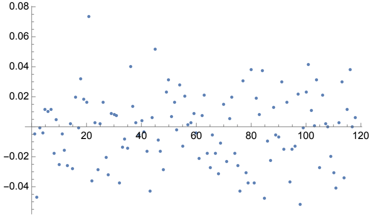
Returns

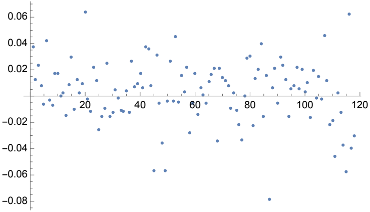
- *Scale-free* measure of investment performance: *return*.
- If the value of an asset or portfolio changes from P_{t-1} to P_t from time $t-1$ to t, this is defined as

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$$

- Amount earned per dollar invested.
- Can be compared across assets and time!
- Let's have a look at this for our previous examples!

- 2022 returns for Google (left) and Exxon (right):



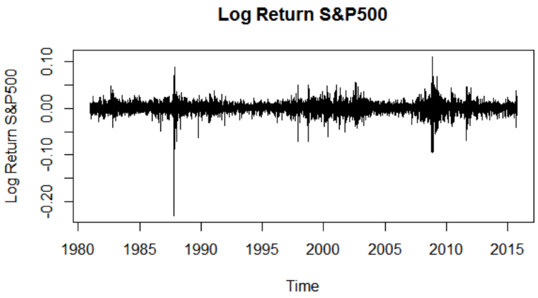


- Standard deviations of returns are very similar, around 2.4%!
- Means are not: -0.2% vs. 0.3%. But similar order of magnitude.

1. Introduction

Comparing Returns ct'd

- S&P500 returns over 35 years:



- Still not independent and identically distributed.
- But comparison over time now makes sense.

1. Introduction

Benchmarking?

- We have seen that returns are a reasonable measure of investment performance.
- But is any investment with a positive return good? No.
  - Even (almost) risk-free bonds pay positive interest rates.
  - Stock prices rise even faster on average.
- Positive returns are not enough.
  - In particular, for actively managed funds that charge investors high fees.
- But *what* is enough?

1. Introduction

Alpha and Beta

- Focus on *excess* returns over the risk-free rate R^f:

$$R_t^e = R_t - R^f$$

- Further decompose excess returns in "*alpha*", "*beta*", and noise:

$$R_t^e = \alpha + \beta R_t^{M,e} + \varepsilon_t$$

  where $R_t^{M,e} = R_t^M - R^f$ is the return of the *market portfolio*.
  - Collection of *all* traded assets.
  - Usually proxied by a broadly diversified index, e.g., MSCI world.
- Parameters computed by running a linear regression of the returns of the strategy against market returns.

1. Introduction

Linear Regression

- Explain variation in a dataset with an explanatory variable.
  - Example: what part of the return of a single stock (or trading strategy) is linked to the market return?
- Inputs:
  - To be explained: $y_1, \dots, y_N$.
  - Explanatory variable: $x_1, \dots, x_N$.
- Simplest causal relationship: linear model

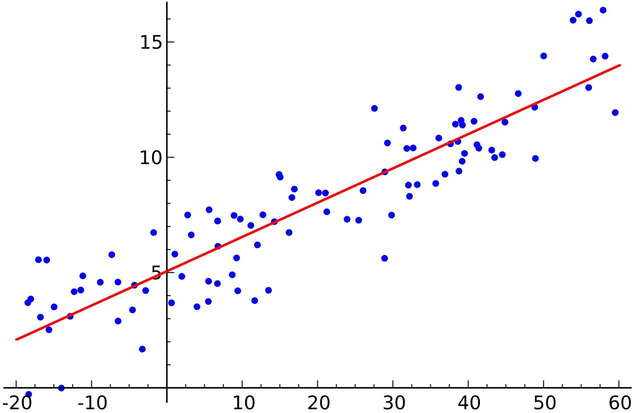
$$y_n = \alpha + \beta x_n, \quad n = 1, \dots, N$$

- Almost never true exactly. Add noise ϵ_n!
  - Hope that noise is reasonably small and "unsystematic".

1. Introduction

Linear Regression ct'd

- Example of a linear regression:



1. Introduction

Linear Regression ct'd

- Linear model

$$y_n = \alpha + \beta x_n + \epsilon_n$$

  - Noise ϵ_n should have mean zero if model is "unbiased".
  - Ideally uncorrelated over time (often not correct).
- How to estimate model parameters α, β?
- Minimize some measure of the fitting errors $y_n - \alpha - \beta x_n$.
- Most tractable: mean-squared error

$$\frac{1}{N}\sum_{n=1}^{N}(y_n - \alpha - \beta x_n)^2$$

"Ordinary Least Squares (OLS) Regression".

1. Introduction

Goodness of Fit?

- How well does the linear model work?
- Predictions of fitted model:

$$\hat{y}_n = \alpha + \beta x_n,$$

Recall: same average as data, $\bar{y} = \frac{1}{N}\sum_n \hat{y}_n = \frac{1}{N}\sum_n y_n$.

- Standard measure for *goodness-of-fit*:

$$R^2 = \frac{\sum_n (\hat{y}_n - \bar{y})^2}{\sum_n (y_n - \bar{y})^2}$$

Fraction of variability in data explained by fitted model.

1. Introduction

Return Decompositions

- Interpretation of return decomposition:

$$R_t^e = \alpha + \beta R_t^{M,e} + \varepsilon_t$$

- β measures sensitivity to market movements.
  - E.g., for $\beta = 0.5$, expect strategy to drop by 5% if market drops by 10%.
  - Exposure to market risk can be bought cheaply through passive index funds – no need to pay high fees if this explains most of the returns!
- Mean-zero noise ε_t describes "idiosyncratic" risks in addition to this "market risk".
  - Such risks can be (hopefully) diversified away by investing into many different assets.
  - In contrast, market risk is a common factor that clearly does not average out.

- Suppose the linear regression is a good approximation.
- Then, market risk can be hedged by shorting β dollars of market exposure for every dollar invested.
- Total excess return is

$$R_t^e - \beta R_t^{M,e} = \alpha + \varepsilon_t$$

- α is the excess return of a *market-neutral* investment.
- This is the "holy grail" of active investing.
  - Extra contribution of skill over just buying market exposure.
- But α and β are only estimates based on historical data..

- How do we know that a positive alpha is not just due to luck?
  - Random shocks ε_t that happened to be positive?
- Regression tools also output an estimate for the standard deviation $\mathrm{Std}(\alpha)$ of the estimate $\bar{\alpha}$.
  - Estimated based on assumptions on "noise", e.g., iid normal.
  - Estimates can be off for correlated, heavy-tailed error terms.
- Standard practice: consider the *t-statistic* $\bar{\alpha}/\mathrm{Std}(\alpha)$.
- Values above 2 are considered as "statistically significant".
  - For normally distributed noise and enough observations, this is approximately the 97.5%-quantile for $(\bar{\alpha} - 0)/\mathrm{Std}(\alpha)$ under the null hypothesis $\alpha = 0$.

1. Introduction

More General Benchmarks

- So far: linear model with market as only systematic factor.
  - Some theoretical justification: CAPM.
- However, various other common factors have also been documented to affect expected returns.
  - Value, Size, Momentum, Liquidity, etc.
  - More details tomorrow!
- Can run the same analysis by forming portfolios that trade on these factors.
- Intercept term then test for outperformance relative to these standard investment "styles".

IMPERIAL

1. Introduction

Risk-Return Tradeoff

- Positive alpha is good and negative alpha is bad.
  - But a larger alpha is not always better
  - Depends on the idiosyncratic risk.
  - Alpha can also be scaled up using “leverage”.
- Conversely, more risk is always bad.
  - But more risk may well be acceptable if it is compensated by sufficiently large expected returns.
- Need to trade off risks and return in a suitable manner!
- This is addressed by *risk-reward ratios*.

1. Introduction

Sharpe Ratio

- Most risk-reward ratios compare a strategy's expected excess return $\mathbb{E}_{t-1}[R_t - R_t^f]$ to the corresponding risk.
- Most well-known example: *Sharpe Ratio (SR)*

$$\mathrm{SR}_{t-1} = \frac{\mathbb{E}_{t-1}[R_t - R_t^f]}{\mathrm{Std}_{t-1}[R_t - R_t^f]}$$

- Nice property: invariant to "leverage".
  - Borrowing at the risk-free rate to make more risky investments (like for a mortgage).
- Why?

- Suppose we start from wealth W.
- Instead of only investing this into our strategy:
  - Borrow an additional amount W at the risk-free rate.
  - Invest $2W$ in the strategy.
- This gives us a an excess return of

$$\frac{2W(1+R_t) - W(1+R_f) - W}{W} - R_f = 2(R_t - R_f)$$

- This has twice the expectation *and* twice the standard deviation of the original excess return $\rightsquigarrow$ same Sharpe ratio!
- Whence, Sharpe ratio does not reward portfolio managers for scaling up their expected returns using leverage.

