ULTIMATE EDITION

The full collection of quant trading resources to guide you along. Suitable for beginners and professionals alike. Beginners may find the Essentials or Comprehensive edition more appropriate as it is more distilled with a stronger learning structure.

QUANT ROADMAP

2024/2025

Author: Stat Arb
Twitter: quant_arb

About The Roadmap

Quantitative trading has a reputation for being very hard to break into, and frankly that is true. Even for professionals, it can be hard to tell what resources are worthwhile.

Thus, the quant roadmap serves as a resource of resources, designed to highlight all the worthwhile materials at your disposal. With all this content available, it can be hard to know where to start. Thus, there are 3 versions of the roadmap this year:

1. Ultimate Edition [link] [backup link]
2. Comprehensive Edition [link] [backup link]
3. Essential Edition [link] [backup link]

The ways they differ focus on the trade-off between being all encompassing and avoiding overwhelming the reader. If you want a pure directory of every worthwhile resource, the Ultimate Edition is for you. We have done our best to organize the sections, but there is not much in the way of guidance on how to learn it.

Comprehensive aims to strike a careful balance between presenting the most important resources directly to the reader, covering a variety of material, and guiding readers on how to learn it.

Finally, for those who feel they have no idea where to start – the Essentials Edition is here for you. This focuses heavily on how to learn the material and being as efficient as possible with the learning (covering core topics as opposed to fringe ones).

The level of noise will also increase going from essential to ultimate. Resources in essentials are maximally orthogonal and skip the more niche topics. Ultimate covers everything and many resources may have significant overlap. That's a balance that's hard to strike, but readers will find that discovering what works best for their own learning process is a fulfilling experience.

Comprehensive also includes coverage of pre-requisites, so does Ultimate. If you know background topics (mathematics, computer science, etc.) then feel free to jump to Essentials and work up. Otherwise, consider starting with Comprehensive.

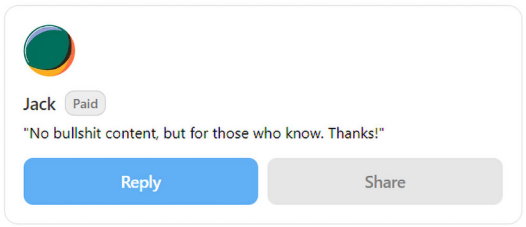
This is no substitute for real work and implementation. The ultimate edition especially focuses on cataloguing the available resources, but you cannot learn them by brute force reading them. They can only truly show you what you don't know you don't know. For true learning you must engage with the material through conversation, implementation, and modification of your own.

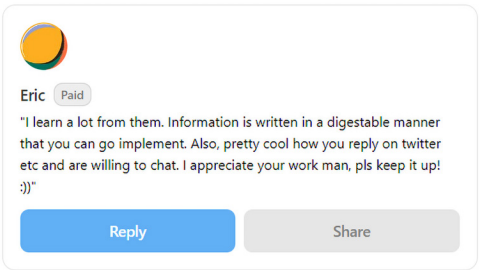
About The Author

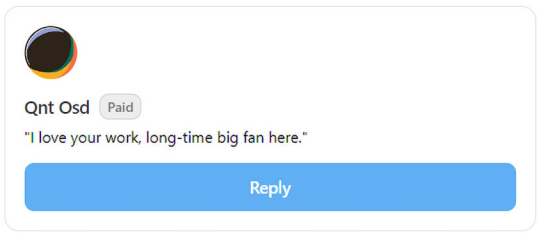
I work as a quantitative researcher in the digital assets space and have led teams across HFT and MFT strategies. Both at my own hedge fund/ shop and at a larger one as head of quantitative research. I'm no boomer with decades of experience, but I hope to say I've got enough to be worth sharing with everyone.

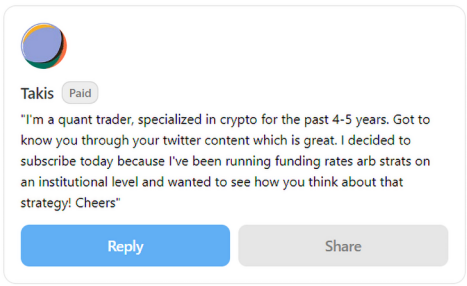
Over the years I've used many usernames, the last edition of roadmap was under BBM and referred to my old Twitter handle @TerribleQuant. I have since changed this because it hasn't done me well in any Twitter arguments funnily enough, but I can now be reached at @quant_arb on Twitter & on Instagram (repost account).

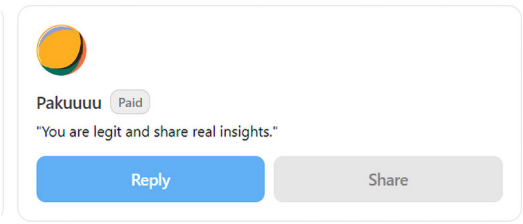
I run www.algos.org which is my blog. It's got tons of articles I've written which I think are a great resource, some are free, others require subscription. Consider my blog the sponsor of this edition of the guide. Here are some reader testimonials, I've removed names in case they didn't know they would be featured, and traders aren't typically a fan of getting any personal publicity:)

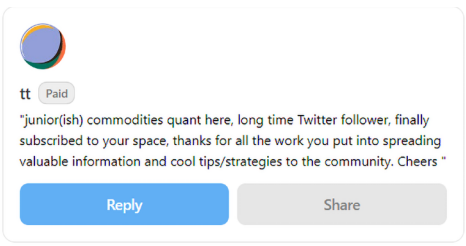

Jack
Paid
"No bullshit content, but for those who know. Thanks!"
Reply
Share

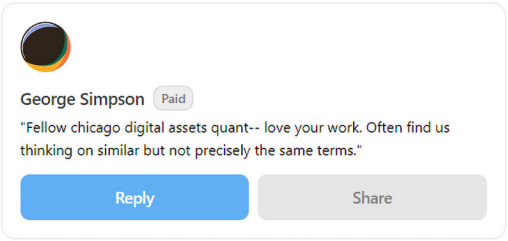

Eric
Paid
"I learn a lot from them. Information is written in a digestable manner that you can go implement. Also, pretty cool how you reply on twitter etc and are willing to chat. I appreciate your work man, pls keep it up! :))"
Reply
Share

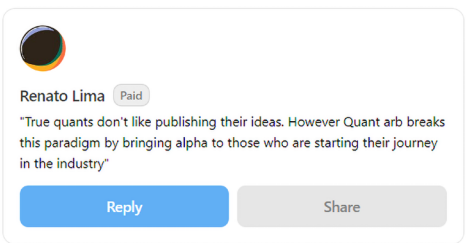

Qnt Osd
Paid
"I love your work, long-time big fan here."
Reply


Takis
Paid
"I'm a quant trader, specialized in crypto for the past 4-5 years. Got to know you through your twitter content which is great. I decided to subscribe today because I've been running funding rates arb strats on an institutional level and wanted to see how you think about that strategy! Cheers"
Reply
Share


Pakuuuu
Paid
"You are legit and share real insights."
Reply
Share


tt
Paid
"junior(ish) commodities quant here, long time Twitter follower, finally subscribed to your space, thanks for all the work you put into spreading valuable information and cool tips/strategies to the community. Cheers "
Reply
Share


George Simpson
Paid
"Fellow chicago digital assets quant-- love your work. Often find us thinking on similar but not precisely the same terms."
Reply
Share


Renato Lima
Paid
"True quants don't like publishing their ideas. However Quant arb breaks this paradigm by bringing alpha to those who are starting their journey in the industry"
Reply
Share

CONTENTS

Chapter 1 *Machine Learning and Algorithmic Trading (Textbooks)*

Anything highlighted in red is optional since it is more of a repeat with extras of the textbook in black before it. Depends on how hard/ fast you want to learn! You should do the first textbooks then decide whether to do the Machine Learning Section or the Derivatives section first, but you can do them simultaneously. They have crossovers and I love both areas although I am more partial to the former, however they are very much independent and do not require knowledge from the other to learn. All textbooks point to Amazon links, but make sure to avoid libgen because it has them all for free.

Disclaimer:

I am not responsible if you commit piracy and I do not recommend you do this because it is wrong, but I hear that some people find it useful for checking if it isn't rubbish before buying. Also buying the actual book means you get an impressive bookshelf/ some think it is better to read, but I enjoy both PDF and physical. PDF purchased through the author of course…

1) "Quantitative Trading 2nd edition"
   1)a"Algorithmic Trading"
   1)b"Machine Trading"
2) Trading and Exchanges - This can be skipped if you are not interested in microstructure, but regardless is quite helpful to know as it can make sure you avoid some silly mistakes.

Note: Quick note on Ernest Chan Books. They aren't very meaty but are an easy intro so feel free to skim through them. Especially 1) a & b (in red) are very basic to the point where unless you are 100% new to quant they should be skipped.

ML (1/3) Machine Learning for Algorithmic Trading

There is a lot of overlap between machine learning… and mastering python… so start with one of them then read finding alphas then read the other. That is why they are noted as either the 1st or 3rd book to read in terms of machine learning.

ML (1/3) Mastering Python for Finance

Go do this all before doing finding alphas if you "audit" the course then it is free to do them all individually.

https://www.coursera.org/specializations/investment-management-python-machine-learning

- Note: This is a 4-part course so there is certainly a lot to go through, but I think it is one of the best resources because it uses legacy models to build intuition, but unlike most courses then goes on to show you some actual methods that are used and work in the industry. Another BIG benefit is that it uses Python in Jupyter notebook which in my opinion is the best way to do research. Orange is good as well and an R kernel in Jupyter is also a nice alternative (more on that later).

ML (2) Finding Alphas: A Quantitative Approach (2nd edition)

ML (4) Advanced Algorithmic Trading - I think all of the backtests shown are overfit to make them look better, but it is a good idea to get familiar with approaches / ideas to improve your own creative process.

ML (5) Advances in financial machine learning (The first few chapters are brilliant, middle chapters are pretty good, and the last chapters are abhorrent. It goes from insanely good to insanely bad. MLDP is truly the Nicolas Cage of quants. He either writes the worst paper you have ever seen about the nichest nerd hole with no relevance to making money ever… or he cooks up an amazing method that is quite useful. I don't know what to say honestly, but regardless of my views on his work the first few chapters are a MUST for all quants)

ML(6) The elements of statistical learning -general ML knowledge – Less math heavy version (Introduction to Statistical Learning in R / Python)

ML(7) Regression Analysis with Python

ML(8) Regression Modelling Strategies

Regressions may appear as though they are the most boring and basic tool that no real quant other than beginners would use, but in reality they are the opposite. It is mostly beginners who use complex machine learning models, and the professionals who use the simplest of models. This may be hard to understand but the core of it is answered by the data itself. The data is noisy, high dimensional, and with the slightest nudge you can overfit to it. Every beginner massively overestimates how much margin they have to fit to the data – you just don't have much. Hence, regressions are the favorite. Ridge, Median Regressions, and MAD personally.

