Machine Learning and Algorithmic Trading of a Mean-Reversion Strategy from the Cloud for Liquid ETFs on Robinhood

Final Report for EAS 499 Senior Capstone Thesis (CIS-ASCS)

Student Name: Fan Zhang

Project Advisor: Professor Abraham Wyner
Department: Statistics

April 25, 2018

Abstract

This is an application of machine learning to (extremely noisy, non-i.i.d.) financial multivariate time series across a variety of asset classes. We lay out a data processing pipeline end-to-end, from sourcing and preprocessing to analysis and system design. The focus centers on exploratory data analysis as well as a regression task followed by an empirical, problem-specific comparison of algorithms and hyperparameters sets within our framework. Out-of-sample performance evaluation is based on two metrics, and eventually along with an active trading strategy, incorporates trading costs in dollar terms. We will outline the design of an algorithmic trading system which could be deployed remotely.

Contents

1 Introduction

1.1 Motivation

1.1.1 Market Efficiency Test

According to the Efficient Market Hypothesis (EMH), financial market prices reflect all publicly available trading information so one cannot consistently beat markets (especially in deep, liquid securities) on a risk-adjusted basis over the long run [33]. The EMH's weak form asserts that this is impossible from only price and volume data (although other sources may provide such opportunities). Sources of criticism for EMH include historical irrational asset bubbles often followed by crashes (of which the 2008 financial crisis and massive surge in cryptocurrencies serve as recent examples) [13], as well as various claims of abnormal returns by individual traders and investment funds (for instance, with stocks' price-to-earnings ratios as a leading indicator of their returns, and decades of outperformance by quantitative hedge funds) [8].

To discover of a potentially profitable trading strategy on paper is far from conclusive proof against even a single instance of market inefficiency can be exploited in practice. There are inevitably barriers in execution speed and trading costs [6]. A regime that exists for any period of time may change in a structural break later due to economic, political, legal, or sociocultural reasons that affect the behavior of market participants. And in analyses involving data mining (whether quantitative in formalities, grounded in the discretionary and qualitative assessment of "experts", or essentially arising from anecdotes) there is always the possibility of overconfidence in unstable patterns and trends. So the usual disclaimers about past performance being not necessarily indicative of future results applies to our work.

Nonetheless, that should not preclude us from making an honest data scientific attempt. For, to pursue signals with a multitude of approaches with varying degrees of successfulness adds to the experience that a detective carries forward in future inquiries (as hunches are frequently made here and there, minus the occasional brute-force exploration). That would naturally imply the usefulness of negative findings, not in immediately disposing of the EMH question in the efficient affirmative, but to point out that a particular method in its exact form does not solve the problem at present.

1.1.2 Curse of Dimensionality

Along the levels of importance in the financial modelling process, the most valuable is data of high quality and relevance (which in itself can be a make-or-break matter), transformed into engineered features (either manually crafted by hand, or automatically generated possibly with some degree of filtering), and finally algorithms that produce models fitted upon the features [9]. In this field of application, it is absolutely critical to have good data, fairly useful to produce appropriate features from them, and nice but not paramount whether one draws from one generator of models or another (among the popularly celebrated tools that generally work well out-of-box).

The combinations of the aforementioned multiply to great numbers, and in this project we will undertake the thrift-constrained challenge of trying to capitalize upon only public data from the Internet (albeit, of the highest calibre freely available and known to us) in the equalized democratic spirit, but seeing what can be made of it using varying in degrees of complexity. Still, it is possible that the domineering masses preoccupied with linear assumptions have missed the nonlinear signals where the rewards of our careful investigation may be reaped [31]. We will produce nonlinearities of both types (functions on individual variables and interactions between them) in models that stretch from the simplest (i.e. a constant) to the direct opposite (involving large black boxes whose intricacies are difficult for humans to pick apart).