1. Introduction

Information Ratio

- Leverage invariance is a nice property of the Sharpe ratio.
- However, gives credit for all returns over the risk-free rate.
  - But achieving extra returns by market exposure requires no skill for example.
  - Should be treated differently from idiosyncratic alphas.
- Addressed by *information ratio (IR)*:

$$\mathrm{IR} = \frac{\alpha}{\mathrm{Std}[\varepsilon_t]}$$

- Here, α and the idiosyncratic risk ε come from a regression of the strategy against some benchmark with excess return $R_t^{b,e}$:

$$R_t^e = \alpha + \beta R_t^{b,e} + \varepsilon_t$$

2. Ingredients for Systematic Trading

Trading Signals

- So how can we come up with strategies that outperform benchmarks such a diversified market portfolio?
- First key ingredient: "signals" that predict which assets have high or low *expected* returns.
  - Predicting is a LOT harder than rationalizing ex post!
  - If this works, can buy assets with high expected returns, sell (or ignore) asset with low expected returns.
  - If we can't do this, there is no point in trying to do active trading.
- Key questions ($\rightsquigarrow$ Tuesday and Wednesday)
  - What datasets and what statistical methods to use to create such signals?
  - How to avoid overfitting as well as data mining and ensure robust out-of-sample performance?

2. Ingredients for Systematic Trading

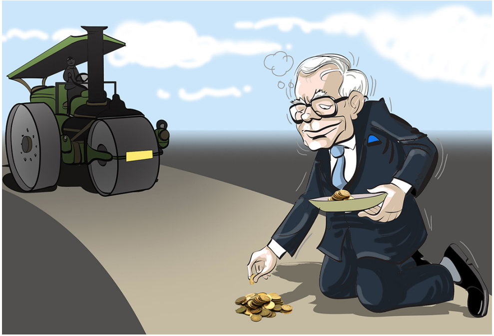
Risk Models

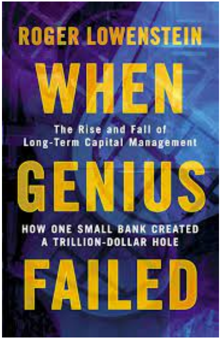
- Suppose we find some good trading signals and somehow combine them to achieve better returns than the market portfolio ($\rightsquigarrow$ Tuesday).
- Does this necessarily lead to a good trading strategy?
- No, depends on the corresponding *risk*.
  - If earning some extra returns requires a gigantic amount of risk, its probably not worth it.
- Key Questions ($\rightsquigarrow$ Thursday):
  - How to measure risk from historical data?
  - How to combine these measurements with trading signals to form (optimal?) portfolios?

2. Ingredients for Systematic Trading

Risk Models ct'd

- *"Picking up nickels in front of the steamroller"..*
- Attributed to an external money manager who warned the heads of Long-Term Capital Management about their high-leverage investment strategies.





- Third ingredient: implementing trades with transaction costs?
  - Strategies that appear profitable in “paper trading” often loose money in real life.
  - One main reason: trading costs such as adverse price impact.
- Frictionless markets: always trade to exploit changing investment opportunities (“market timing”).
- But this is a very bad idea with trading frictions..

2. Ingredients for Systematic Trading

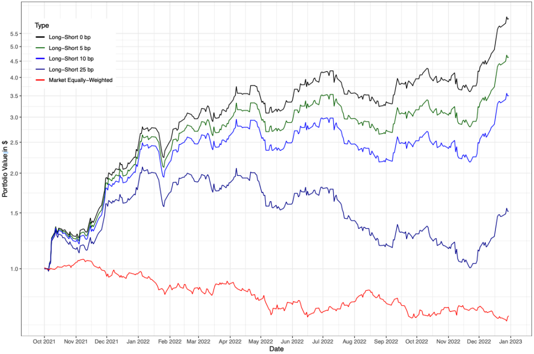
Trading Costs ct'd

TECH

ChatGPT may be able to predict stock movements, finance professor shows

PUBLISHED WED, APR 12 2023·6:24 PM EDT UPDATED THU, APR 13 2023·6:14 PM EDT

Kif Leswing
@KIFLESWING



2. Ingredients for Systematic Trading

Trading Costs ct'd

- So how to take trading costs into account?
- Trade less, and only when it's really worth it.
- But how?
- Need to solve dynamic optimization problem.
- Can either be done using analytical methods (stochastic control theory) or numerically (Reinforcement Learning).
- More on that on Friday!

3. Examples of Trading Strategies

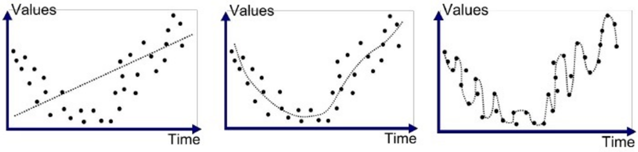
Ingredients for Systematic Trading

- Now: some concrete examples for the ingredients from the previous part:
  - Trading signals.
  - Risk models.
  - Trading costs.
- Let's start with the most crucial ingredient today: trading signals.
- Huge amount of data available – but unlike in other applications (e.g., image recognition) most of it is noise?

3. Examples of Trading Strategies

Overfitting

- When fitting to the same data, more complex models *always* work better!
  - Simpler model is a special case (with some zero parameters).
- Does this mean they are always better?
- No! Pitfall of *overfitting*.
  - Sufficiently complex model can always fit the data perfectly.
  - But may just fit noise rather than causal relationships.



Underfitted | Good Fit/Robust | Overfitted

3. Examples of Trading Strategies

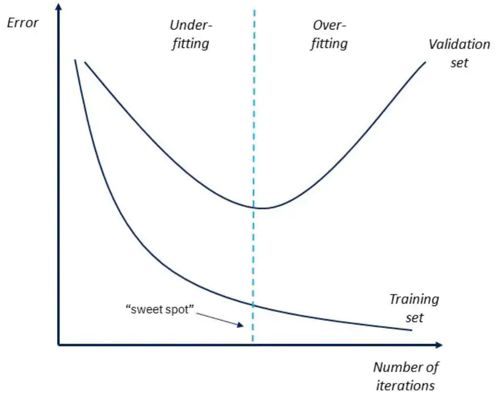
Overfitting ct'd

- What to do about overfitting?
- Always complement *in-sample fitting* with *out-of-sample testing*!
  - More complex models do *not* always work better on data they have not been fitted to.
  - Unless they really do a better job of describing the mechanism underlying the data.
- Split data set into two parts:
  - Training data used to fit the model.
  - Testing data used to validate the model *out of sample*

3. Examples of Trading Strategies

Overfitting ct'd

- Typical: "sweet spot" for model complexity that maximizes out-of-sample performance:



3. Examples of Trading Strategies

Attribution vs. Prediction

- Crucial difference: performance attribution and prediction.
- (Relatively) easy to explain asset returns with *contemporaneous* characteristics.
  - Individual stocks tend to go up when overall market goes up.
- Much harder problem: use characteristics today to predict *future* return.
- Can again try to use linear model:

$$R^e_{t+1} = \alpha + \beta^M R^{M,e}_t + \cdots + \varepsilon_t$$

- Parameters can again be estimated using linear regression, but R^2 values are generally MUCH lower.
  - If we know market went up today, this tells us very little about whether individual stocks will go up or down tomorrow.

3. Examples of Trading Strategies

Return Prediction

- There is a vast amount of (academic and industrial) research on return prediction.
- Also uses many asset characteristics other than market and accounting variables:
  - Index membership, industry membership, liquidity, option implied volatilities, bond-yield spreads, "ESG", etc.
- Linear models common due to tractability ($\rightsquigarrow$ Wednesday).
- Lots of recent work on nonlinear machine learning models.
  - E.g., fit to historical data for the entire limit order book of many stocks ($\rightsquigarrow$ Thursday).
- Major challenge in each case: overfitting and data mining.
- One way out: use theory to use your data budget wisely.

3. Examples of Trading Strategies

Equity Valuation and Investing

- Most basic investment style: “value”
- Nicely summarized by Warren Buffet:

  > *Intrinsic value is an all-important concept that offers the only logical approach to evaluating the relative attractiveness of investments and businesses. Intrinsic value can be defined simply: it is the discounted value of the cash that can be taken out of a business during its remaining life*.

- Here, the term *intrinsic value* (or *fundamental value*) distinguishes from the market price of the stock.
  - Should coincide in fully-efficient markets.
  - In contrast, “value investors” search for stocks whose market price they deem cheap relative to the intrinsic value.

3. Examples of Trading Strategies

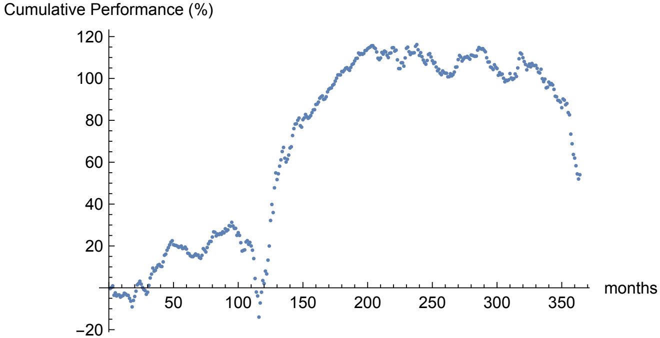
Equity Valuation ct'd

- How to estimate fundamental value?
- One approach: "dividend discount model".
  - Forecast future cash flows generated such as dividends.
  - Discount these to account for "time value of money" and risk.
  - Both steps aren't easy.
- Alternative: compare accounting data to market prices.
  - Standard example: book value vs. market value.

3. Examples of Trading Strategies

Value ct'd

- One typical implementation: buy stocks with a high book value compared to their market value.
- This simple strategy was profitable historically, but has done very poorly in recent years:



3. Examples of Trading Strategies

Value ct'd

- *Value investing*: buy securities that appear cheap, possible short expensive securities (relative to intrinsic values).
- Sounds simple. But is not straightforward in practice, since it means going against the conventional wisdom.
  - Assets are often expensive for a reason.
  - There is something other investors love about expensive stocks, and that makes them uneasy about cheap ones.
- Investors only focusing on value risk owning fundamentally flawed stocks (the "value trap").
- One way mitigate this: also consider *quality* characteristics.