Also, may be useful to go do the machine learning courses on Coursera, but of course it won't be finance focused just building a general understanding of what things are and how they work.

Bonus Books:

- Elements of Quantitative Investing Draft (this is a book written by @__paleologo on Twitter, and is in draft form but is incredibly good so well worth the read)
- Quantitative Portfolio Management
- Advanced Portfolio Management
- Trader Construction Kit
- Optimal Trading Strategies
- Advanced Futures Trading Strategies

4th Edition Extras:

Algorithmic Trading Methods – Goes through some models

Trades Quotes and Prices – Microstructure textbook

Machine Learning in Finance – More rigorous for ML in Finance

Data Driven Science and Engineering – Not specifically quant, but there is huge alpha from engineering/ signal processing.

5th Edition Extras:

Analysis of Financial Time Series

Bayesian Data Analysis

Digital Signal Analysis: An Introduction (R. Anand) – Not specifically quant, but alike 'Data Driven Science and Engineering' it presents a lot of opportunities to find interesting methods that can be applied to quant.

Robert Carver Textbooks (These are an alternative to some of the initial textbooks):

Systematic Trading

Leveraged Trading

With the exception of the Robert Carver books which are solely there to replace the Ernest Chan books at the discretion of the reader (or if they find one of them confusing), the textbooks are ordered in terms of value for the extras. The first selection of bonus textbooks are the ones I believe to be essential additions, and then from there they become more and more additional.

Chapter 2 *Derivatives, and Volatility Trading (Textbooks)*

I never expected to be doing much options trading in my lifetime beyond running statistical arbitrage strategies when I wrote the 4th edition over 2 years ago. Since then, I've worked on building out an options market making operation, and I can certainly say that the knowledge will eventually come in handy so at least the basics are worth learning regardless. If your career is long enough you'll interact with options enough to at least think about how they work.

Derivatives (1) Hull Options Futures and other Derivatives

Derivatives (1 Alternative) Option Trading & Volatility Trading (both textbooks by Euan Sinclair)

Derivatives (2) Positional Option Trading

Derivatives (3) Trading Volatility

Extra Derivatives:

Option Volatility and Pricing: Advanced Trading Strategies and Techniques

Trading Options Greeks: How Time, Volatility, and Other Pricing Factors Drive Profits, Second Edition

The Volatility Surface

The Volatility Smile

Currency Derivatives

Exotic Options and Hybrids – This is more for if you want a career on an exo desk

Dynamic Hedging (NNT) – This is mainly for people pricing exotics, which is helpful for a very popular starting position on an exo desk or doing options MM at a prop firm. I found this useful when looking at options market making because it's how you price complex risks which you often get into when doing option market making.

Stochastic Volatility (Textbook)

BONUS CONTENT: Options Market Making

How does option market making work? Roughly speaking, you have a surface of all options you quote, likely one per exchange, and then you modify it based on trade impacts, moves in spot, and of course your own inventory to get a dynamic fit of it.

Your basic surface is just a fit of the market and then you will add skew based on your Greeks. For more on how to fit the surface in an advanced way (not yet public in the academic literature) here is a great article I wrote.

For an old, but still highly relevant textbook, I recommend checking out this:

Option Market Making

Chapter 3 *YouTube Videos*

Here are some great videos by Ben Felix. I can honestly recommend all of his videos but these grasp at key point all traders need. Very asset pricing model/ EMH based and whilst I go for EV (Expected Value) it is still important to know. In the podcasts section Vivek Viswanathan on Flirting with Models gives a good explanation of how EV models can work with factor investing and what is wrong/ right about factor models.

https://www.youtube.com/watch?v=jKWbW7Wgm0w https://www.youtube.com/watch?v=foqswJT3Spc https://www.youtube.com/watch?v=yco0sC7AJ2U https://www.youtube.com/watch?v=IzK5x3LlsUU

LEARN VOLATILITY

Patrick Boyle has some great books, but I also recommend his playlists. Especially the last 3 rows, which is a full education in derivatives, and he breaks down financial news in a meaningful and educational way that is fun to watch, but without the narrative:

https://www.youtube.com/c/PatrickBoyleOnFinance/playlists

Leonardo Valencia (Some really great volatility videos, I recommend you watch them)

Tasty Trade (I like their Greek videos, but still prefer Patrick Boyle)

KeyPaganRush (Great visualizations for Charm and Vanna)

LEARN ALGOTRADING

Part time Larry:

Crypto Wizards:

Quantconnect YT channel more on them later

Quantra

Trade options with me (teaches Quantconnect)

Jacob Amaral (Nothing too remarkable, but a few decent trading algo vids)

Algo vibes: (great for people who are just starting)

Note: Algovibes is probably one of my top recommendations. He demonstrates how to build a full system on many occasions and makes it as simple as possible plus also how to research in Jupyter notebook.

LEARN SIGNAL PROCESSING

Mike X Cohen (Signal Processing applied to neuroscience, but still great)

Two great playlists by him (good alpha here)

https://youtube.com/playlist?list=PLn0OLiymPak2jxGCbWrcgmXUtt9Lbjj_A

https://youtube.com/playlist?list=PLn0OLiymPak2G__qvavn3T8k7R8ssKxVr

Abhishek Agrawal (signal processing stuff in python)

Esther Explains: (Some cool signal processing stuff)

LEARN QUANT TOPICS

1. AP Monitor (optimization, very niche/complex, but I like it)
2. Niche optimization channel (again I like it)
3. Quantpie
4. Niche and complex math channel lol
5. AI Quant (Extremely complex stuff, but it's GOLD)
6. H&T (I will not be mean to them in this edition. They're mostly interns and make similar mistakes as all beginners to the practical side that I've seen – overcomplicating things mostly, but the work is high quality)

LEARN DATA SCIENCE/GENERAL CODING

1. Neural Nine is more general Python, but has some good algotrading vids
2. Data Science Dojo (great data science stuff)
3. Ken Gee (general data science)
4. Coding Jesus
5. Finn Eggers (java DL stuff)
6. Keith Galli (good python tutorial)
7. Gerard Taylor (specifically I recommend his ML in C++ course)
8. Data Professor (Just general data science):
9. Ahmad Bazzi

LEARN STATS/DS MATH

1. Luis Serrano: Good statistics videos
2. Complex math channel lol
3. More easy data science/ stat videos
4. Stat Quest (really engaging guy for complex stat topics broken down so anyone can get it)

OTHERS

1. RCM Alternatives
2. Martin Shkreli
3. Mutiny (Listen to every single podcast they have; you won't regret it)

Chapter 4 Courses

On Coursera Robert Shiller has a course called Financial Markets. It is free without the certificate $50 for the certificate. The videos are also on YouTube. This is an amazing start for finance and the markets in general and will teach you the basics of everything in the markets. Coursera link below:

https://www.coursera.org/learn/financial-markets-global

Medium to Higher level:

Andrew Ng has a course on Machine Learning and Deep Learning on Coursera. Those are really good but quite math heavy.

For the math Imperial College London has a Mathematics for Machine Learning course series and it has multivariate calculus, linear algebra and PCA. All of which will be super helpful.

There is also a Financial Engineering Part 1 & 2 course on Coursera as well.

This is also a great one for machine learning in python and has some really great strategies included in there:

https://www.coursera.org/specializations/investment-management-python-machine-learning#courses

https://www.coursera.org/projects/intro-time-series-analysis-in-r

This is a great project you can do in R. Amazing stuff 100% recommend. I really do stress that this is a great resource.

RobotJames & HangukQuant on Twitter both have courses that are currently out, as well as Euan Sinclair but they're all very expensive.

I plan on releasing a course on arbitrage and HFT in digital asset markets co-created with a developer from the equities world in a few months so perhaps it will be out by then. Half of it at least is coded up at the time of writing (it's very heavy in the code provided as it aims to leave you with the ability to immediately put everything into practice)

Chapter 5 *Podcasts*

(One of the best resources DO NOT TAKE LIGHTLY)

Most information especially the most useful is not in textbooks so you need to religiously study podcasts. Think of textbooks as foundational but read them to understand what is talked about on podcasts.

- Tick Talk (My Podcast)
- Flirting with models - Also, amazing and you NEED to listen to all the episodes (I have listened to some episodes 6+ times.)
- Mutiny fund (YouTube channel shown)
- RCM Alternatives, the derivative
- Market Champions

Chapter 6 *Trading Platforms & Brokerage Firms*

For equities, and practically all other asset classes I recommend IBKR. I use them personally and find that they have the best offering. They also offer incredibly cheap data along with their brokerage services, and a somewhat limited PB service as you scale out.

TD Ameritrade is a great alternative, and equally as high quality as a retail brokerage offering.

If you are an institutional player in the equities space, then you will need a prime brokerage service. All the major investment banks offer them, so think JMPC, MS, GS, etc. What do they come with?

- The usual brokerage service
- Lower costs to trade
- Shorting capabilities (ability to get large borrows)
- Leverage
- Insurance against exchange counterparty (you face the prime broker not the exchange, whether that's good or bad is your own call, but big funds spread their cash around different PBs to hedge this risk)
- Hedge fund services (many of them will do everything from raising capital to the legal work of a hedge fund and all you do is lay back and make money – they take a hefty cut for this service but some go for it)

In the digital assets realm, your broker is also the exchange – so things get a little bit more nuanced. Most large shops will trade on many exchanges, your 3 largest being:

- Binance [link]
- Okx [link]
- Bybit [link]

These are all high quality. For crypto options, Deribit [link] is the largest, but for futures (the largest market in crypto by volume), Okx, Bybit, and Binance are the largest. As I currently write this, fiat on ramp/off ramp is not the easiest – so CDC, Kraken, and Coinbase are my recommendations. US firms in general are not great in terms of flow but very easy to get fiat into crypto through, and vice versa.

Prime brokers in the digital assets space offer a bit more of a limited service, but they include:

- HRP (Hidden Road Partners)
- FalconX
- Matrixport
- LTP
- Copper

I have contacts at all of these and can make introductions if you ask. US firms or even firms with a slight US connection are barred from all of these in the current regulatory environment. I hope this will have become an obsolete sentence by the next edition, but for now this is the case.