On the one hand, we seek to extract more artifacts from the information we have, and on the other, we strive to limit the scope of our search for sake of model parsimony and computational feasibility. The questions answered are thus: Do higher-order nonlinearities contribute meaningfully to quality of fit? What part of our data is fruitless noise that can be cut away? Which hyperparameters are worth tuning?

1.1.3 Interpretation of Results

Throughout the early exploratory stages and later evaluation of tuning, we aim to gain a better understanding of market structure through observation and reasoning. What differences are there between groups, and which configurations yield the best output? Although there is inevitably the possibility of overinterpretation, we make no pretense that these complications can be easily resolved by (mis-)applying hypothesis tests where they do not belong. That is to say, where fundamental assumptions such as independence and identical distributions are severely violated, we resist the urge to present p-values as if they would have any meaning.

That is not to abandon all hope of grasping at truth in paranoia of false discovery versus false rejection, but merely to excuse the author for underreporting guesses on standard errors which when taken out of their original contexts become insidiously unreliable in a highly dynamic environment. We describe the past within the setting of a single project. The emphasis here is on suggestions and insight regarding historical factors.

It is a most basic exercise to test a large number of hypotheses (as separate "experiments"), train them in-sample and evaluate out-of-sample, then announce only the most optimistic results (with promise that they are acceptably above some arbitrary significance cutoff). Despite techonlogical advances into modern data torture techniques, such is not valid extrapolation, as it cannot extract actionable truth out of them. Due to widespread publication bias (in pressure to publish exclusively positive findings), the author of this paper makes the data scientist's equivalent of the Hippocratic Oath for doctors – to treat the patient (data) well, against data snooping and the discarding of evidence too weak to reject the null, instead disclosing negative conclusions where interesting (if the intrinsic concern would be deemed of value had the conclusion been the opposite) [4].

1.2 Environment

1.2.1 Technology Stack

The workhorse employed in this campaign is an old personal laptop running Linux. No specialized OS functionalities were invoked (from kernel space), although it is convenient for myriad reasons in comparison to another (non Unix-based) household name that would abruptly restart and install obscure updates during the middle of running a data processing job, much to the author's irritation – prior to their long-overdue and happy separation without further alimony.

All data was downloaded from the web using scripts and stored on disk, primarily as plain CSV files (rather than in relational databases, because our data is numerical rather than intrinsically-structured). Our codebase contains a mix of `STL` C++ (for simple preprocessing operations with high throughput on larger raw data), Python with `pandas` and `sklearn` (most of the machine learning in the intermediate section of our integrated pipeline), and R with `MASS` and `readxl` (statistical analysis and visualization, for its conciseness and domain specificity).

In theory, any other Turing-complete language could have been substituted in any part; so these appear to be trivial personal preferences. Still, in practice there is a large performance difference in difficult-to-vectorize computations between these languages. Idiomatic, modern implementations of non-vectorized operations in C++ tend to be on the order of 10-100x faster than their Python and R counterparts [12, 23]. When an operation on a small dataset can be completed in 1 second, it seems insignificant. However, for larger datasets (requiring more than weeks of computation) this can become a bottleneck in the R&D cycle.

Besides speed, memory usage of Python and R are higher than in C++. On my dual-core (4 threads) laptop with 8GB of memory, inflating a 1.4GB file by a factor of 8x would leave no space for other processes, and swapping between RAM and hard drive would severely slow down an already slow operation. The easiest solution would have been to buy more memory, and by market prices, 64GB could cost as little as $300 at the time of this thesis' writing (but throwing more money at the problem is antithetical to our spirit).

Since this is an implementation project (as opposed to theoretical research), reasonable efforts will be made to enable replication of results. Code and data are uploaded to the author's Box (an online file-sharing and backup service), with technical assistance available upon request.