  Whether we are talking about socks or stocks, I like buying quality merchandise when it is market down (Warren Buffet).

3. Examples of Trading Strategies

Momentum

- Next, let us consider "trend-following"/"momentum":
  - Buy after prices have gone up, sell after prices have gone down.
  - No additional accounting data; just price history.
  - Typical trading indicator is *moving average* of past stock returns:

    $$MA_t^N = \frac{1}{N}\sum_{s=t-N}^{t} R_s^e$$

    where N is the "lookback period".
  - Sometimes, people also use *exponentially weighted moving averages*, which discount older values at rate $\delta \in (0,1)$:

    $$EMA_t^N = \frac{1}{N}\sum_{s=t-N}^{t} (1-\delta)^{t-s} R_s^e$$

3. Examples of Trading Strategies

Momentum ct'd

- Efficient market hypothesis: information from past returns is already "priced in".
  - Not useful as extra trading signal.
- But markets are not fully efficient.
- Various forms of "momentum" are one of the most commonly used strategies to try to exploit this.
- Underlying economic mechanism?
  - Initial underreaction to news.
  - Eventual overreaction.
  - Finally corrected.
- How to formalize and test this?

3. Examples of Trading Strategies

Momentum ct'd

- Simple linear prediction model:

$$R_t^e = \alpha + \beta MA_{t-1}^N + \epsilon_t$$

- ϵ_t is unpredictable noise, MA_{t-1}^N is moving average observed when making the prediction.
- α, β (and variance σ_ϵ^2 of the noise) can be estimated by regressing realized returns against their moving averages.
- In contrast, lookback period N is a "hyperparameter".
  - Must be chosen somewhat arbitrary.
  - Have to be careful about explicit and implicit data mining.
  - Same comments apply to discount rate δ of EMAs.

3. Examples of Trading Strategies

Momentum ct'd

- To incorporate more general trends: general linear model

$$R_t^e = \alpha + \sum_N \beta_N MA_{t-1}^N + \epsilon_t$$

  - Different lookback periods N, e.g., one week, one month, one quarter, one year.
- Parameters can still be estimated from linear regression against several predictors.
  - If factors are too correlated, this can lead to instabilities.
  - Can be addressed using more sophisticated techniques like generalized least squares.
- More parameters – always improves in-sample performance!
- Need to test out of sample to make sure gains are robust!

3. Examples of Trading Strategies

Cross Sectional Momentum

- So far: "time-series momentum".
  - Exploit trends in the time series of a single asset.
  - Then average results over many stocks in a second step to obtain more stable results.
- Alternative: cross sectional momentum.
  - Sort stocks by their performance in lookback window.
  - "Buy winners and sell losers".
  - Simplest implementation: equal weighted portfolio of 10% of most successful stocks, financed by shorting equal weighted portfolio of worst performers.
  - Long-short portoflio hedges out market risk.
  - Averaging over several stocks leads to more stable performance.
  - See Jegadeesh and Titman (1993) for more details.

3. Examples of Trading Strategies

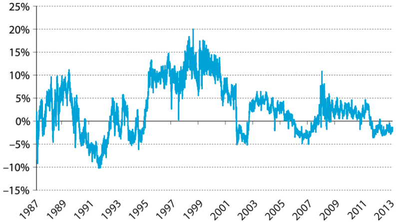
Cross Sectional Momentum ct'd

- To implement this:
  - Get returns for large set of stocks, say monthly returns for S&P500.
  - Each month, sort returns from best to worse.
  - Choose amount of money you want to invest, say $1MM.
  - Invest an equal percentage of 2% in each of the 50 best-performing stocks, i.e., $20k.
  - Finance this by shortselling the same amount of each of the 50 worst-peforming stocks.
  - Hold until the next month, then evaluate peformance.
  - Repeat the above procedure and rebalance to the new portfolio.

3. Examples of Trading Strategies

Statistical Arbitrage

- Another important class of trading strategies: based on mean-reversion.
  - "Pairs trading" and "statistical arbitrage".
- Classical example: "dual-listed stocks":
  - Shares of two companies that have merged but remain listed separately on different exchanges (e.g., Unilever)
  - Prices follow each other closely, but there is sometimes a significant spread between them.

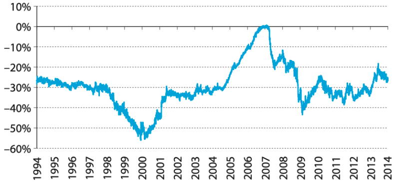


IMPERIAL

3. Examples of Trading Strategies

Statistical Arbitrage ct'd

- Stat arb traders trade on the discrepancies between such "twin stocks".
  - Reduces the arbitrage spreads, but often does not eliminate them completely in efficiently inefficient markets.
  - Often, the less liquid of the twins trades at a discount.
- Another well-known stat arb: different share classes for the same company, e.g., preference and ordinary shares of BMW:



3. Examples of Trading Strategies

Statistical Arbitrage ct'd

- Third classical stat arb: combinations of assets that behave similarly in a statistical sense.
- No explicit arbitrage link.
- Classical example is "pairs trading".
  - Constructs long-short portfolio of similar stocks.
  - Then bet on its mean-reversion, i.e., price reversals.
- Same approach is also applied to long-short portfolios of other assets.
  - E.g., exchange rates, bonds with different maturities, spreads between interest rates for different currencies.

3. Examples of Trading Strategies

Liquidity

- As a another example, let's now discuss another strategy that focuses on the "cross section" of different stocks.
- Like for cross sectional momentum: buy stocks with high expected returns, sell stocks with low expected returns.
  - For market timing: buy during times with high expected returns, sell in times with low expected returns.
- One prominent class of strategies focuses on "liquidity".
  - That is, how easy it is to trade different assets.

3. Examples of Trading Strategies

Liquidity ct'd

- Idea: if an asset is difficult to trade, investors will only hold it if it earns a higher expected return.
- With a long time horizon, it should therefore be profitable to hold a portfolio tilted towards illiquid investments.
- But:
  - Difficult to liquidate in times of crisis.
  - Illiquid assets are typically more risky and have higher transaction costs.
- How to formalize this?
- First need measure of "(il)liquidity"!

- One popular measure (Amihud, 2002):

$$ILLIQ_t = \frac{1}{N} \sum_{s=t-N}^{t} \frac{|R_s|}{Volume_s}$$

- Here, $Volume_s$ is the trading volume at time s. Often measured in dollar amount traded.
- Idea: market is illiquid if market moves a lot in response to little volume.
- Advantage: easy to compute for long time series of publicly available data.
- Can be used to compare liquidity of different stocks!

3. Examples of Trading Strategies

Liquidity ct'd

- How to build a trading strategy based on this?
- Compute *ILLIQ* for a set of stocks.
  - E.g., all stocks in the S&P500 or even all stocks traded on NYSE, for example.
- Sort stocks by liquidity.
- Buy stocks with low liquidity (high *ILLIQ*), shortsell (or avoid) stocks with high liquidity (low *ILLIQ*).
- Many different ways to choose portfolio weights.
  - E.g., equal weighted between top and bottom ten percent.
  - Or equal risk contributions, mean-variance optimization, etc.
  - Long-short portfolio reduces exposure to market risk, but cannot be implemented by everyone.

3. Examples of Trading Strategies

Liquidity ct'd

- In addition to *market liquidity risk*, hedge funds also take on *funding liquidity risk*.
  - Risk that they cannot fund a leveraged position throughout the lifetime of a trade.
- How to leverage in reality? (Similar for shortselling.)
- Suppose you want to buy 1 million bonds at $100 each.
- Like when buying a house, you can try to use the bonds as *collateral* to obtain a loan.
- To protect themselves against falling prices, lenders ask for a "haircut" (or "margin requirement"), e.g., $10 per bond.
  - Like a down payment for a house.
- So in this example you need $10 million for the margin requirement to buy $100 million worth of bonds.
- Whence, your maximal leverage is 10.

3. Examples of Trading Strategies

Liquidity ct'd

- Margin requirements are typically set to cover some "worst-case" price move with a certain confidence.
- Updated dynamically: the positions of a hedge fund are "marked-to-market" each day using the current market prices.
  - Changes are credited or debited to margin account.
- If there is insufficient cash in the margin account, the the hedge fund receives a *margin call* from its broker.
  - Either cash needs to be added to the account, or positions need to be reduced.
  - If the fund does not do one or the other, the broker will liquidate the positions.
- Whence, for leverage or shortselling, you need to continuously monitor your positions and your margin accounts.

3. Examples of Trading Strategies

Liquidity ct'd

- How are asset prices affected by how easy or difficult it is to fund a security?
- If investment in an asset is difficult and expensive to fund, then it is natural to expect that investors demand compensation for holding such "cash-intensive" securities.
- Therefore, required returns should increase with the margin requirement.

3. Examples of Trading Strategies

Liquidity ct'd

- Another implication of funding and leverage constraints:
  - Many investors prefer to buy risky securities over leveraging safer alternatives.
  - This helps to explain why riskier securities tend to offer lower risk-adjusted returns than safer ones within in each asset.
  - Put differently, a portfolio of risky stocks tends to underperform a leveraged portfolio of safer stocks.
  - This underlies "risk-parity investment", where one does not invest an equal *fraction of wealth* in each asset, but instead equalizes *risk* across different investments.
- Illiquid assets with high transaction costs tend to also be difficult to finance, and vice versa.