HRP offers exchange insurance where you never actually give them money, they just loan you it based on your balance sheet and then charge you interest + exchange collapse insurance (which is incredibly underpriced btw).

They all differ by what fees they offer, as currently LTP is the biggest, but a year ago this was completely different. Not all exchanges do fees by account, some do it by subaccount, so the PB model doesn't offer amazing exchange fees for every exchange anymore and is a leverage provider after that now.

Speaking of leverage, if you want access to leverage, there are firms like Tesseract, Maple, & Cicada which can provide capital for levering which is underwritten against a firms balance sheet. Again, this is crypto specific.

For trading platforms that make it much easier to implement strategies, it's worth having a look at QuantConnect which is my favorite, but Nautilus trader also has a fair bit of progress. For implementing simple strategies, you can't go wrong with QuantConnect.

Chapter 7 *Neural Networks / ML / Hype*

As many will know, neural networks and machine learning methods were a favorite of mine, especially differential geometry based approaches. Beware of manifold learning alphas as they are not some amazing solution to the markets as many will believe at first. This was my original view when I had more time to toy around. Nowadays, I use bar charts, scatter plots, and linear regressions. I recommend you don't make my same mistakes and go too deep into these topics.

My recommendations for what is worth learning are as follows:

- Regressions
- Non-Parametric Methods
- Trees
- GAMs

Neural networks are not a core area of the markets. They are not the best way to forecast price, and have some niche applications to alternative data. I do not recommend becoming a neural networks professional unless you already have this expertise because you will be put into a niche. You are at the start of your career, it's already a busy niche so I'd say it's worth letting yourself fall into a niche rather than forcing yourself into one.

A lot of the resources for neural networks in finance are obsolete. They teach LSTMs, same with the papers in the literature. Tree models are known to be better for tabular data (which is what we have in finance). Your main application is alternative data (NLP on pundits on CNBC for example). CNNs are not cutting edge in this field anymore… although convolution is still important. You may do well to play around with the many open-source models in search of this alpha instead of building your own, but then again, this is a niche that may not be for everyone. Certainly isn't mine.

Great NN guide (github book):

Week 1 · Deep Learning (atcold.github.io)

Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow

Pattern Recognition and Machine Learning

Probabilistic Machine Learning

The Elements of Statistical Learning (referenced earlier, but this is a great textbook)

Andrew Ng Coursera Courses:

https://www.coursera.org/learn/neural-networks-deep-learning

https://www.coursera.org/professional-certificates/tensorflow-in-practice

https://www.coursera.org/specializations/generative-adversarial-networks-gans

https://www.coursera.org/specializations/tensorflow-advanced-techniques

https://www.coursera.org/specializations/natural-language-processing

Reinforcement Learning is a well talked about topic to learn and can be used for HFT in LOBs where orders will significantly move the market because we are dealing with microstructure. It is helpful for market making as well since MM is a control problem inherently, and RL is the NN application to control problems. I don't feel that RL is very well applied to topics like option pricing where we already have solutions for them, it's a way to overcomplicate the problem, but I do know of firms that have used RL successfully to trade in HFT manners. Certainly, there is a use to online parameter tuning, but not necessarily the neural network component.

ML/RL course coursera – The last two courses in the specialization are the best two.

Applying Deep Reinforcement Learning to Trading (Lecture YT)

Implementing Deep Reinforcement Learning Papers in Python (Helpful for implementing MM papers for RL/ whatever your chosen application is, freeCodeCamp is a great channel)

Chapter 8 *Key Mathematics Concepts*

Whilst it is entirely possible to go through algorithmic trading topics whilst minimizing mathematics exposure, there are key caveats that you will often miss without a deep understanding of your models. I am not necessarily a believer that long equations provide intuition at a high or mid-level, but at a low level (which is less intuition, more a deep understanding) it is key.

For those determined to avoid mathematics, stick to YouTube video, papers with code examples, and most of all Packt textbooks which do a brilliant job of explaining. I prefer this format where you are free to explore the deep theory separately.

Measure Theory (Brings intuition to a lot of things):

Solutions to a great book for measure theory

Measure Theory

Measure Theory YT Playlist (I recommend any of the playlists on this YT channel, will cover a lot)

General Mathematics Topics (Econometrics included as it is large/ appliable):

Econometrics (This is incredibly important)

MIT 6.042J Mathematics for Computer Science, Spring 2015

MIT: Topics in Mathematics with Applications in Finance

Coursera Mathematics for ML (In courses)

Stochastic Calculus (Masters/Advanced Undergrad topic for derivatives/MM/pairs):

Stochastic Calculus and Financial Applications (Most quants I speak to have read stochastic calculus for finance instead, but having read this one fully, and skimmed the other, I prefer this by a mile)

Stochastic Control for Finance (This is necessary for anyone who wants to go prop/ MM)

Stochastic Control Theory and High Frequency Trading (PPT by Knight Capital, this applied more)

Probability Theory (Needed for everything, great intuition/ thinking framework):

MIT RES.6-012 Introduction to Probability, Spring 2018

MIT 6.041SC Probabilistic Systems Analysis and Applied Probability, Fall 2013

Probability Theory: The Logic of Science (The OG probability textbook)

FEA (Physics alpha):

MIT Linear Finite Element Analysis

MIT Nonlinear Finite Element Analysis

Chapter 9 *Optimization (Deterministic & Stochastic)*

Why is optimization important? In my time I've used optimization models for:

- Estimating Implied Volatility [link]
- Estimating Implied Distributions [link]
- Automated Alpha Discovery [link] [link]
- Portfolio Optimization [Article I wrote that touches on this with some code]
- Pairs Trading Strategies [Chapter 20]
- Market Making [Chapter 19]
- Optimizing Strategy Parameters

When the gradient is not clear we can use genetic algorithms, however, these are noisy and can get stuck. They also use a lot of compute resources. Alternatively, we can take a more complex approach by estimating the gradient with an actor. This is an approach borrowed from reinforcement learning. We can use this as our gradient and apply methods that require them. I find that reinforcement learning works best in an HFT environment when you can only rely on live trading (i.e. test in prod) to get any real results. Some firms are more practically minded than others and prefer to optimize parameters, whereas others will tune them manually.

If you are calculating optimal portfolios the optimization can get quite complicated, and same with some advanced pairs trading strategies so hence this is a field worth learning. At a minimum, you should understand the simpler algorithms:

- Genetic Algorithms
- Basin Hopping
- Monte Carlo Optimization
- Convex Optimization Basics
- Linear Optimization Basics
- Simplex

- Newton Raphson

If we know the gradient however, we can use most optimization algorithms on this. Optimization problems may have constraints such as budgets this is something that is key to understand when engineering this into your optimization algorithms.

Monte Carlo methods are very often encountered in my own work so I recommend learning these, but ensure that you don't overcomplicate the problem. Getting to a minimal effective solution in as little work as possible is the final goal at the end of the day.

Linear Algebra and Optimization for Machine Learning: A Textbook

Algorithms for Optimization

Convex Optimization Algorithms

Variable Ordering Structures in Vector Optimization

Numerical Optimization

Hands-on genetic algorithms with Python

Accelerated Optimization for Machine Learning First-Order Algorithms

First-Order and Stochastic Optimization Methods for Machine Learning

Note: Convex Optimization & Optimization In General

When I first got into quantitative finance, we didn't have any LLMs obviously, but nowadays you can ask ChatGPT, Claude, or whatever LLM is best by the time you are reading this to produce a solution in Python to your optimization problem.

For that reason, there is a less of an importance on understanding some of these methods, but not all of them. For certain tasks, you will have a hard time getting LLMs (in their current state

08/07/2024 -- DD/MM/YYYY) to create convex relaxations for problems although they aren't too bad if you use the state of the art LLMs to try and get a linear solution to the problem.

When it comes to optimization for options pricing, I have an article about it but I recommend using libraries for this (py_volllib or black_scholes in Rust).

Genetic algorithms, there is no chance you will be able to get ChatGPT to make for you in their current state so you actually need to learn this one. I know plenty of people who have implemented automated alpha discovery successfully.

None of this is to say that it is useless, but please do reflect on what you expect LLMs to make obsolete, or at least be aware of the fact that LLMs are useful to solve many of these problems nowadays assuming they are quite simple.

Chapter 10 *High Frequency Trading & Market Making*

Here is a large dump of resource for HFT. In terms of textbooks, I recommend High-Frequency Trading: A practical guide to algorithmic strategies and trading systems 2nd edition.

You can also check out any of these textbooks:

- Inside The Black Box
- Dark Pools and High Frequency Trading For Dummies
- Algorithmic and High-Frequency Trading

I don't feel any of the textbooks above, including the one outside of the bullet point list, are going to teach you how to do HFT. That's going to come from playing around with the data, and learning the dynamics. You'll also get a benefit from forecasting and machine learning knowledge. The textbooks really teach you the basics of HFT – not much beyond that.

The Nanex research articles are one of the best resources out there.

See the light reading section, but Flash Boys is amazing for HFT (basically the book that made HFT well known) and so are these books (this overlaps with light reading, but I wanted to highlight):

Dark Pools
Broken Markets
The Problem of HFT
Flash Boys: Not So Fast – This is an insiders review of Flash Boys and is really great
Trading at the speed of light

For learning how market makers function:

Empirical Market Microstructure

Trades Quotes and Prices – Microstructure textbook (REPEAT FROM ML SECTION)

List of links (not made by me):

https://sniperinmahwah.wordpress.com/2014/09/22/hft-in-my-backyard-part-i/

https://sudonull.com/post/93403-Online-Algorithms-in-High-Frequency-Trading-Problems-of-Competition-ITI-Capital-Blog

https://www.youtube.com/watch?v=AS7HLtErlI8

Avellaneda strategy: A technical deep dive

https://hummingbot.io/blog/2021-04-avellaneda-tech-deepdown

A comprehensive guide to Avellaneda & Stoikov's market-making strategy

https://hummingbot.io/blog/2021-04-avellaneda-stoikov-market-making-strategy

https://www.youtube.com/playlist?list=PL2F82ECDF8BB71B0C

https://medium.com/@eliquinox

https://www.youtube.com/watch?v=XgFzHXOk8IQ&list=PLQnljOFTspQUGjfGdg8UvL3D_K9ACL6Qh&index=9

https://alexabosi.wordpress.com/2014/08/28/limit-order-book-implementation-for-low-latency-trading-in-c/

https://github.com/rubik/lobster

https://www.youtube.com/playlist?list=PL5Q2soXY2Zi_FRrloMa2fUYWPGiZUBQo2

https://www.youtube.com/watch?v=_OJmxi4-twY

https://www.youtube.com/watch?v=NmarI5ErisE

https://www.youtube.com/watch?v=9nuAjYRbITQ

https://t.co/Rcnw26Bzyr?amp=1

https://web.archive.org/web/20110219163448/http://howtohft.wordpress.com/2011/02/15/how-to-build-a-fast-limit-order-book/

https://www.guru99.com/os-tutorial.html

https://github.com/theopenstreet/VPIN_HFT

https://github.com/hudson-and-thames/mlfinlab/blob/master/mlfinlab/data_structures/imbalance_data_structures.py

https://drive.google.com/file/d/0B4pk0Nap6TZLNTBhblRHcUJUVmM/view?resourcekey=0-G3T886oNA-ZXtLQE7tKZDA