1.2.2 Execution Brokerage

The titular brokerage advertises commission-free trading of securities (including equities and exchange-traded funds; and as of 2018, options in addition to Ethereum and Bitcoin) listed on major exchanges [24, 25]. In reality, trades incur the customary SEC charges, and Robinhood Financial routes orders to high-frequency trading (HFT) firms listed in their RHF PFO Disclosure [26]. These fees are relatively small, although they add up over many trades. Robinhood makes no guarantees with respect to spreads and slippage. However, the advantage we gain is the absence of commissions (typically, around $5 to $10 per trade) collected by the brokerage. Its target customer segment is small-scale retail investors, rather than institutions and algorithmic traders.

Robinhood offers both market and limit orders, although its so-called "market" orders are effectively sent as limit orders with a 5% margin (above the ask in the case of buy orders, and below the bid when selling) [27]. We will trade actively, although there is no idiosyncratic needs for Robinhood's "market" orders (we can just submit limit orders at whatever price we deem opportune).

The majority of users use the Robinhood smartphone app for iPhone (from Apple's App Store) and Android (on the Google Play store). There is, however, also an API with meager official documentation. Users have created some third-party open-source wrappers in the open source community, but more on that topic later.

1.3 Business and Economic Analysis

1.3.1 Value Proposition and Customer Segmentation

Contrary to the affluence of ultra-HFT and low-latency strategies more appropriate for hedge funds and proprietary trading firms with well-endowed infrastructures, all of the tools involved in the making of this project are available to everyday traders. Little sophistication is required, and capital requirements are practically zero. If we succeed, financial markets invite broader egaliatarian participation (contra inequalities). Otherwise, civilians can waste less time wondering about unproductive ventures.

There is criticism of primarily a moral nature against traders. What value do they serve in society? They are not surgeons who save lives, engineers who build bridges, or firefighters who protect public and private property alike. Some vehemently decry Wall Street's financial market participants as thieves and rogues (and even aquatic animals, i.e. sharks), but the response from academic and professional finance has been mostly steered toward improving market efficiency [7]. Marketmakers and contrarian investors provide liquidity necessary for continued faith in the financial system. Proof of said belief: it has suffered shocks and survived. Increasing volumes and tightening spreads over the past century have shown that – contrary to the phobias of naysayers and conspiracy theorists – markets are thriving more than ever.

Financial liquidity greases the wheels of the economy so that investment capital moves to where it is most desired (and ideally, efficient; though inevitably we will have hiccups from time to time) [30]. In the approximate free market, selling may "harm" a company only to divert away from an unattractive investment manifesting stagnated growth and distressed situations.

1.3.2 Stakeholders

The primary stakeholders of a trading strategy are those who deploy capital to its operation (with compensation indirectly tied to P&L). Secondarily, are the counterparties who trade against them (by increased

access to trading partners). Third, are other market participants who are affected by cross-correlated movements (in reaction to the game situation). Finally, society is affected as a whole through diminishing ripples across the globally connected network [5].

1.3.3 Competition and Opportunities

Because those driven by human psychological biases and daily emotional prejudices do not react to market information rationally, in sufficient masses they may sway the market away from efficiency [18]. Likewise, technical chartists seldom backtest their "setups" (or apply otherwise sound statistical methodology indiscriminately), hence our minor edge in integrity and rigor.

Furthermore, bulky investment vehicles such as mutual and pension funds are at a scale too large to quickly change their positions without being noticed by HFT piranhas (and this phenomenon exists even in deep, liquid dark pools committed by manifesto in the war against trading "bots") [19]. Investment banks are subject to regulations on their prop trading departments (which have nearly shut down completely) and S&T desks that restrict their dexterity.

Unbeknown to most retail traders, there is a large variety of financial derivatives among other OTC products (some which may sound more familiar, and others relatively exotic). While we don't have access to all such institutional luxuries (nor do such securities have bountiful public data available for statistical modelling), our set of securities under consideration covers different asset classes; a multi-asset portfolio of baskets of underlying assets offers lower idiosyncratic risk. More pertinently, the macro view allows leveraging information from one asset class or continent in statistical (or "soft") arbitrage toward another [16]. This panoramic, multi-layered perspective that requires a large working and episodic memory is another enterprise that machines can perform efficiently.