3. Examples of Trading Strategies

Liquidity ct'd

- Holding illiquid securities is one way to earn market and funding liquidity premia.
- Another is *market making* (or *liquidity provision*).
  - Many investors want to trade immediately.
  - But buyers and sellers are not always present in the market at the same time.
  - Market makers (*liquidity providers*) step in and smooth out supply-demand imbalances.
  - Earn higher "ask" prices for buying than the "bid" prices they pay for selling the same securities.
- Risks faced by market makers:
  - Inventory risk.
  - Adverse selection.

3. Examples of Trading Strategies

Size

- Another characteristic of the cross section of assets: *size*.
- Can be measured by market capitalization of each stock relative to the entire reference market.
  - E.g., percentage contributions μ_n of each stock n to total market capitalization of S&P500 or Russel 3000.
- Can measure "diversity" of overall market in various ways, e.g., using

$$\mathbf{D}_p(\mu_1, \ldots, \mu_N) = \left(\sum_{n=1}^{N} \mu_n^p \right)^{1/p}, \quad 0 < p < 1$$

(maximal for $\mu_n = 1/N$ for all $n = 1, \ldots, N$)

3. Examples of Trading Strategies

Size ct'd

- Diversity is mean-reverting!
- If measured by $\mathbf{D}_p$ with $p = 1/2$:

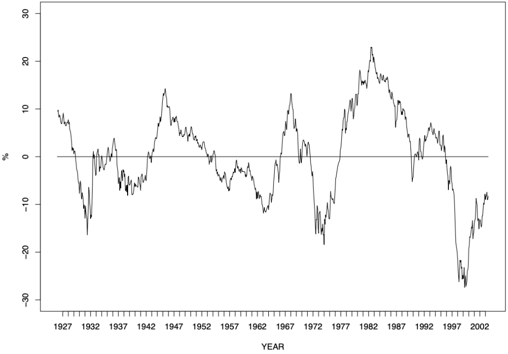


Figure 2: Cumulative change in market diversity, 1927–2004.

3. Examples of Trading Strategies

Size ct'd

- Long-term stability is only possible if smaller stocks have higher returns than larger stocks.
- Can build long-term trading strategies based on this similarly as for liquidity.
  - Characteristics of course have some similarity: smaller stocks tends to be less liquid and vice versa.
  - Smaller stocks also tend to be more risk and pay less dividends.
- Example: diversity weighted portfolio from stochastic portfolio theory with portfolio weights

$$\pi_n = \frac{\mu_n^p}{\mathbf{D}_p(\mu_1, \ldots, \mu_N)^p}$$

Bigger weights for small stocks, normalized to add up to one.

4. Constructing Trading Signals

Forecasting Using Linear Regression

- We now discuss how to construct "trading signals", i.e., forecasts of future price changes.
- Basic idea: suppose variable to be predicted is a function of explanatory variables plus noise, e.g.,

$$R_{t+1} = f(R_t, R_{t-1}, \ldots) + \epsilon_{t+1}$$

- Then try to estimate the function f by minimizing fitting error on the training part of your data.
- Finally assess the out-of-sample performance of your model on the test part of your data set.
- Simplest model class: linear function $f \rightsquigarrow$ linear regression ($\rightsquigarrow$ practical session today).

4. Constructing Trading Signals

Forecasting Using Deep Learning

- Linear regression: baseline approach for forecasting and in turn for constructing trading signals.
- Alternative more sophisticated tools accounting for non-linearity?
- Now: introduction to *deep learning* for forecasting.
  - Introduction to deep learning
  - Hands-on example 1: price prediction based on limit order book data (*high-frequency data*)
  - Hands-on example 2: momentum strategy using deep learning predictors on daily returns (*low-frequency data*)

4. Constructing Trading Signals

Classical Applications of Deep Learning

Well-established breakthroughs in:

- image recognition
- speech recognition and synthesis
- optimal decision making
- synthetic image generation
- ...

4. Constructing Trading Signals

Example: Alpha Go

ARTICLE

doi:10.1038/nature16961

Mastering the game of Go with deep neural networks and tree search

David Silver[1*], Aja Huang[1*], Chris J. Maddison[1], Arthur Guez[1], Laurent Sifre[1], George van den Driessche[1], Julian Schrittwieser[1], Ioannis Antonoglou[1], Veda Panneershelvam[1], Marc Lanctot[1], Sander Dieleman[1], Dominik Grewe[1], John Nham[2], Nal Kalchbrenner[1], Ilya Sutskever[2], Timothy Lillicrap[1], Madeleine Leach[1], Koray Kavukcuoglu[1], Thore Graepel[1] & Demis Hassabis[1]

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the difficulty of evaluating board positions and moves. Here we introduce a new approach to computer Go that uses 'value networks' to evaluate board positions and 'policy networks' to select moves. These deep neural networks are trained by a novel combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without any lookahead search, the neural networks play Go at the level of state-of-the-art Monte Carlo tree search programs that simulate thousands of random games of self-play. We also introduce a new search algorithm that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, our program AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0. This is the first time that a computer program has defeated a human professional player in the full-sized game of Go, a feat previously thought to be at least a decade away.

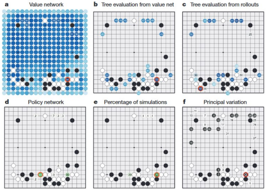


Figure 5 | How AlphaGo (black, to play) selected its move in an informal game against Fan Hui. For each of the following statistics, the location of the maximum value is indicated by an orange circle. **a**, Evaluation of all successors s' of the root position s, using the value

d, Move probabilities directly from the SL policy network, $p_\sigma(a|s)$; reported as a percentage (if above 0.1%). **e**, Percentage frequency with which actions were selected from the root during simulations. **f**, The principal variation (path with maximum visit count) from AlphaGo's

4. Constructing Trading Signals

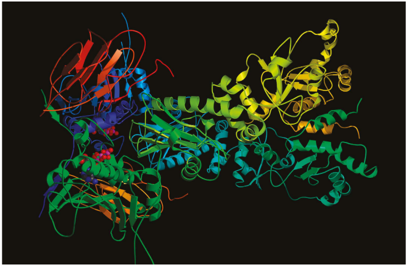
Example: AlphaFold

nature

NEWS | 30 November 2020

'It will change everything': DeepMind's AI makes gigantic leap in solving protein structures

Google's deep-learning program for determining the 3D shapes of proteins stands to transform biology, say scientists.



4. Constructing Trading Signals

Example: Synthetic image generation using StyleGAN2

Artificial cat images

IMPERIAL

Key applications in finance?

4. Constructing Trading Signals

Deep learning in finance

- Financial data is typically noisy and scarce $\rightarrow$ deep learning needs to be used with care!
- Not suitable for all types of problems.
- Some successful applications:
  - Derivative pricing
  - Model calibration
  - Hedging
  - Market scenario generation
  - *Portfolio optimization*

4. Constructing Trading Signals

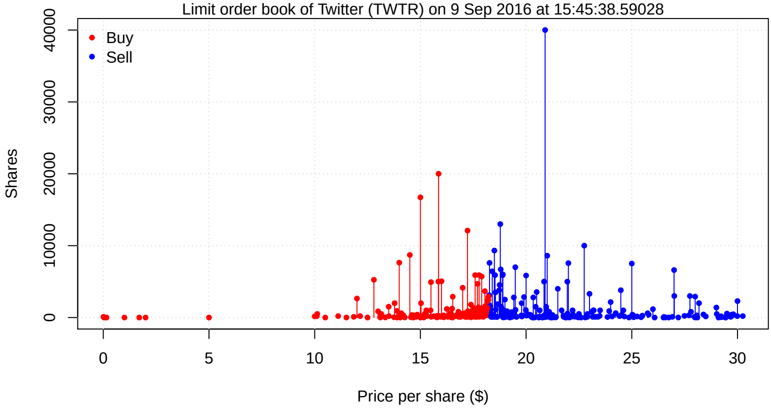
Deep learning in finance

Some important recent papers:

- Kolm, Turiel & Westray: Deep Order Flow Imbalance: Extracting Alpha at Multiple Horizons from the Limit Order Book
- Zhang, Zhoren & Roberts: Deep Learning for Portfolio Optimisation
- Gu, Kelly & Xiu: Empirical Asset Pricing via Machine Learning
- Horvath, Muguruza & Tomas: Deep learning volatility: a deep neural network perspective on pricing and calibration in (rough) volatility models
- Wiese, Knobloch, Korn & Kretschmer: Quant GANs: deep generation of financial time series
- Buehler, G., Teichmann, Wood: Deep hedging
- ...