INFRA:

https://medium.com/prooftrading/proof-engineering-the-algorithmic-trading-platform-b9c2f195433d#a1f7

https://medium.com/prooftrading/selecting-a-database-for-an-algorithmic-trading-system-2d25f9648d02

https://www.linkedin.com/in/silahian/detail/recent-activity/posts/

http://www.caravaggioinbinary.com/HFT-Simulation-Lab/

https://rickyhan.com/jekyll/update/2019/12/22/how-to-simulate-market-microstructure.html

https://youtu.be/b1e4t2k2KJY

https://medium.com/prooftrading/the-trading-strategy-63183bd231cd

https://medium.com/prooftrading

https://mattgosden.medium.com/tutorial-using-pythons-unsync-library-to-make-an-asynchronous-trading-bot-9ee2ae881272

https://www.youtube.com/watch?v=SOTamWNgDKc

Strats and Backtesting:

https://towardsdatascience.com/application-of-gradient-boosting-in-order-book-modeling-3cd5f71575a7

http://jonathankinlay.com/2021/05/machine-learning-based-statistical-arbitrage/

https://letianzj.github.io/cointegration-pairs-trading.html

https://hudsonthames.org/caveats-in-calibrating-the-ou-process/

https://www.youtube.com/playlist?list=PLv-cA-4O3y95J6xmwSaCILL4FlGJZO0PJ

https://teddykoker.com/2019/05/momentum-strategy-from-stocks-on-the-move-in-python/

Programming:

https://www.youtube.com/watch?v=NH1Tta7purM

https://www.youtube.com/watch?v=pBKwWl56uXc

Market Data:

https://cdn.tun.to/minute/

http://www.kibot.com/free_historical_data.aspx

https://www.dukascopy.com/swiss/english/marketwatch/

3rd Edition Extras:

https://www.youtube.com/c/dYdXprotocol/videos

https://medium.com/open-crypto-market-data-initiative/simplified-avellaneda-stoikov-market-making-608b9d437403

https://www.tastytrade.com/shows/geeks-on-parade/episodes/market-making-with-shelly-geeks-2019-07-19-2019

https://quant.stackexchange.com/questions/36073/how-does-one-calibrate-lambda-in-a-avellaneda-stoikov-market-making-problem

http://proceedings.mlr.press/v128/wisniewski20a.html

https://github.com/valeman/awesome-conformal-prediction

https://medium.com/prooftrading/building-a-high-performance-trading-system-in-the-cloud-341db21be100

Algorithmic and Advanced Programming in Python:

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 1 https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3953589

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 2 https://t.co/uq4l7V4oa3

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 3 https://t.co/1ojWwr2SPO

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 4 https://t.co/Su7OZD1SpU

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 6 https://t.co/hJoREZaOYc

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 7 https://t.co/TBs30cXoND

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 8 https://t.co/kGhjQZMBYX

Algorithmic and Advanced Programming in Python - Syllabus in Computer Science, Decision Making & Data - Masterclass 9 https://t.co/FoSCRn4B5V

These will be a few more technically focused textbooks, but they explain some of the latency tricks related to optimizing trading algorithms:

- Professional Automated Trading [REALLY GOOD READ]
- Developing High-Frequency Trading Systems
- Trading Systems: Performance Unleashed

Chapter 11 *Additional Volatility/Derivatives Resources*

This is just one link, but it contains many links within so don't treat it lightly. This is a top recommendation.

https://moontowerquant.com/select-content-from-the-quant-and-vol-community

Options Starter Pack

https://moontowerquant.com/options-starter-pack

Even more of me blatantly copying and pasting Kris's recommendations/ resources, but they're really good

https://moontowerquant.com/moontower-content-by-kris-abdelmessih

Wilmott forum:

I cannot highlight how great of a resource this is. I think it may be one of the only resources where a genuine discussion for pricing exotic derivatives etc can be found. There are some alpha bits in there as well, but in the technical section of the forum there are discussions on models from pricing everything from vanilla European options to Bermudian swaptions, and better yet this is by people who work in the industry.

https://forum.wilmott.com/?_ga=2.79159636.1304736795.1642875642-1589239950.1642875641&_gl=1*3qlrhp*_ga*MTU4OTIzOTk1MC4xNjQyODc1NjQx*_ga_51FRCD57RP*MTY0Mjg3NTYzNy4xLjEuMTY0Mjg3NTY0OS4w

Nuclear Phynance (for some strange reason Wilmott and NP users hate each other) is a great resource. Not so much derivatives only, both NP and Wilmott are diverse, but Wilmott is mainly derivatives. This is an awesome server.

> https://nuclearphynance.com/

Sadly, Nuclear Phynance got deleted. I have left the above in as a memorial to what the site once was. I am sorry it is outdated, but I feel it should be remembered because it was a great site. ThePythonQuant on Twitter is working to recreate it as I understand via discord.

Chapter 12 *Coding Languages Review and Resources*

This is mainly opinion, but for research you should know either R or Python very well (I recommend Python as there are more resources) although when you become more advanced basic Python will not be fast enough so you will need to learn advanced Python and Cython (using C/C++ through Python).

Learning Quantitative Finance with R

High Performance Python

Clean Code In Python

Advanced Python Programming

Expert Python Programming

If you want to do High Frequency Trading (HFT) or Market Making (MM) you will need to learn C/C++ because it is the fastest (in crypto Rust is preferred). Hard to learn, so don't start with it, but rewarding.

In terms of research languages some quants will use R or MATLAB specifically because it has lots of statistical functions that are optimized for data analysis. Ernest Chan loves MATLAB but in reality, it isn't very good so stick with R if you want to learn a 2nd/3rd language for researching math heavy topics specifically. Otherwise, Python should fill the role.

Anaconda (100% free) is an easy way to install R and Python and comes with Jupyter notebook and Spyder (I like Spyder as an IDE although this is a big debate), but Jupyter is without fail the best for working in for research, and both R and Python can be used in it.

If you are hopeless at programming, or just want to go fast then Orange (comes with Anaconda) uses a graphical user interface and requires no programming. It uses scikit learn models which can easily be implemented when you do pick up some code as well (for implementation for example). Really great stuff with loads of models, and you can run your own Python scripts in it if the model is not available. There are external packages to install like time series models and NLP specifically, but the standard library is great as well.

Below I have added some resources for learning algorithmic trading skills in alternative coding languages as well as textbooks for generally becoming good at these languages, but these are quite advanced and are for later in your journey. Focus on the textbooks in the start of the roadmap:

C++ High Performance For Financial Systems

High Performance C++ (Textbook)(ADVANCED)

Machine Learning in C++ YouTube Playlist/Course

Machine Learning with C++ (Textbook)

Modern Data Mining in C++ and CUDA C

Data Mining Algorithms in C++

Testing and tuning market trading systems: algorithms in C++ (Textbook)

These are lecture series on algorithms, data structures, logic etc. These will help you to program better and learn to find solutions that are as fast and efficient as possible:

In my time working in the digital assets industry, I have come to find that Rust is actually far more popular than C/C++, so hence I have decided to include a new section on my Rust programming textbooks:

The Rust Programming Language

Rust For Rustaceans

Rust High Performance

Concurrency with Rust

Rust Network Programming

In my own Rust journey, we were building a cross-exchange spot arbitrage bot and this was basically my first time touching the language. Having only known Python, R, and C++ as my best languages, I jumped straight into coding it. I was literally googling how to do for loops and if conditions in Rust while I coded. I knew there obviously was some syntax for these things as I'd done them in C++, but over the period of a week whilst coding this bot I went from 0 knowledge to being half decent at it. I suggest that readers do the same.

It was a good lesson that I was sitting watching videos about Solidity and one of the devs on my team came up to me and went "why aren't you just coding a smart contract, you'll learn 10x as much" and that is just as true now as it was at the time. In hindsight I already knew this and was being a bit lazy. The truth is that if you simply read these textbooks (and it goes for all textbooks, but especially the programming textbooks, hence why this has it's own learning note instead of going in the 'how to learn this material chapter') you will forget everything you have learned in no time. You need to put the textbook on the desk next to you and treat it as a way of discovering what you don't know you don't know – but you can't actually "learn" from the textbook, only discover that you didn't know something.

Chapter 13 Projects

Many of these projects involve neural networks or complicated methods I do not necessarily recommend taking such routes as it is my view that they are overcomplicating things. Master regressions, and practically minded approaches. These are a list of resources for if you are very focused on deep learning already and want to expand. It is not my suggestion that anybody choose these as their first projects (I include them because my goal here is to include anything relevant).

I have full sections on 5 different types of strategies:

1. Pairs Trading
2. Arbitrage
3. Market Making
4. Momentum
5. Seasonality

These are by no means comprehensive, but they are the areas I have put into production by far the most and have a lot of experience with. Thus, I have written sections specifically on them so that you can get ideas for projects. I feel this is the most practical source of inspiration for those projects.