1.3.4 Market Size

The "market cap" figures for ETFs are generally poor proxies for measuring the notional value of their underlying market sizes. An ETF commonly consists of a basket of securities whose value is derived from a weighted combination of trading on derivative products (futures, options, and swaps) and underlying assets in public exchanges, dark pools, along with OTC relationships.

To give a rough sense of the order of magnitude, however, major foreign exchange pairs (currencies), commodities (agriculture, energy, and metals), US interest rates (Treasury bonds), and global equity markets total trillions of dollars with billions transacted daily. We cover mostly liquid securities in this project to highlight the alpha forecasting problem (transaction cost analysis is a hugely lucrative aspect of algorithmic trading systems, albeit deserving of extensive separate assignment).

1.3.5 Profit Model

In the standard active trading framework, returns times capital correspond to revenue, and transaction fees (including the spread in dollar terms) correspond to costs. Their difference is profit.

As an aside – for delta-hedged marketmakers, revenue can be approximated by the product of volume and spread (or more precisely, the integral of differential spread over trading volume) plus commissions paid by brokerages and exchanges, which when subtracted by costs (slippage and transaction fees paid to regulators), yields profit. Other passive traders have the same cost model without designated marketmakers' commissions accumulating to their revenue line.

2 Data

2.1 Description

2.1.1 Survey of Sources

Due to the costs of streaming and storing large volumes of tick data (measured in the gigabytes to terabytes, as a lower-order estimate, although this depends on the depth of book and features included), few comprehensive and reliable sources distribute it freely. Two which we have found are HistData (with several years' bid and ask prices at top-of-book)[15] and Dukascopy (top-of-book bid price, bid volume, ask price, and ask volume over the past year)[10]. Although HistData has more forex minor pairs and observations dating as far back as May 2000, older data becomes stale eventually; importantly, it lacks lacking any explicit indication of volume.

Dukascopy is a Swiss forex bank and marketplace whose data cover a broader range of securities than the former, and we have selected it for use in this project. Its official website only allows downloading one day's tick data for a single security per request on its Java web applet, but data up to one year ago can be downloaded via an API provided by user `giuse88` on GitHub[14].

2.1.2 Financial Securities

On the next page we exhibit a table of all 44 securities that we will consider. A couple of acronyms to note are `CMD` for Commodities, `IDX` for Index, `US` for US-listed, and the standard currency denotations (`AUD` for Australian dollar, `CAD` for Canadian dollar, `CHF` for for Swiss franc, `EUR` for Euro, `GBP` for British pound, `JPY` for Japanese yen, and `USD` for US dollar). Organized by asset classes:

- 26 equity (`AUS.IDX/AUD, CHE.IDX/CHF, DEU.IDX/EUR, DVY.US/USD, EEM.US/USD, EFA.US/USD,ESP.IDX/EUR, EUS.IDX/EUR, EWJ.US/USD, EWZ.US/USD, FRA.IDX/EUR, FXI.US/USD,GDX.US/USD, GDXJ.US/USD, IVE.US/USD, IVW.US/USD, IWM.US/USD, JPN.IDX/JPY,QQQ.US/USD, SPY.US/USD, USA30.IDX/USD, USO.US/USD, XLF.US/USD, XLI.US/USD,XLP.US/USD, XOP.US/USD`);
- 6 commodities (`BRENT.CMD/USD, COPPER.CMD/USD, GAS.CMD/USD, GLD.US/USD, LIGHT.CMD/USD, XAG.US/USD`);
- 6 forex pairs (`AUD/USD, EUR/USD, GBP/USD, USD/CAD, USD/CHF, USD/JPY`);
- 3 bond (`EMB.US/USD, JNK.US/USD, TLT.US/USD`);
- 3 miscellaneous (`IYR.US/USD, VXX.US/USD, XIV.US/USD`);

where we have indicated in parentheses the Dukascopy abbreviation in (`security/base currency`) format. Notably, a number of countries are (non-randomly) sampled and not all security prices are denoted in US dollars. Even so, this is a relatively small subset extracted from Dukascopy's entire dataset, filtering out securities not available on Robinhood (neither directly nor by some close substitute instrument).