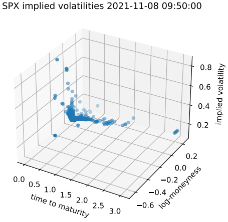
4. Constructing Trading Signals

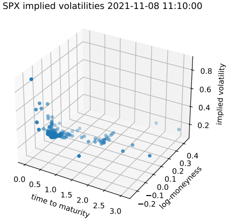
Example: trading signals from limit order book data

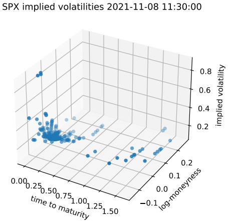


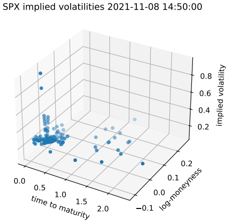
4. Constructing Trading Signals

Example: option pricing (inter/extrapolation)









IMPERIAL

4. Constructing Trading Signals

Example: options hedging

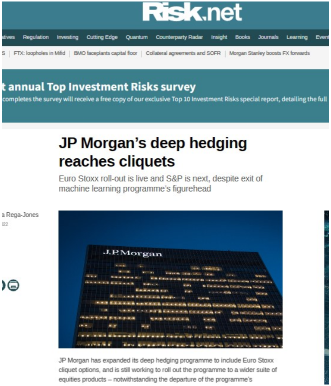
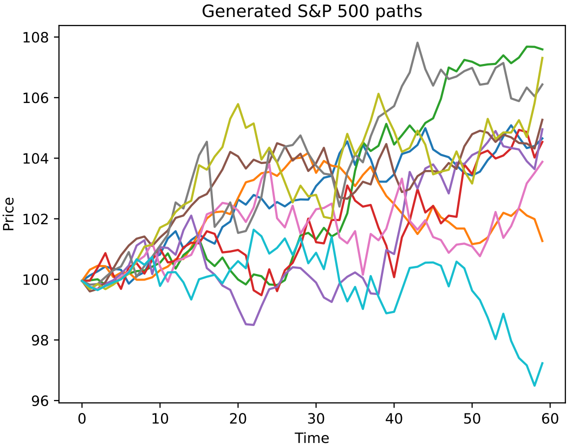


Figure 1: Risk.net article from May 2022

4. Constructing Trading Signals

Example: artificial financial time series generation



4. Constructing Trading Signals

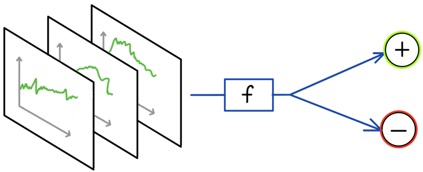
Basic setting

Goal: design an "optimal" input-output map $f : \mathbb{R}^d \to \mathbb{R}^m$ that turns d-dimensional *inputs*

$$x = (x_1, \ldots, x_d)$$

into m-dimensional *outputs*

$$f_1(x_1, \ldots, x_d), \ldots, f_m(x_1, \ldots, x_d)$$



4. Constructing Trading Signals

Simple classification example

- Input dimension: $d = 512 \cdot 512$ (# pixels)
- Output dimension: $m = 1$
- Goal: design $f : \mathbb{R}^d \to [0,1]$ such that $f(x) \hat{=}$ probability that picture x contains a cat.

$\xrightarrow{f} 1$

$\xrightarrow{f} 0$

$\xrightarrow{f} 0$

4. Constructing Trading Signals

Basic approach

Designing such an f using deep learning:

- Choose $f = f_\theta$ a *deep neural network* with parameters θ.
- Collect a large number of training data points of *input/output pairs* $(x^1, y^1), \ldots, (x^n, y^n)$.
- In previous example:

  (, 1), (, 0), (, 1), (, 0), $\ldots$

- Find θ such that $f_\theta(x^i) \approx y^i \to$ choose a *loss function* ℓ and minimize

$$\frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(x^i), y^i) \quad \text{over } \theta$$

- Example: Binary cross-entropy

$$\ell(\hat{y}, y) = -y \log \hat{y} - (1-y)\log(1-\hat{y})$$

4. Constructing Trading Signals

Examples of loss functions

- Common choice in classification context: binary cross-entropy

$$\ell(\hat{y}, y) = -y \log \hat{y} - (1-y)\log(1-\hat{y}).$$

  - For labels with $y = 0$: $\ell(\hat{y}, 0) = -\log(1-\hat{y})$. In this case, $\hat{y}$ closer to 0 leads to a smaller loss. $\hat{y}$ closer to 1 leads to higher loss:

$$\lim_{\hat{y}\uparrow 1} \ell(\hat{y}, 0) = \lim_{\hat{y}\uparrow 1} -\log(1-\hat{y}) = \infty$$

  - For labels with $y = 1$: $\ell(\hat{y}, 1) = -\log(\hat{y})$. In this case, $\hat{y}$ closer to 1 leads to a smaller loss. $\hat{y}$ closer to 0 leads to higher loss:

$$\lim_{\hat{y}\downarrow 0} \ell(\hat{y}, 1) = \lim_{\hat{y}\downarrow 0} -\log(\hat{y}) = \infty$$

- In other contexts (regression): $\ell(\hat{y}, y) = |\hat{y} - y|^2$

Definition

Consider

- $d \in \mathbb{N}$: dimension of input layer
- $L \in \mathbb{N}$: number of layers
- $\varrho_1, \ldots, \varrho_L : \mathbb{R} \to \mathbb{R}$: (non-linear) functions called *activation functions*
- $T_i : \mathbb{R}^{N_{i-1}} \to \mathbb{R}^{N_i}$, $i = 1, \ldots, L$, where $T_i(x) = W^{(i)}x + b^{(i)}$ for matrices $W^{(i)}$ and vectors $b^{(i)}$ called *weights and biases*.

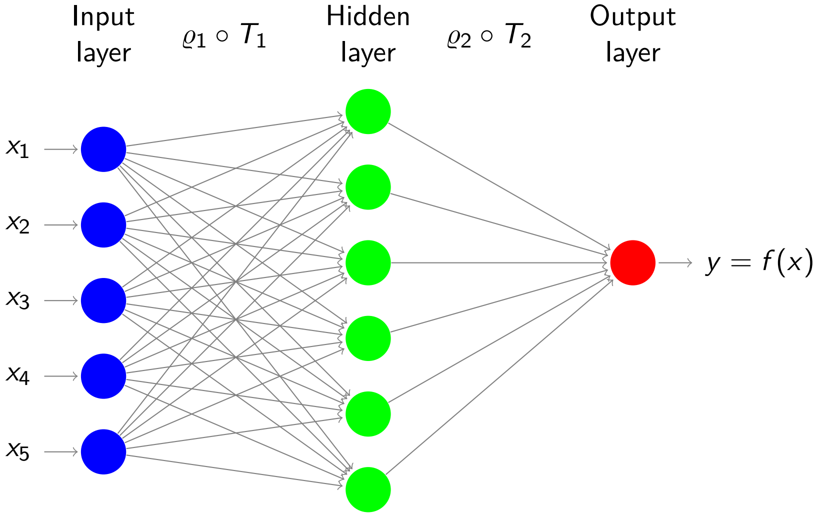
Then $f : \mathbb{R}^d \to \mathbb{R}^{N_L}$ given by

$$f(x) = (\varrho_L \circ T_L) \circ \cdots \circ (\varrho_1 \circ T_1)(x), \quad x \in \mathbb{R}^d$$

is called a *(deep) neural network*.

4. Constructing Trading Signals

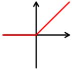
Graphical representation of an example



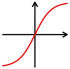
4. Constructing Trading Signals

Examples of activation functions

ReLU (rectified linear unit): $\varrho(x) = \max(x, 0)$



$$\varrho(x) = \tanh(x)$$



Why neural networks?

- Traditionally:
  - Inspired by biological neural networks
  - Universal approximation property

- View today:
  - Hardware, large datasets, efficient algorithms & implementation
  - Expressive power

IMPERIAL

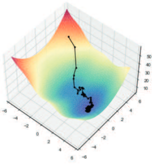
4. Constructing Trading Signals

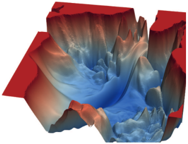
Basic approach in a bit more detail

- Select a *network architecture*: L, $\varrho_1, \dots, \varrho_L$, $N_1, \dots, N_L$.
- Write $\theta = (W^{(1)}, \dots, W^{(L)}, b^{(1)}, \dots, b^{(L)})$ for the *trainable parameters* and f_θ for neural network with parameters θ.
- For training data $(x^1, y^1), \dots (x^n, y^n)$ and a loss function ℓ let

$$\mathcal{L}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(f_\theta(x^i), y^i).$$

- Find an (approximate) minimizer θ^* of $\mathcal{L}$ (using a stochastic gradient descent algorithms and backpropoagation).
- Trained neural network f_{θ^*} can be used for predictions.

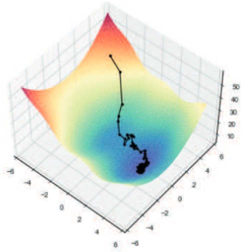




4. Constructing Trading Signals

Gradient descent

- What is gradient descent?
  - Start with initial guess for neural network parameter
  - Compute “best direction” of change (gradient $\nabla \mathcal{L}$) for these parameters and update parameters accordingly
  - Repeat until you reach the minimum
- For neural networks:
  - Gradient can be computed using the “chain rule” from calculus – the backpropagation algorithm allows to do this efficiently
  - Much more complex loss $\rightarrow$ need stochastic gradient descent

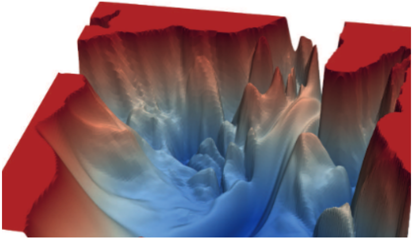


IMPERIAL

4. Constructing Trading Signals

Stochastic gradient descent

- Why stochastic gradient descent?
  - For neural networks: the loss is much more complex than in previous pictures $\rightarrow$ there is a risk of getting stuck in a "local" minimum of the loss.
  - Evaluating the loss on *all* data points at once is computationally infeasible.
- What is the difference to gradient descent?
  - In each step we randomize the descent direction by computing the loss only based on $n_{\mathrm{batch}} << n$ data points.
  - Helpful to avoid "traps" & optimization is feasible for large datasets



IMPERIAL

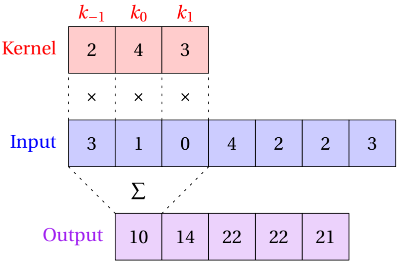
4. Constructing Trading Signals

Advanced architectures

- Basic approach is underlying also more complex applications.
  - Reinforcement Learning
  - Transformers
  - Large Language Models
- Next: overview on some important advanced architectures.