Subject	Structure	Method	Information	URL
Deep Learning of Small Portfolios for Index Tracking	Deep learning of sparse autoencoder (SAE) models	Topology reshaping techniques which help to identify less complex models that capture the essential structure in the data	1) the sparsification techniques force it during training to generalize better, and 2) the robustification using a heavy-tailed noise model diminishes the effect of outliers	https://www.linkedin.com/pulse/deep-learning-small-portfolios-index-tracking-nikolay-nikolaev/
Algorithmic Cryptocurrency Trading with Sharpe-optimal Deep Learning	Sharpe-optimal Connectionist Learning (SCL), Recurrent Reinforcement Learning (RRL)	Recurring Nonlinear Pattern	directional fluctuations in the given series which are sufficient for sending trading signals.	https://www.linkedin.com/pulse/algorithmic-cryptocurrency-trading-sharpe-optimal-deep-nikolaev/

Universal Cointegration for Pairs Trading via Machine Learning	CCM(kNN) -Cross Covergent Mapping (k cluster Nearest Neighbors)	Estimated Price by Pair Nonlinear Modeling(Estimated Single Asset Price by Nonlinear modeling)	Linear combination produces a stationary residual difference series (also called spread).	https://www.linkedin.com/pulse/universal-cointegration-pairs-trading-via-machine-nikolay-nikolaev/?trackingId=nKswdHj1R%2FagxOErDCoMSg%3D%3D
Local Prediction of Mid-Prices in Limit Order Book Markets	kNN Regression	Arrival Time(kNN Regression)	Order Book Mid Price Arrival time	https://www.linkedin.com/pulse/local-prediction-mid-prices-limit-order-book-markets-nikolay-nikolaev/
Machine Learning with Dynamic Time Patterns for Algorithmic Trading	pdf(kNN), CDF(kNN),Adaptive Shape Distance (ASD) [3], Complexity Invariance Distance (CID) [4], Dynamic Time Warping (DTW)	Nonparametric forecasting of time series using dynamic time patterns	Local Density kNN	https://www.linkedin.com/pulse/machine-learning-dynamic-time-patterns-algorithmic-trading-nikolaev/
Nonparametric Machine Learning for Algorithmic Cryptocurrency Trading	Self-adaptive Local Learning Machine (SLLM)	Adaptive repeating nonlinear patterns by proportional degree of similarity reflecting the essential characteristics of the data trajectory	Essential characteristics of the data trajectory	https://www.linkedin.com/pulse/nonparametric-machine-learning-algorithmic-trading-nikolay-nikolaev/
Overfitting Avoidance in Portfolio Construction using Probabilistic Neural Networks	Multivariate General Regression Neural Network (MGRNN)	Network Architecture	Proportionally density of the effective range (stock spread)	https://www.linkedin.com/pulse/overfitting-avoidance-portfolio-construction-using-neural-nikolaev/
Generative Adversarial Networks for Machine Learning of Constrained Portfolios	Generative Adversarial Network (GAN)	Generator network and a discriminator network that play an adversarial game	Closeness between the utilities obtained with sampled weights	https://www.linkedin.com/pulse/generative-adversarial-networks-machine-learning-nikolay-nikolaev/
Deep Neural Networks for Prediction-based Portfolio Construction	Robust DNN (RDNN)	Nonlinearity	Rectified Units/Gradient Convergence Speed	https://www.linkedin.com/pulse/deep-neural-networks-prediction-based-portfolio-nikolay-nikolaev/
Regime-based Machine Learning of Green Stock Portfolios	Markov Regime-Switching (MRS) model	Volatility based regime switching	Portfolio weights are calculated dependent on the particular regime, that is the portfolio is conditioned on the regimes.	https://www.linkedin.com/pulse/regime-based-machine-learning-green-stock-portfolios-nikolay-nikolaev/
Efficient Direct Reinforcement Learning of Low-risk Portfolios	Connectionist Reinforcement Learning machine for efficient computation of Online Portfolios (CRLOP)	Multi-output feed-forward neural network	Risk-averse investment strategy as a sequence of allocations and re-balancing decisions	https://www.linkedin.com/pulse/efficient-direct-reinforcement-learning-low-risk-nikolay-nikolaev/
Deep Learning of Dynamic Factor Models for Asset Pricing	Deep Dynamic Factor Model (DDFM)	Deep network structure, Neural network pruning technique, using a cardinality parameter to control the degree of sparseness	asset returns from inferred factor realizations, more precisely, the factors are forecasted arrangements of individual asset contributions to the overall portfolio.	https://www.linkedin.com/pulse/deep-learning-dynamic-factor-models-asset-pricing-nikolay-nikolaev/

Support Vector Machine Learning of Sparse Portfolios	epsilon Support Vector Regression (e-SVR) machine	Sparse regression algorithm	Two hyperparameters: the first determines the size of the sensitivity, and the second determines its influence on the magnitude of the fitting error.	https://www.linkedin.com/pulse/support-vector-machine-learning-sparse-portfolios-nikolay-nikolaev/
Online Portfolio Trading by Dynamic Reinforcement Learning	Reinforcement Learning machine for Online Portfolio trading (RLOP)	Recurrent neural network	Momentum and adaptive learning rate for switching between passive and aggressive updating depending on the recent profitability.	https://www.linkedin.com/pulse/online-portfolio-trading-dynamic-reinforcement-nikolay-nikolaev/
Building Smart Beta Portfolios with Large-Scale Machine Learning	Smart Beta portfolios, like the Most Diversified Portfolio (MDP), Risk Parity Portfolio (RPP)	Alternating Direction Method of Multipliers (ADMM)	Statistical characteristics and econometric performanceof the proposed ADMM-based tool for creating Smart Beta portfolios	https://www.linkedin.com/pulse/building-smart-beta-portfolios-large-scale-machine-nikolay-nikolaev/
Bayesian Machine Learning for Robust On-line Portfolio Selection	Bayesian Robust Online Portfolio Selection (BROPS)	Ornstein–Uhlenbeck stochastic differential equation	Heavy-tailed models of returns on prices (based on an approximation of the Student-t density by an infinite mixture of Gaussians)	https://www.linkedin.com/pulse/bayesian-machine-learning-robust-on-line-portfolio-nikolay-nikolaev/
Efficient Computation of Sparse Risk-based Portfolios using Machine Learning	Most Diversified Portfolio (MDP), Equal Risk Contribution (ERC)	Nonlinear programming algorithms	Maximize the ratio between the weighted average volatility of the assets and the total portfolio volatility	https://www.linkedin.com/pulse/efficient-computation-sparse-risk-based-portfolios-using-nikolaev/
Robust Portfolio Optimization via Connectionist Machine Learning	Mean-Variance Portfolio (MVP), Connectionist Optimization Machine (COM)	Quadratic Programming (QP)	Optimized Mean and Minimum STD, Convergence Rate (Speed)	https://www.linkedin.com/pulse/robust-portfolio-optimization-via-connectionist-machine-nikolaev/
Deep Cleaning of Covariance Matrices for Portfolio Allocation	Covariance Matrix	Autoencoder Machine (AEM)	denoised versions of the eigenvectors of the covariance matrix which help to recover its genuine structure	https://www.linkedin.com/pulse/deep-cleaning-covariance-matrices-portfolio-nikolay-nikolaev/
Finding Structure in the Co-movement of Stock Prices via Adaptive Metric	Centroid	K-means algorithm and the Self-Organizing Map (SOM) networks	mean-reverting eigenportfolio with the stocks from each cluster	https://www.linkedin.com/pulse/finding-structure-co-movement-stock-prices-via-metric-nikolaev/
Deep Learning Autoencoders for Building Principal Component Portfolios	Portfolio arbitrage	Principal Component Analysis (PCA) and the Autoencoders (AE), Variational Bayesian inference	eigenportfolio as a linear combination of all stocks which are allocated contributions according to their corresponding coefficients in the first principal component	https://www.linkedin.com/pulse/deep-learning-autoencoders-building-principal-nikolay-nikolaev/

Machine Learning of Heavy-Tailed Dynamic Spread Models for Statistical Arbitrage	Robust Dynamic Mixture Models	Statistical arbitrage (statarb)	mean-reverting spreads based on a discrete version of the Ornstein–Uhlenbeck stochastic differential equation, Dynamic Hedge Ratios	https://www.linkedin.com/pulse/machine-learning-heavy-tailed-dynamic-spread-models-nikolay-nikolaev/
Self-Tuning Local Learning Machines for Prediction of Stock Market Returns	Self-tuning Local Learning Machine (SLLM)	Local Learning machines (LLMs) based on the nearest neighbours approach	autocorrelations, behaviour patterns	https://www.linkedin.com/pulse/self-tuning-local-learning-machines-prediction-stock-market-nikolaev/
Deep Learning of Heteroskedastic Volatility Models for Risk Estimation	Dynamic heteroskedastic volatility models (D2GARCH)	Deep connectionist structure by unrolling the network in time.	Volatility rank and probability, time relationships in the data	https://www.linkedin.com/pulse/deep-learning-heteroskedastic-volatility-models-risk-nikolay-nikolaev/
Deep-Memory Networks vs. Deep Learning Networks for Stock Market Prediction	Deep-Memory Neural Networks (DMNN)	Dynamical system using sequential Bayesian estimation to accommodate properly the temporal dimension.	Feature Extraction -> long-term dependencies via memory feedbacks and gates for behavioural control	https://www.linkedin.com/pulse/deep-memory-networks-vs-deep-learning-stock-market-nikolay-nikolaev/

Chapter 14 *Data*

Data is typically not the easiest to come by. There are 3 different sources of data that are free you can find on the internet:

1. Data dumps from people who have paid for it
2. Free data directly from the exchange (popular in crypto)
3. Competition data

There's also Yahoo Finance which I would consider an exception, and of course there are ways to get data via trial and web scraping sites like investing dot com, yahoo finance, etc.

Before we get started, here are some articles about data (cleaning, sourcing, and pre-processing):

- QR1: Demons Lurk In Your Data by 0xfdf
- Quant Arb's Data Sourcing Guide
- Quant Arb's Data Pre-Processing Guide

First, I will start with a list of providers – this is not necessarily my top sources for budget, but simply an overview of who is considered the go-tos. For crypto this list is (heavily based on systematicls' tweet which was community driven with modifications of my own):

Tick Data:

1. Tardis (majority + preferential over Kaiko)
2. Kaiko (majority)
3. LO Tech (founded by @TimMeggs)

I have not tried LO Tech, but Tim has a strong reputation in the community, worth checking out. Tardis is generally accepted as the default standard. It is vastly cheaper and easier to use than

Kaiko – although still in the tens and tens of thousands at the institutional level. Kaiko has much better coverage than Tardis, so that's mostly the niche that is filled with it.

Blockchain:

1. Dune (rec by @spxudi)
2. Glassnote (rec by @quant_arb)
3. Santiment (rec by @quant_arb)
4. Arkham (rec by @spxudi)
5. IntoTheBlock (rec by @friendscallmeap)
6. Flipside (rec by @friendscallmeap)
7. The Tie (rec by @sersabr)

News:

1. Ravenpack (majority)
2. Velo (majority)

You can also get news data from Tiingo if you want it to be quite cheap, and you can get sentiment data that is similarly priced to Ravenpack from Alexandria research.

OHLCV data can be acquired directly from the exchanges in digital assets.

Tiingo - $30/mo. Cheap, but the data is only fundamental, EOD, minute data, and news data. There are some issues with the crypto data because it is aggregated cross-exchange so the highs and lows and extreme relative to what you would normally see. Live data as well.

Polygon – a bit more than Tiingo, but has tons of coverage and a lot more different types of data. Still priced in a very retail friendly way.

Binance – FREE. Only crypto data and the historical data is of course limited to aggregates historically and only a brief window if you want historical sub-minute data. But I have scraped quotes so ask and I can provide.

IBKR - $10-30. Super cheap and you can basically get all the data you could ever want out of it but you need an account and $500 deposited with them since they're a broker (my broker recommendation btw). The API is slow as shit so if you want to download their entire options data library you better make a scraper in AWS Cloud because it will literally take over a month. (AWS SageMaker Jupyter notebooks are an easy way to scrape data without needing to set up servers using anything technical, this method can also be done for live hosting trading algorithms)

Links: (Over a TB of data, worth a fortune, but handed out here for free!)

Numerai uses community sources alpha for running it's fund and gives out loads of free data

https://numer.ai/

G-Research Crypto gives out data as well, but don't submit code. Read the legal docs you are giving them it basically. Blatant code grab. (God this data is awful. I think it was broken on purpose to make the challenge harder)

https://www.kaggle.com/c/g-research-crypto-forecasting/data

22GB of IQFeed Data

https://mega.nz/folder/HUQzDCgK#rc45NqXhRA8SFgK1l2MYcw

1.2GB of CompuStat data.

https://mega.nz/file/6IwnQKQL#Xb1PQja8veVCRWy7nJ_o45ZKeDyDy4lYV7QAHQnv7A4

Cryptocurrency data for Bittmex

https://www.kaggle.com/tencars/392-crypto-currency-pairs-at-minute-resolution

3k stocks, decade+ of financial data

https://www.kaggle.com/miguelaenlle/parsed-sec-10q-filings-since-2006

More data. Also not great, but hey anything helps, and it can be fixed of course

https://www.kaggle.com/c/ubiquant-market-prediction/data

Loads of NLP data. Headlines and analyst reports mainly

https://www.kaggle.com/miguelaenlle/massive-stock-news-analysis-db-for-nlpbacktests

Google search data

https://www.kaggle.com/miguelaenlle/google-trends-history-for-4000-stocks

Optiver IV prediction data

https://www.kaggle.com/c/optiver-realized-volatility-prediction/data

Chapter 15 GitHub Repositories

One of the best ways to find good examples are on GitHub and one of the best repositories for algorithmic trading is the one that accompanies the machine learning for algorithmic trading textbook referenced earlier.

https://github.com/stefan-jansen/machine-learning-for-trading

Barter is a Github made by someone over at Keyrock, which is a market making firm in the digital assets space. It's become a bit dry, but was the initial inspiration for one of the core trading libraries I use at work. Worth having a read through to get a better idea of how things should be structured.

https://github.com/barter-rs/barter-rs

Another repository that comes from a textbook is the repository that comes from Mastering Python for Finance. The models in the textbook are quite good especially LSTAR models which aren't usually in time series courses, but are great models.

https://github.com/jamesmawm/Mastering-Python-for-Finance-source-codes

This is a github repository that links to other notebooks and has quantitative resources in itself. It has a lot more risk models/ pricing models especially compared to the first repository referenced which is purely about generating alpha, and still has loads of purely alpha based model so is loaded with resources.

https://github.com/letianzj/QuantResearch

This github provides some basic examples toward applying signal processing in Python which can be used as features in the feature engineering process rather successfully.

https://github.com/SparkAbhi/SignalProcessingWithPython

This is an interesting project that applies one of the most important pairs trading papers in the literature in a detailed manner with up to date code examples as well. The use of PCA to generate multivariate portfolios for both portfolio optimization and mean reversion trading is a key advancement here.

https://github.com/alexdai186/Eigenportfolios

Generating features/ finding examples of great features to use is always a good thing to have so the next two repositories give great examples of basic feature engineering. The second one doesn't make as good features as these are only basic price features, but it shows how to use PySpark (I personally prefer Dask – it's more effective) which is used for big data applications such as with HFT data (a couple months of quote data can be 50GB compressed -> ½ TB uncompressed and 50TB if you engineer 500 features, so distributed computing is needed!)

https://github.com/hjeffreywang/Stock_feature_engineering/blob/master/Feature_generation.ipynb

https://github.com/MiaDor12/Advanced_Feature_Engineering_of_Raw_Data_of_Stocks-with_PySpark/blob/master/Advanced%20feature%20engineering%20with%20pyspark%20on%20raw%20data%20of%20stocks.ipynb

Here is another example of good feature engineering and the use of fractional differencing to make the data stationary which then lets you use models that assume stationarity such as FFT (Fast Fourier Transform) although there are non-stationary signal processing models as well. More details are in a thread I wrote on this subject on twitter

https://github.com/alexbotsula/Price_direction_forecast

Here is a github full of microstructural models for high frequency trading (the next few repos will be HFT/MM).

https://github.com/gjimzhou/MTH9879-Market-Microstructure-Models

VPIN is an important model for market making and is one of the latest models in the literature so here is an implementation repository.

https://github.com/jheusser/vpin

High frequency trading statistical arbitrage example github repository.

https://github.com/clfrenchgit/gdax-bot

HFT using DL models for statistical arbitrage.

https://github.com/scibrokes/real-time-fxcm

This one isn't quite github but is a great resource for finding example C++ code for developing low latency C++ systems.

http://dlib.net/

A great example of C++ HFT MM algorithms. An improvement idea I have suggested to the author but can also be attempted by interested algotraders is that a fast model like XGBOOST (there is a C++ library) is used alongside some alphas to make spreads asymmetric before traders can trade against you and you get negative edge in those trades. A large part of market making is cheaply executing alphas by trying to get inventory on the side of your predictions and also by getting out the way of adverse conditions by making your spreads asymmetrically wide (traders with alpha against you).

https://github.com/hello2all/gamma-ray

Massive dump of mainly papers and some textbooks!

https://mega.nz/folder/g90whIDK#8f2uZESbHFTBEzG-0udqrA

These are more just other people's resources, but they are all in drives and I enjoyed most of them, but obviously I prefer my own resources as I have vetted them more.

https://github.com/beimingmaster/quant-resources

This is a GitHub repository written by @BeatzXBT on Twitter. It's full of tons of great market making content. Well worth checking out!

https://github.com/beatzxbt

Chapter 16 *Light Reading*

This section includes books that are not quite textbooks but build a general knowledge of how the industry works. This is good for showing you know the industry well in interviews, and just really great common sense in modelling.

Liars Poker

Flash Boys (This and Dark Pools are a great resource of understanding HFT)

Irrational Exuberance

The intelligent investor

Pragmatic capitalism (there is a great list of books to read at the end of the book)

The black swan – tail risk

Fooled by randomness – tail risk/ understanding randomness

The quants (Scott Patterson) (so much common sense, and lessons in this book)

Dark Pools (Scott Patterson)

More money than god – A full history of hedge funds

Black edge – Steve Cohen Biography. What a legend

The man who solved the markets – Great book but here is a summary of the lessons from a friend (but do read the book as well, it's interesting)

A man for All Markets: Beating the Odds, from Las Vegas to Wall Street

The (Mis)Behaviour of Markets: A Fractal View of Risk, Ruin, and Reward

When Genius Failed – A very important lesson emphasized by the quants

A Random Walk Down Wall Street

Billion Dollar Whale

The Predators' Ball

Broken Markets

The Problem of HFT

Flash Boys: Not So Fast – This is an insiders review of Flash Boys and is really great

Trading at the speed of light

Investing for adults – Easily the greatest book for passive investing (It's a mini-series, but basically as long as a single book)

The rise of carry

Bombardiers - Po Bronson

FTSE: The Inside Story

Flash Crash: A Trading Savant, a Global Manhunt and the Most Mysterious Market Crash in History

Chapter 17 *Careers*

The usual career path:

1. Attend Target University (this means top university)
2. President of Finance Club / Student Consultancy / Interest Shown
3. University Internships
4. Spend at minimum 3 months memorizing every question in a quant interview question textbook and learning every brain teaser you can find.
5. Cold emails/ coffee chats/ networking
6. Get a job at a prop shop / hedge fund / investment bank
7. Move to a smaller shop and run risk

Here is a very useful article I wrote about the pathways to running risk in a lot more details:

https://x.com/quant_arb/status/1801233401136992756

The first thing to make this better is to get a top internship in high school or early university. This is hard, and by no means necessary but it certainly helps you along and differentiates you. Resources are provided below. Once an internship is obtained it is referred to as a conveyor belt because it becomes far easier to get another. 85% of those with internships come back. When you receive a summer internship do not wear a Rolex, Gucci sleds, etc. You are there to wear a plain Casio, always leave later than your boss, and get that return offer.

Read this thread by Rich Handler for some great internship advice.

One resource is Wall Street Oasis. This is very heavy on investment bankers and finance focused individuals, but there is certainly room for quant. r/quant on reddit has a good few career posts on there, worth checking out.

Watch Alpesh Patel on Tiktok (@greatinvestments) and check out the “internship” he offers. They are a 100m+ fund and offer an open internship. It’s basically a course and you won’t learn much, but anyone can do it and it will +1000 points your resume.

Same deal with “theforage.com”. You can get virtual GS, and JPMC internships (no application/ rejection it’s open to all) which aren’t really actual ones, so I do recommend being careful with this, but anything to bolster the resume can help until you’ve worked in some real roles, but make sure to be transparent.

Here is a YouTube channel all about investment banking and hedge funds/ private equity. This is coming from the non-quantitative side of things and probably refers to Macro or ELS (Equity Long/Short) funds more than quant funds and is also more M&A (Mergers and Acquisitions) than S&T (Sales and Trading), but that doesn’t really matter because the recruiting processes are basically the same. Some really good videos about coffee chats. The fact is that you will be sending 1000s of cold emails/LinkedIn messages. Kris Sidial would follow senior people to work and give them his resume, and he’s doing very well now. Don’t be embarrassed to do this because otherwise someone else will. The competition is massive so the process for recruiting is ruthless. The usual process is:

- Cold email/LinkedIn/twitter message (Be thoughtful there are guides in the YT channel for all of these parts btw)
- Attempt to get a call. This is your chance to shine and why you’ve been reading piles of textbooks. You need to be impressive because otherwise they won’t want the next step
- Coffee chat. Try and get an in person meeting. You should be subtle about it but once it is going well ask about internships. The entire product of all this effort is to get a single comment from them to HR in the break room along the lines of “If you see an application from xx, he really knows what he’s talking about and is really interested”. https://www.youtube.com/c/PeakFrameworks

Another tip I will give is that showing that you can learn fast and are willing to put in those hours to get there is just as important as knowing what you are talking about.