4. Constructing Trading Signals

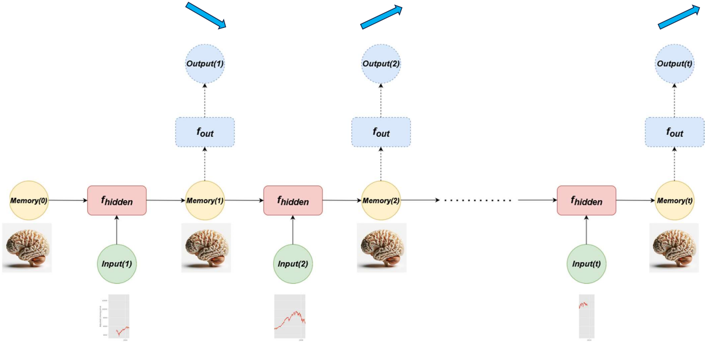
Convolutional neural networks



- Allow to aggregate adjacent input data.
- Can be useful for computing "generalized" moving averages of time series data. The time horizon is selected directly as part of the neural network training.
- Very commonly used for other types of data: e.g. on image data it allows to aggregate pixels next to each other.

4. Constructing Trading Signals

Recurrent neural networks



- Recurrent neural networks capture the history of all inputs in a “memory”. This allows to process sequences of different length with the same network.
- Frequently used for time series prediction ($\rightsquigarrow$ Coding session)
- Various variants (SimpleRNNs, LSTMs) exist.

IMPERIAL

4. Constructing Trading Signals

Generative adversarial networks

- Generator network $\boldsymbol{g}$: returns for randomly generated "pure noise" input $\boldsymbol{z}$ a synthetic data point $\boldsymbol{g}(\boldsymbol{z})$.
- For any (real or synthetic) data point $\boldsymbol{x}$, the discriminator network $\boldsymbol{d}$ returns $\boldsymbol{d}(\boldsymbol{x}) \in (0,1)$, the probability that $\boldsymbol{x}$ is a real data point.
- Loss function:

$$\mathcal{L}(\boldsymbol{g}, \boldsymbol{d}) = \frac{1}{N_1} \sum_{i=1}^{N_1} \log(\boldsymbol{d}(\boldsymbol{x}^i)) + \frac{1}{N_2} \sum_{i=1}^{N_2} \log(1 - \boldsymbol{d}(\boldsymbol{g}(\boldsymbol{z}^i))),$$

  where $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^{N_1}$ are real data points and $\boldsymbol{z}^1, \ldots, \boldsymbol{z}^{N_2}$ are randomly sampled inputs.
- $\log$ and $\log(1 - \cdot)$ penalize predicted probabilities close to 0 and 1, respectively.
- Discriminator: aims to *maximize* $\mathcal{L}$ w.r.t. $\boldsymbol{d}$.
- Generator: aims to *minimize* $\mathcal{L}$ w.r.t. $\boldsymbol{g}$.
- Variants of these methods: commonly used for synthetic data generation.

Next: coding example (high-frequency)

`DL Limit Order Book Classification.ipynb`

- Goal: predict next price move based on current state of limit order book.
- Rationale: could be used as trading signal.
- Along the way, we learn
  - how to implement and train a neural network,
  - how to address *overfitting*.

Next: coding example (low-frequency)

`DL Predicting Returns.ipynb`

- Goal: use deep learning to predict future returns based on daily stock price data.
- Build a momentum strategy based on this trading signal.
- Along the way, we learn
  - how to implement and train a recurrent neural network,
  - how to address *overfitting*.

5. Risk Models

Risk Measurement

- So far, we have focused on evaluating and predicting the expected returns of trading strategies.
  - Need to be significantly positive (robustly out of sample) relevant to appropriate benchmark.
  - Otherwise, there is no point to trade the strategy.
- But even if a strategy consistently delivers positive excess returns, then the strategy still may not be all that attractive.
  - Return may just be a little bit higher than for alternatives, but corresponding *risk* may be much bigger.
  - Individual risks matter, but also their correlations with other investments in the overall portfolio.
- Need to measure and manage risk!

5. Risk Models

Volatility

- Risk can and should be measured in different ways.
- One very common measure of risk is *volatility*.
  - Standard deviation of an uncertain payoff or return.
  - One-to-one with the corresponding variance.
- Natural for *normal distributions* with density

$$f_{\mu,\sigma}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

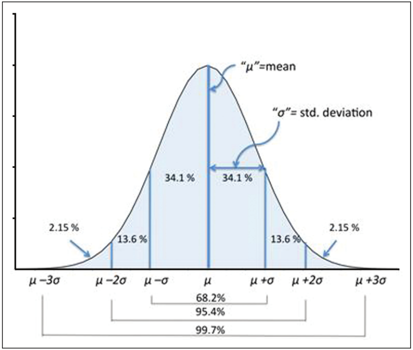
- Volatility σ fully describes dispersion around mean μ.

IMPERIAL

5. Risk Models

Volatility ct'd

- Illustration of the normal distribution and its mean and standard deviation:

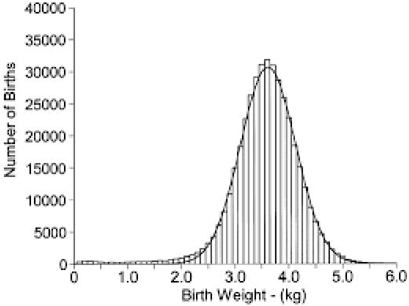


- But is the normal distribution a reasonable model for stock returns, for example?

5. Risk Models

Volatility ct'd

- Normal distribution works amazingly well in many contexts, e.g., birthweights of babies:



- Reason: *central limit theorem*. Many small independent random shocks *always* lead to normal distribution.

5. Risk Models

Volatility ct'd

- Financial shocks are not always small and independent..
- Fitted normal density vs. nonparametric estimate for return distributions:

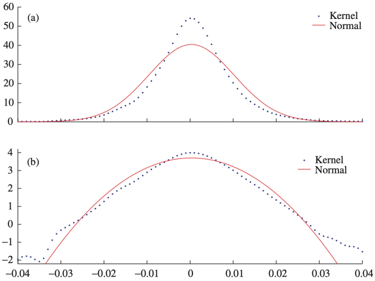


Figure 4.1 (a) Normal and Gaussian kernel density estimators and (b) log densities of the daily log returns of the S&P 500 Index.

5. Risk Models

Volatility ct'd

- Normal distribution is still really good for just two parameters!
- But does not capture skewness and excess kurtosis.
- Particularly problematic for hedge-fund returns..
  - For normal distributions: two standard deviations from mean are uncommon. Five standard-deviation events never happen.
  - For returns of hedge-fund strategies, two standard deviations are common and five standard deviation events do happen.
  - Negative extreme events much more likely than positive ones.
- In summary: volatility is very useful for distributions that are not too asymmetric or exposed to extreme crash risk.
- But don't rely on volatility alone as a risk measure!

5. Risk Models

VaR

- Another very popular risk measure is the *value-at-risk (VaR)*.
- Maximum loss that can occur with a certain confidence.
- For example, the 99%-VaR is the most you can loose with 99% confidence:

$$\mathbb{P}\left[\text{Loss } \leq \mathrm{Var}_{99\%}\right] \geq 0.99.$$

- VaR is multiple of volatility for normal distributions.
- How estimate the 99%-VaR directly from historical data?
  - Simply sort your past returns.
  - Then find a return for which 1% of returns are worse and 99% are better.
  - One has to be careful if the positions of the trading strategy and the market environment have changed significantly.

5. Risk Models

Expected Shortfall

- VaR says nothing about the magnitude of the losses if this threshold is exceeded.
- Addressed by so-called *expected shortfall*: expected loss, given that the VaR is exceeded:

$$\mathrm{ES} = \mathbb{E}[\mathsf{Loss}|\mathsf{Loss} > \mathrm{VaR}].$$

- Quite difficult to estimate reliably, since one is taking the average of only very few historical data points.
  - Sample average of returns smaller than VaR, i.e., subsample of e.g. 1% of original dataset.
- Still only takes an average over very bad losses.
- Some other – even more cautious – risk measures use convexity to put more weight on catastrophic losses.
  - Most extreme example: worst case risk measure.

- Another important characteristic of strategies, in particular for hedge funds: performance relative to their past maximum.
  - Many hedge funds only charge performance fees only when the fund performance exceeds the *high water mark*

$$\mathrm{HWM}_t = \max_{s \le t} P_s.$$

  - Typical: “two and twenty”. Management fee 2%, performance fee 20% on gains relative to the high water mark.
- Important risk measure in this context: *drawdown*

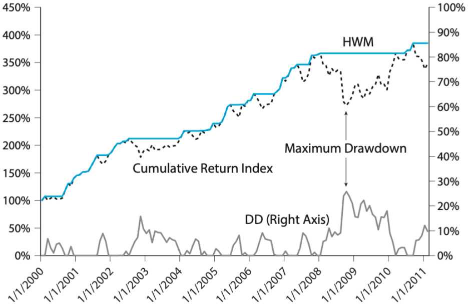
$$\mathrm{DD}_t = \frac{\mathrm{HWM}_t - P_t}{\mathrm{HWM}_t}.$$

  - Crucial because investors often withdraw their money when the fund drops too far below the high-water mark.