A great article about careers from some top industry characters. I will highlight a quote that I really took away from it and personally agree with. It is a lot easier to teach a mathematician

to trade than a trader to solve PDEs. The math you learn in your degree is a way of thinking as much as it is useful

https://notion.moontowermeta.com/career-advice

For those looking to learn derivatives and volatility and work on an exotics desk at an investment bank (one of the best ways to learn), in addition to the resources posted earlier Benn Eifert on Twitter often posts interview questions that can't be found elsewhere.

Here is a good channel for careers for extra.

https://www.youtube.com/c/DimitriBianco

Chapter 18 *Arbitrage Guide*

This chapter will mostly focus on arbitrage in digital assets, as this is where I am most familiar with the topic. There are many different pieces of knowledge involved. Some are more general like how to use limit orders to improve a strategy (and this relates heavily to market making), and then there are components that are specific to each trade.

I've talked about the specifics of each trade before in these articles:

1. Funding Arbitrage [part 1]
2. Triangular Arbitrage [link]
3. Spot Arbitrage [part 1] [part 2]
4. Perpetuals Arbitrage [link]
5. Event Arbitrage [Substack] [Twitter]

I've written about how to optimize the latency component here, and I've also talked about execution components here, but also as part of a general article on how to improve arbitrages from a higher level perspective (improvements that apply to all arbitrages)

For funding arbitrages, there are these websites to scan for them (but I recommend you build your own):

- Bybitpremiums [link]
- Bybitpremiums Liquidity Goblin [link]
- Coinglass [link] [link2]
- Crypto and Carry [link]
- Crypto Funding Tracker [link]
- Coinalyze [link]

Here are some alternative blog articles on funding rate arbitrage:

1. https://blog.biqutex.com/funding-rate-arbitrage/
2. https://medium.com/@Xulian0x/mastering-funding-rate-arbitrage-in-crypto-a-comprehensive-guide-27b4c3bb0f90
3. https://www.binance.com/en/support/faq/what-is-the-binance-funding-rate-arbitrage-bot-and-how-does-it-work-f330e17d6fc04679b9b21d6f9350e787
4. https://blog.amberdata.io/the-ultimate-guide-to-funding-rate-arbitrage-amberdata
5. https://learn.bybit.com/bybit-guide/bybit-funding-fee-arbitrage/
6. https://academy.synfutures.com/funding-rate-arbitrage-in-crypto-exchanges-opportunities-and-risks/
7. https://docs.trade.polynomial.fi/strategies-and-tools/funding-rate-arbitrage-101
8. https://medium.com/@blex_education/funding-rate-arbitrage-guide-59d4878539ba

For cross-exchange spot arbitrages, here are some scanner sites, but again I recommend you code up your own because many exchanges have wash flow or the data can be inaccurate. It's very important that you are able to control the data yourself to ensure this. These are great for finding new exchanges or opportunities to add new features:

1. Free Crypto Arbitrage [link]
2. Xpher [link]

Chapter 19 *Market Making Guide*

Market making isn't the easiest way to make money – that's for sure, but it scales a lot better than arbitrage strategies, and you don't have to worry about the trade eventually dying out. You can still have your lunch eaten, and markets can still get more competitive over time, but with arbitrage you know there will be a day when you can no longer compete, and you need to constantly think about growing into a new trade.

For many people, that new trade is market making. They begin making into the arbitrage to try and get a leg-up on their competitors and improve their fills, and in no time they are market making. This is a common path I tend to hear about and have experienced it myself.

Now, that I've talked a bit about how people end up doing it – let's get down to what matters. Your priorities are as follows:

1. Edge
2. Spreads
3. Risk

Edge manifests itself via your ability to accurately forecast mid-price, and to react to events with low enough latency. If you consider yourself more of a statistical person, and don't know what you are doing on the latency front – either prepare to learn or move up into the minute frequency because you need to optimize latency at some point in the trade. That or pick an absurdly inefficient market.

Spreads are about how wide you are. It's not so much how wide you are on average, that's actually quite easy to tune. Say I want to be X% of the volume in this asset, I tune my spreads until I am. You can also tune off PNL, but that's a lot noisier so that component that tunes your spread should focus over a longer period of time, with a much shorter tuning based around the volume of the asset.

A starting point of reference for your spreads can be the EWMA of the spread width over time. This will put you in a bad position if spreads blow out, so your next step is figuring out when this is wrong – i.e. when this is the worst advice you've ever heard.

Economic events are an obvious time when you may not want to be quoting. There's going to be a brief few seconds where everyone who trades are the people who have just got the event data before you could and are now on a mission to eat your lunch, but prior to the event… go-ahead, at that point it's just retail goons who want to bet on CPI. That is unless you suspect information leakage… there's a Nanex article on that, which is one of many reasons I've put it in the HFT resources.

Now, on to risk. This is the part that gets focused on the most by everyone. It's the reason you see these complicated equations to balance your inventory, but in reality it's not a great idea to do that. Those correlations you see in your models don't necessarily hold and they can often be a reason for your algorithm to take on tons of *toxic* inventory because it believes it's fully hedged against another asset. In this regard, you get adverse filled against when this correlation does not hold. You get adverse filled against damn near anything you can be adverse filled on in all honesty.

Going back to edge. How long should my forecast horizon be? Well, it's based on how long you expect to hold. That's the period you care about after all. That said, we can see a pretty exponential decay in the level of signal once filled when measuring adversity, and you probably wouldn't want to get a fill like that to begin with. That said, if you are extremely fast and only care about the ultra-short-term then forecast out that far, because that's where you have the best forecasting edge anyways. The same rule is a bit iffier when it comes to whether you should just ignore adversity if you plan on holding inventory for longer – at that point you end up thinking about taking, and can treat the adversity as a trading cost.

These are just my thoughts afterall, but I think it's a half decent run through on the basics of it all. Keep refining your system with new insights – it's all quite mechanical and clear afterall, and eventually you'll make money.

Chapter 20 *Pairs Trading Guide*

Pairs trading is an ever evolving field, to start let's go through some articles. I think @systematicls on Twitter has one of the best articles out there where he implements a statistical arbitrage strategy. I also have a section of my blog with many articles on there related to pairs trading.

Some resources:

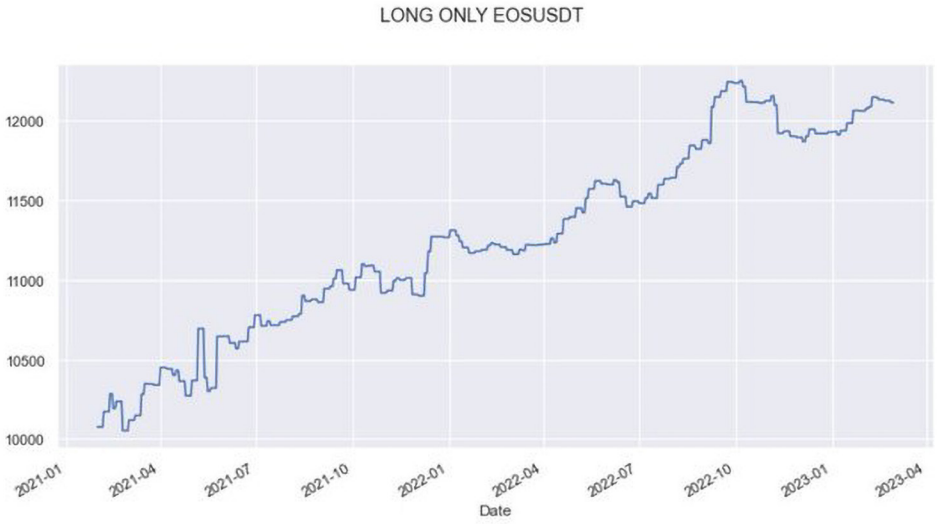
- Article on eigenportfolios.
- Github full of pairs trading strategies
- Great article by liquidity goblin on eyeballing spread
- Articles by H&T
- Guide by H&T
- Ready-to-run strategies on QC (free):
  - https://www.quantconnect.com/research/15298/pairs-trading-copula-vs-cointegration/p1
  - https://www.quantconnect.com/research/15347/intraday-dynamic-pairs-trading-using-correlation-and-cointegration-approach/p1
  - https://www.quantconnect.com/research/15300/pairs-trading-with-stocks/p1
  - https://www.quantconnect.com/research/15299/pairs-trading-with-country-etfs/p1
  - https://www.quantconnect.com/research/15355/mean-reversion-statistical-arbitrage-strategy-in-stocks/p1

Chapter 21 *Seasonality Guide*

On the blog I have a full overview of seasonality strategies already so I've linked that below:

https://www.algos.org/p/seasonality-a-comprehensive-overview

I also co-wrote a paper on a strategy that generalizes seasonality with HangukQuant. Here's the PNL curve when applied to crypto markets (curve is smoother when trading many coins):



https://www.algos.org/p/seasonality-in-commodities-markets

Chapter 22 *Momentum Guide*

I wrote this a while ago on my blog so I won't re-write it, but I will link to resources.

https://www.algos.org/p/breaking-down-momentum-strategies

Chapter 23 *Blogs To Read*

Blogs are often a great place to find novel ideas and information that is very practically minded compared to academic literature. If you exclusively read papers, you'll get bogged down in academics and emerge with the same habits as them – overcomplicated methodologies that lead to precisely wrong instead of roughly right answers.

- The Quant Stack [link]
- The Quant Playbook [link]
- HangukQuant [link]
- VertoxQuant [link]
- Diary of a Quant [link]
- Low Latency Trading Insights [link]
- TaiwanQuant [link]
- Algo Trading & AI [link]
- Zero Cost Abstractions [link]
- EntropyChase [link]
- Liquidity Goblin [link]
- Quantocracy [link]
- Ed West [link]
- Quant Trading Rules [link]
- Alpha Architect [link]
- Quantpedia [link]
- OSM [link]
- Portfolio Optimizer [link]
- Mark Best [link]
- Dekalog Blog [link-1] [link-2]
- Finomial [link]
- Alvarez Quant Trading [link]
- Allocate Smartly [link]
- Robot Wealth [link]
- Quantifiable Edges [link]
- Quant Insti [link]
- Milton FMR [link]
- Quant Start [link]
- Chase The Devil [link]
- Clarus FT [link]
- Quants R Us [link]
- Quantlib [link]
- Implementing Quantlib [link]
- HPC Quantlib [link]
- Oxford Strat [link]
- r/quant on reddit
- Nanex [link]
- Parasec [link]
- Ricky Han [link]
- Quant Stack Exchange [link]
- Max Dama (really good) [link]

Chapter 24 *Twitter Accounts To Follow*

Twitter is one of the best resources out the for information, so short of my own handle @quant_arb, I figured I would list off some accounts that I feel are worth following.

There will be a lot of accounts below, there is no particular order and my inclusion is in no way a recommendation of their content as high quality. It's a rough guess that based on what I've seen (which may be a couple tweets, it may be almost all of their tweets) that it's worth staying tuned into. There are people on this list I think sometimes produce content that isn't the best, but they are included because they have produced great content on other occasions.

To me being worth following simply means there is a reasonable expectation they may provide some value to your feed at some point in the future. That said, many of these consistently output bangers. But yes, some of these accounts may be LARPs so please diversify who you listen to for there is no true messiah.

And yes, there are a lot of accounts, but I'm sure the algorithm will filter out the bad ones for you regardless. It's a busy list, and I'm sure a couple will be less than useful, but many are great!

1. Vertox_DF
2. Ninjaquant_
3. BeatzXBT
4. Gumby_arc
5. Dub0x3A
6. stalequant
7. LiquidityGoblin
8. 0xfdf
9. Systematicls
10. __paleologo
11. SeattleISmith
12. TaiwanQuant
13. Oxbquant
14. PtrPomorski
15. MoonDevOnYT
16. 0xLoris
17. Gametheorizing
18. Quantseeker
19. Globalflows
20. QuantumHoneybee
21. CMTDIntern
22. Mevintern
23. toxic_intern
24. TimMeggs
25. LilQwantXBT
26. GryptoGoon
27. QuantFiction
28. Idro___
29. Predict_addict
30. ScottPh77711570
31. OnlyDeFiGuy
32. Larpcapitalwc
33. Citrini7
34. Super_macro
35. EntropyChase
36. GoblinCap_

37. OxLouisT
38. Yenwod_
39. Quantaraum
40. Nik_algo
41. KrisMachowski
42. BlackSwan_ptf
43. DegenQuant
44. DrApeLincoln
45. CheifGolem
46. InvestingIdiocy
47. HighFreqAsuka
48. Quantymacro
49. Phoenixstealthy
50. EquidityCapital
51. NikitaAFadeev
52. TradingTrotter
53. Benjaminwfelix
54. Valkmit
55. OrthogonalAlpha
56. Sasuke___420
57. Just_a_stream
58. Worstcontrarian
59. 0xTurdleTrader
60. PenguinsTrading
61. Josusanmartin
62. Dfauchier
63. Macrocharts
64. Louis_hyper
65. Perfiliev
66. Ltrd_
67. Mikevanrossum
68. AgustinLebron3
69. Crypto_hades
70. Mgnr_io
71. Alex_bcg
72. Robot_wealth
73. GautierMarti1
74. Hughesanalytics
75. Eigensteve
76. IDrawCharts
77. Technicallyrain
78. Bookdepth
79. Quantpedia
80. cryptoquantHQ
81. nadogoyard
82. weaponizedFOMO
83. folkvangtrading
84. bahamastrading
85. sisSoftware
86. 0xLightcycle
87. 0xdoug
88. Robertmartin88
89. Wifeyalpha
90. Macroalf
91. Mikeharrisny
92. Carlcarrie
93. NotoriousLOB
94. HangukQuant
95. Minimalexcess
96. FabiusMercurius
97. Macro_synergy
98. EmanualDerman
99. Quantian1
100. Ryxcommar
101. Quantrob
102. Shubham_quant
103. QTRResearch
104. Nntaleb
105. Squeezemetrics
106. SinclairEuan
107. Quantocracy
108. Ksidiii
109. HariPKrishnan2
110. FadingRallies
111. Macropotamus
112. Conejocapital
113. Dirty_swan
114. Choffstein
115. VolatilityVIX
116. QuantVol
117. Volmagorov
118. Salr_nyc
119. Krisabdelmessih
120. Therobotjames
121. FinancePirates
122. Timebargains
123. QuantSymplectic
124. CapitalParadox
125. Macrocephalopod
126. ShedCapitalLLC
127. Nope_its_lily
128. Quantkaks
129. Lightspringfox
130. Pyquantnews
131. Christinaqi
132. goshawktrades

Chapter 25 *How To Learn This Material*

In my view, this chapter is as important as any other chapter in this document. It is the multiplier that will be applied to everything you learn. If you have a terrible method of learning, then it will be 0. I will not be talking about bullshit focus tricks or giving a ted talk. Just talking about what is useful and what is not.

Firstly, you won't read all of this content. There's no way and even if you do manage it then you have not spent your time well. You should be constantly filtering down, finding what's most relevant to you, and most importantly SKIMMING. Most chapters make their points quickly, same with most books. A lot of books – especially non-textbook ones, make a couple of points and then spend the rest of the book rambling. Go read the black swan for an example of a petit philosopher with a couple great points nestled in between there.

When it comes to filtering, you can often get the PDF online, through various means. Avoid libgen for free PDFs and make sure not to accidentally go to sci-hub (any of it's endings, .se, .st, .ru, etc) for free papers. Skimming PDFs and the contents is a nice and fast way to get an idea of if a book is relevant to you or not.

Now, the most important part – what you do with this knowledge. You should implement it, talk about it, write your notes about it, hey, maybe even tweet about it. Any way you can find to ensure that this information doesn't go in one ear and out the other.

A lot of material (papers, textbooks, blogs) are useless, and worse they're often not accurate. You need to be implementing these things and seeing if they really would make money in production. Live PNL, calculated of your own doing, and not an overfit backtest they've sneakily presented to you, is the only way to truly judge a strategy, but you can at least do your own research.

Not only will you build practical research skills (which frankly toying around with Pandas and knowing your way about the research process is half the knowledge anyways), you'll learn the right knowledge by building your intuition for what actually looks valuable to you.

I cannot stress this enough – if you do not approach everything with a goal of making PNL as fast as possible, then you will get nowhere. Even Citadel asks this of it's researchers. How can I make money today? @TheRobotJames talks about this a lot, and has many great intuitive explainers on general advice, but my advice to you is to follow the money. Aiming to make real PNL keeps you away from nerd holes. It keeps you from spending months building a neural network only to realize it doesn't work. You should've figured this out days ago, but you decided to spend 3 weeks building a fast backtester for it in Rust first that only really works with this algorithm specifically.

Avoid building too much tooling or vanity projects. I have had this conversation with many senior quants and we all complain about the same thing junior researchers do. They all want to build things! Stop trying to build a pretty dashboard, cool backtester, or any of this stuff unless you have needed it multiple times in a row and know for certain that given the amount it's *already* happened it's worth implementing. DO NOT BUILD UNLESS IT IS NECESSARY. So many researchers get into this terrible hole of wanting to build things and doing it on company time. If you want to toy around with Rust, but honestly should be using Python – go build in Python and build the cool Rust project on your own time. This is maybe a professional complaint in some ways, but it's good advice. Those ideas that sound fun to work on aren't always what you should be working on. Sometimes it's grinding out simple CTA strategies and mastering plain old regressions.

The glitzy neural networks and machine learning projects fall under this as well. If you are well and truly excited about a project – whilst this will make you very motivated and I don't dare to discount that, you need to also think about whether this is what you *should* be working on or is some interesting fun little model / project. Is it complicated or involves building things that don't immediately make money… especially if you aren't a dev? If yes, maybe reconsider.

Also, nothing has to be read in full. I don't read anything in full – certainly not papers, and don't tick anything off as read. You'll forget great wisdoms overtime or perhaps you read it at the start of your career and didn't fully appreciate some of the advice because it was boring, and about linear regressions (and all you wanted to hear about was cool fancy models). Re-reading

what you mentally have ticked off as having read is not always such a bad idea, or re-skimming rather.

I've seen fancy models work, but from my own experiences – they take a damn long while of mastering that specific niche to really start making money with. Took me about 2 years with pairs trading of non-stop hammering out hundreds of notebooks (yes, hundreds and hundreds) before I felt I truly had some theories that gave me a leg up on people in terms of understanding the space.

Don't write notes for the sake of notes. If you find yourself looking for notes to write, and not writing them out because you are truly inspired by what you have read then don't bother. You won't remember it after the fact, and you certainly won't be doing much more than copy pasting. If you read something, think oh wow that's really interesting, and then write the notes – especially in absence of the actual material next to you (so it's just your memory of the idea) then those notes start to be useful.

I write my blog articles and tweets often from ideas I've had in conversations, the work I'm doing, the material I've read, and even day to day inspirations. They don't come from reading chapter 4 of a textbook and regurgitating it because there's nothing of value there – especially not to you and for your own thought development. Notes must come from your own interpretation and understanding of the material and not from blatant copying of it- that is the only time notetaking shines as a source of valuable learning.

Some take notes and re-read them, I sometimes re-read my notes, but I do most of my learning when I actually write them instead because it forces me to neatly organize my ideas. It's a great channel for cleaning up your understanding on the topic. Nowadays with ChatGPT, there's no use in summarizing texts in your notes, the AI can do that for you.

When you catch yourself tying concepts together in ways that weren't described in the material and making those connections, that's when your note taking is working. Same with the research. When you have an idea, and you start pulling on your knowledge bank of "hey I saw this effect in commodities, I wonder if it works in crypto", and then "I think funding rates are a big factor in crypto… hey what if instead of using volume + momentum like in commodities, I used funding rates and momentum", and suddenly you are making your own strategies. They may suck – in fact, they usually do (you won't realize this at the start though because you'll be too busy

overfitting and looking ahead, but you'll come around), but it's a learning process, and as you develop your theories and ideas of what works, you'll eventually get there.

It's been quite a long chapter, a text heavy one, but I think it's important to distill what I've learned for the next generation of quants and I hope this section will be as treasured as the rest of the book because it's as important in my view.

Chapter 26 *Other Roadmaps*

- Vertox's Resource List
- Moontower's Resource List (Best our there for options)
- Moontower's Blogs & Online Writers
- Options Starter Pack
- QM's Textbooks
- QuantGuide (Paid, but good for interview prep – there's also plenty of textbooks for this like Heard on The Street)

Old axioma papers are worth reading, if you can find them. Same with the old GS quant publications. Sell-side research tends to be quite high quality if you get it from banks, but of course the research is not easy to get a hold of.

Credits

If you received this document, you probably have my contacts anyways so feel free to ask questions, but I also have a few worthwhile threads on twitter at:

https://twitter.com/quant_arb

-Quant Arb / BBM

Big thanks to the rest of the editorial team:

Rayyy, BlueCrab, DegenQuant, & TKShadow

And to all of those who sent in their contributions!