5. Risk Models

Drawdowns ct'd

- Example for high water mark and corresponding drawdown:



5. Risk Models

Stress Tests

- Another important class of risk measures: *stress tests*.
  - Simulate portfolio returns in extreme market scenarios.
  - Can include past events as well as hypothetical future events.
  - Also used by central banks to asses banks' capital reserves, for example.
- Goal: make sure that no position is so large that it puts the entire fund at risk.
- Challenge: how to create realistic and relevant scenarios?
  - Historical simulation.
  - Synthetic data generated from a model? E.g., parametric stochastic models, generative adversarial networks.

6. Portfolio Construction

The Task at Hand

- Suppose you have identified some promising investments.
  - Could be individual assets that look attractive.
  - Or composite strategies that appear to perform well.
- How to combine these into an overall investment *portfolio*?
  - Requires to estimate the risk and rewards associated to each investment *and* the dependencies between them.
  - Then, need to size the different positions in order to achieve the best *overall* risk-reward tradeoff.
- For active portfolios such as most hedge funds, *risk management* also is of crucial importance.
  - Risks vary substantially over time, as investment opportunities, price volatilities, and correlations change.
  - Controlling risks is particularly important with leverage or shortselling – cannot simply “ride out” long drawdowns.

6. Portfolio Construction

Budget Equations

- Starting point: wealth W_{t-1} at time $t-1$.
- Choose dollar amounts $x_{t-1} = (x_{t-1}^1, \dots, x_{t-1}^N)$ to invest in risky assets $n = 1, \dots, N$.
- Invest remaining wealth $W_{t-1} - \sum_n x_{t-1}^n$ into the safe asset.
- Once the returns $R_t = (R_t^1, \dots, R_t^N) = R^f + R_t^e$ from time $t-1$ to time t are realized, time-t wealth is

$$W_t^{x_{t-1}} = W_{t-1}(1 + R^f) + x_{t-1}^\top R_t^e$$

- What functional of wealth should we optimize?
  - Just maximizing average wealth is ill posed.
  - Want high wealth on average, but also little risk!

- Most tractable goal functional: maximize expected wealth penalized for variance:

$$\begin{aligned} J(x_{t-1}) &= \mathbb{E}_{t-1}\left[W_t^{x_{t-1}}\right] - \frac{\gamma}{2}\mathrm{Var}_{t-1}\left[W_t^{x_{t-1}}\right] \\ &= W_{t-1}(1+R^f) + x_{t-1}^\top \mu_{t-1} - \frac{\gamma}{2} x_{t-1}^\top \Sigma_{t-1} x_{t-1} \end{aligned}$$

  where $\mu_{t-1} = \mathbb{E}_{t-1}[R_t^e]$ and $\Sigma_{t-1} = \mathrm{Cov}_{t-1}[R_t^e, R_t^e]$.
- Linear-quadratic in control variable x_{t-1}.
  - Want to maximize linear reward.
  - But quadratic risk term discourages very large positions.
- Makes mean-variance optimization well posed *and* tractable!

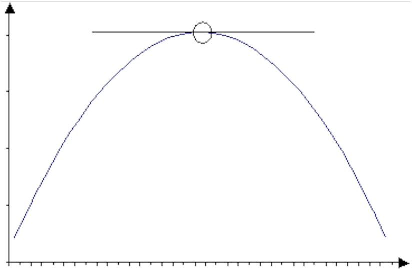
6. Portfolio Construction

Mean-Variance Optimization ct'd

- First-order condition for optimality is linear:

$$0 = J'(x_{t-1}) = \mu_{t-1} - \gamma \Sigma_{t-1} x_{t-1}$$

- Necessary and sufficient for maximum of a concave function.



6. Portfolio Construction

Mean-Variance Optimization ct'd

- First-order condition for optimality is linear:

$$0 = J'(x_{t-1}) = \mu_{t-1} - \gamma \Sigma_{t-1} x_{t-1}$$

- Simplest case: only one risky asset (say, broadly diversified index fund representing the S&P500.
- Then expected return μ_{t-1} and covariance matrix Σ_{t-1} are just numbers, optimal investment is:

$$\hat{x}_{t-1} = \frac{\mu_{t-1}}{\gamma \Sigma_{t-1}}$$

  - Invest a lot if expected returns are high, little if variance times risk aversion γ is large.

6. Portfolio Construction

Mean-Variance Optimization ct'd

- With N risky assets, μ_{t-1} is an N-vector and Σ_{t-1} is an $N \times N$ matrix.
- First-order condition is a system of N linear equations:

$$0 = J'(x_{t-1}) = \mu_{t-1} - \gamma \Sigma_{t-1} x_{t-1}$$

- Linear algebra: three cases (only depend on Σ_{t-1} not μ_{t-1}):
  1. No solution (e.g., if $\Sigma_{t-1} = 0$).
  2. Infinitely many solutions (if some equations are equivalent).
  3. Exactly one solution (generic case on which we focus here).
- In Case 3, the solution can be described in terms of the *inverse* Σ_{t-1}^{-1} of the matrix Σ_{t-1}.
  - Explicit formula for 2×2 matrices.
  - Useful theoretically, but often more efficient in practice to solve optimality equations numerically, especially with portfolio constraints (e.g., long only).

6. Portfolio Construction

Example: Time-Series Momentum

- Simple model for time series momentum?
- Suppose $\mathbb{E}_{t-1}[R_t^e] = \beta\mu_{t-1}$ where μ_t is an "autoregressive process":

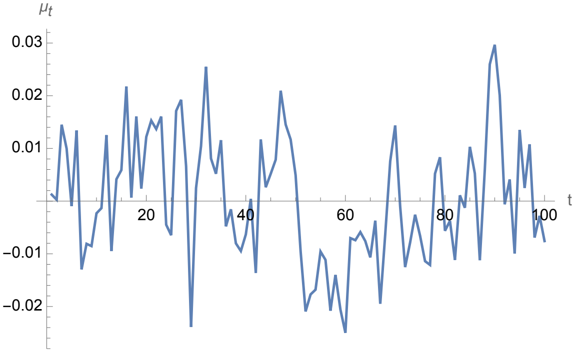
$$\mu_t = \phi\mu_{t-1} + \epsilon_t$$

  for $\phi \in (0,1)$ and iid noise ϵ_t with variance η^2.
  - Trading opportunities are partly predictable using "signal" μ_t.
  - Many possible choices used in practice, e.g., moving averages of past returns or dividend yields.
  - ϕ models persistence of trading opportunities – random fluctuations if $\phi \ll \eta$, almost constant for $\phi \gg \eta$.
- For simplicity: constant risk $\mathrm{Var}_{t-1}[R_t^e] = \sigma^2$.
- Optimal mean-variance portfolio in this context?

6. Portfolio Construction

Example: Time-Series Momentum ct'd

- Sample path of an autoregressive process μ_t:



6. Portfolio Construction

Example: Time-Series Momentum ct'd

- Mean-variance optimal portfolio at time $t-1$:

$$\hat{x}_{t-1} = \frac{\mathbb{E}_{t-1}[R_t^e]}{\gamma \mathrm{Var}_{t-1}[R_t^e]} = \frac{\beta \mu_{t-1}}{\gamma \sigma^2}$$

- Mean reverts – long in good times, short in bad times.
- How to implement in practice?
  - Choose predictive variables μ_t. E.g., dividend yield, or moving averages of past returns (over different horizons).
  - Run linear regression of excess returns R_t^e against μ_{t-1} to determine β and σ^2.
  - Choose risk aversion γ to scale portfolio to desired size.

6. Portfolio Construction

Example: Time-Series Momentum ct'd

- Parameters ϕ, η of the trading signal play no role here?
  - Only affect unconditional Sharpe ratio, not optimal strategy.
- Changes when transaction costs are considered!
  - If ϕ is large relative to η, positions are held for a long time which helps earn back transaction costs.
  - If ϕ is small relative to η, positions fluctuate quickly, making it hard to earn back costs.
- With transaction costs, signals that decay quickly should be scaled back more than persistent ones.
- Formalizing this requires dynamic multiperiod models as in, e.g., Garleanu/Pedersen '13.

6. Portfolio Construction

Example: Risk Models for Multiple Assets?

- For a single asset are strategy, the corresponding variance is relatively easy to estimate from historical data.
- But for many assets, the number of parameters grows quickly even if the covariance matrix is assumed to be constant over time.
  - For example, the covariance matrix of the S&P500 has $500 \times 500/2 + 250 = 125250$ parameters..
- Without imposing additional structure, this leads to various problems:
  - In theory, covariance matrices are positive definite, but high-dimensional estimates often are not.
  - Then risk can become negative and optimization explodes..
- Way out?

- How to simplify high-dimensional covariance estimation?
- Impose lower dimensional factor structure.
- Simplest case: market model.
  - Covariance only via correlation with market portfolio.
  - Only need to estimate variances and covariance with market portfolio.
  - For S&P500: $500+500=1000$ instead of 125250 parameters.
- Commercial risk models (e.g., by Barra or Northfield) are more sophisticated versions of this.

- So far: transactions trade off expected returns and risk.
- But if positions of a strategy turn over quickly, trading costs accumulate quickly in real life.
- Strategies that look profitable in “paper trading” often end up loosing money net of costs.
- How to address this?
  - Reduce turnover!
  - Only trade when trading opportunities are really worth it.
  - But how? Depends on what transaction costs are the relevant ones.

- Decreasing transaction costs (as a function of trade size):
  - In OTC markets, you often have to call a dealer on the phone to trade.
  - Processing each order takes the dealer a similar amount of time, independent of the size.
  - Therefore costs per share tend to be smaller for large order in such markets.
  - Makes small trades very unfavorable.
  - Thus it is optimal not to trade at all until the actual position deviates too far from a continuously adjusted "target position".
  - When this happens, you call the dealer and trade back to your preferred allocation.

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Constant costs: bid-ask spreads:
  - Also called *proportional* costs since they are proportional to trade size (but constant per unit traded).
  - Relevant for strategies that trade positions small enough to be filled at the best bid-ask prices in electronic limit-order books.
  - Constant costs penalize small trades less severely than decreasing costs.
  - Optimal strategies thus do not prescribe to trade all the way back to a target strategy once a certain threshold is breached.
  - Instead, one performs just enough trading to keep the deviation from the target small enough.

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Increasing costs (as a function of trade size): market impact.
  - If the trades of a strategy exhaust the liquidity provided at the best bid and ask prices, then it causes *market impact*.
  - Adverse effect on execution prices is bigger the larger the position that is trade.
  - Whence, if these costs are the most relevant ones, then it it optimal to split up larger trades into many small ones.
  - One the gradually trades towards the target allocation, e.g., by reducing the deviation by a certain percentage each day
- Main source of "slippage" for institutional investors, so we focus on these costs here.

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Simplest model for market impact costs:
  - Impact affects each trade separately and is *linear* in traded amount.
  - Trade $\Delta x_{t-1} = x_{t-1} - x_{t-2}$ at time $t-1$ settled at $P_{t-1}(1 + \frac{1}{2}\lambda\Delta x_{t-1})$ instead of P_{t-1}.
  - Reduces return $(P_t - P_{t-1})/P_{t-1}$ by $\frac{1}{2}\lambda\Delta x_{t-1}^2$.
  - Expected returns are reduced, (conditional) variance remains unchanged.
- Mean-variance optimization problem becomes

$$\mu_{t-1}x_{t-1} - \frac{\gamma\sigma^2}{2}x_{t-1}^2 - \frac{\lambda}{2}\Delta x_{t-1}^2$$

(One risky asset and zero interest rates for simplicity.)

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Mean-variance optimization with market impact:

$$\mu_{t-1}(x_{t-2} + \Delta x_{t-1}) - \frac{\gamma\sigma^2}{2}(x_{t-2} + \Delta x_{t-1})^2 - \frac{\lambda}{2}\Delta x_{t-1}^2$$

- Incoming positionx_{t-2} can no longer be adjusted freely but becomes an extra "state variable" that influences the solution.
- First-order condition for optimality:

$$0 = \mu_{t-1} - \gamma\sigma^2(x_{t-2} + \Delta x_{t-1}) - \lambda\Delta x_{t-1}$$

- Optimal trade:

$$\Delta x_{t-1} = \frac{\mu_{t-1}}{\gamma\sigma^2 + \lambda} - \frac{\gamma\sigma^2}{\gamma\sigma^2 + \lambda}x_{t-2}$$

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Optimal position with market impact:

$$x_{t-1} = x_{t-2} + \Delta x_{t-1} = \frac{\gamma\sigma^2}{\gamma\sigma^2 + \lambda} \frac{\mu_{t-1}}{\gamma\sigma^2} + \frac{\lambda}{\gamma\sigma^2 + \lambda} x_{t-2}$$

- Convex combination of incoming holdings x_{t-2} and frictionless mean-variance portfolio $\mu_{t-1}/\gamma\sigma^2$.
- Intuition:
  - Without trading costs, would trade to mean-variance portfolio.
  - With costs, only trade part of the way. Fraction depends on ratio of trading costs λ to risk $\gamma\sigma^2$.
- Dynamic version with many trading times?

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Garleanu/Pedersen (2013): choose trades Δx_t to maximize

$$\mathbb{E}\left[\sum_{t=0}^{\infty}\rho^t\left(\mu_t x_t - \frac{\gamma\sigma^2}{2}x_t^2 - \frac{\lambda}{2}\Delta x_t^2\right)\right]$$

  for a discount factor $\rho \in (0,1)$.
- With trading costs, each trade influences the incoming positions for the subsequent ones, so the trades cannot be optimized separately.
- But problem can be solved with the "dynamic programing" approach of stochastic optimal control.

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Structure of the solution is similar to the one-period case:
- Optimal position at time t is convex combination of incoming holdings and a "target portfolio".
- But target is generally no longer the frictionless optimum.
- Instead: "aim in front of this moving target".
  - Trade to suitable average of current and future values of the frictionless portfolio.
  - Reduces and smoothes out holdings to save transaction costs.
  - Trading speed again depends on ratio of trading costs to risk (and time discounting).

6. Portfolio Construction

Extensions: Transaction Costs ct'd

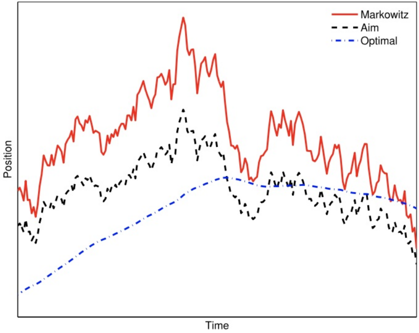
- Intuition for "aiming in front of the target":
  - *A great hockey player skates to where the puck is going to be, not where it is* (Wayne Gretzky)

Panel D. "Skate to where the puck is going to be"   Panel E. Shooting: lead the duck   Panel F. Missile systems: lead homing guidance

6. Portfolio Construction

Extensions: Transaction Costs ct'd

- Optimal portfolios with and without market impact costs:



6. Portfolio Construction

General costs: problem formulation

- Let c denote a cost function. Examples:
  - quadratic costs (as above): $c(\Delta x_t) = \frac{\lambda}{2}\Delta x_t^2$
  - power costs: $c(\Delta x_t) = \lambda_q \Delta x_t^q$ for $q \in [1, 2)$ (empirical studies: $q \approx 3/2$).
- Aim to choose trades Δx_t to maximize

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \rho^t \left(\mu_t x_t - \frac{\gamma\sigma^2}{2} x_t^2 - c(\Delta x_t)\right)\right]$$

  for a discount factor $\rho \in (0, 1)$ or

$$\mathbb{E}\left[\sum_{t=0}^{T} \left(\mu_t x_t - \frac{\gamma\sigma^2}{2} x_t^2 - c(\Delta x_t)\right)\right]$$

- For simplicity focus on the second problem. The same techniques apply to the first problem and many related contexts.

6. Portfolio Construction

General costs: problem formulation

- Aim to choose trades Δx_t to maximize

$$\mathbb{E}\left[\sum_{t=0}^{T}\left(\mu_t x_t - \frac{\gamma\sigma^2}{2}x_t^2 - c(\Delta x_t)\right)\right]$$

- Each trade influences the incoming positions for the subsequent ones, so the trades cannot be optimized separately.
- For many realistic choices of c the problem can *not* be solved using "dynamic programing".
- Deep learning approach:
  - Computes approximately optimal strategy numerically
  - Applicable in very general contexts
  - May be computationally expensive and require hyperparameter tuning (learning rates, batch size, ..., as in previous coding examples).

6. Portfolio Construction

General costs: deep learning approach

- Aim to choose trades Δx_t to maximize

$$\mathbb{E}\left[\sum_{t=0}^{T}\left(\mu_t x_t - \frac{\gamma\sigma^2}{2}x_t^2 - c(\Delta x_t)\right)\right] \tag{1}$$

- How can we use deep learning to solve such a problem?
  - We don't know the optimal trading strategy.
  - Different from previous examples: no input-output pairs of data are available.
- *Deep learning approach:*
  - Use a neural network f_θ to choose your holdings at each time t.
  - Use (1) as your "loss function"
  - Use stochastic gradient methods to find optimal neural network parameters
- Example of *reinforcement learning*

6. Portfolio Construction

General costs: deep learning approach in detail

- First change signs: aim to choose trades Δx_t to minimize

$$-\mathbb{E}\left[\sum_{t=0}^{T}\left(\mu_t x_t - \frac{\gamma\sigma^2}{2}x_t^2 - c(\Delta x_t)\right)\right]$$

- Now choose Δx_t as a neural network. In detail:
  - Conjecture: optimal strategy at t only depends on μ_t and the previous position x_{t-1}
  - This motivates to choose holdings $x_t^\theta = f_\theta(t, x_{t-1}^\theta, \mu_t)$ for a neural network f_θ
  - Note: $x_{t-1}^\theta = f_\theta(t-1, x_{t-2}^\theta, \mu_{t-1})$ is itself the output of a neural network at $t-1$.
  - The neural networks are iteratively applied to each other $\rightarrow$ implementation is more challenging.
  - Neural network can be chosen as recurrent or feedforward.

6. Portfolio Construction

General costs: deep learning approach in detail

- The loss $\mathcal{L}$ can then be optimized using stochastic gradient descent, as before for the case

$$\frac{1}{n}\sum_{i=1}^{n}\ell(f_\theta(x^i), y^i).$$

- Now $\mathcal{L}$ is more complex $\rightarrow$ we need to build a custom loss function in the implementation.
- Look at it in detail in Guangyi's tutorial.

6. Portfolio Construction

General costs: discussion of deep learning approach

- Reinforcement learning: we learn the optimal trading strategy directly from maximizing the trading objective.
- The same approach can be applied to a wide variety of related problems.
- Extremely flexible, is indeed able to learn good strategies!
- Main point to be careful about: for longer time horizon T we concatenate more and more neural networks $\rightarrow$ harder to find optimal neural network parameters.
  - Good performance hinges on right selection of hyperparameters.
  - Training the neural network can be computationally expensive.
  - Potentially long training times.