The first and second columns of the table are identical to those shown in Dukascopy's web applet. The third column represents what type of transformation would need to be applied on a forecasted price (in terms of Dukascopy's securities) to obtain the price of the equivalent security traded on Robinhood and whose information is shown through the 5th to 9th columns. The 4th column lists the name of equivalent security names on HistData (where available, for comparison). Each stock has a primary exchange (although it is also listed on others) in column 5, with the ticker (primary RIC) and full name in columns 6 and 7. Columns 8 and 9 represent the market cap (in USD) and average daily volume (number of shares trades), gathered in January 2018, which change from day to day and should generally be cursorily considered only as inexact indications on relative orders of magnitude.

Dukascopy abbreviation	Dukascopy description	transformation	HistData	primary exchange	ticker	company or fund name	market cap	average volume
AUD/USD	Australian Dollar vs US Dollar	[none]	AUD/USD	NYSEARCA	FXA	Guggenheim CurrencyShares Australian Dollar Trust	$131.84M	34159
AUS.IDX/AUD	Australia 200 Index	multiply AUD/USD	AUX/AUD	NYSEARCA	EWA	iShares MSCI Australia Index Fund (ETF)	$1.79B	2M
BRENT.CMD/USD	US Brent Crude Oil	[none]	BCO/USD	NYSEARCA	BNO	United States Brent Oil Fund LP	$103.51M	156586
CHE.IDX/CHF	Switzerland 20 Index	divide USD/CHF		NYSEARCA	EWL	iShares MSCI Switzerland Index Fund(ETF)	$1.33B	718205
COPPER.CMD/USD	High Grade Copper	[none]		NYSEARCA	CPER	United States Copper Index Fund ETV	$14.30M	8077
DEU.IDX/EUR	Germany 30 Index	multiply EUR/USD	GRX/EUR	NYSEARCA	EWG	iShares MSCI Germany Index Fund (ETF)	$4.86B	3M
DVY.US/USD	iShares Select Dividend ETF	[none]		NASDAQ	DVY	iShares Select Dividend ETF	$18.34B	711627
EEM.US/USD	iShares MSCI Emerging Markets ETF	[none]		NYSEARCA	EEM	iShares MSCI Emerging Markets Indx (ETF)	$44.69B	52M
EFA.US/USD	iShares MSCI EAFE ETF	[none]		NYSEARCA	EFA	iShares MSCI EAFE Index Fund (ETF)	$90.75B	18M
EMB.US/USD	iShares J.P. Morgan USD Emerging Markets Bond ETF	[none]		NASDAQ	EMB	iShares J.P. Morgan USD Emerging Markets Bond ETF	$11.73B	2.19M
ESP.IDX/EUR	Spain 35 Index	multiply EUR/USD		NYSEARCA	EWP	iShares MSCI Spain Capped ETF	$1.08B	1M
EUR/USD	Euro vs USD Dollar	[none]	EUR/USD	NYSEARCA	FXE	Guggenheim CurrencyShares Euro Trust	$247.54M	597416
EUS.IDX/EUR	Europe 50 Index	multiply EUR/USD	ETX/EUR	NYSEARCA	FEZ	SPDR Euro STOXX 50 ETF	$4.54B	3M
EWJ.US/USD	iShares MSCI Japan ETF	[none]		NYSEARCA	EWJ	iShares MSCI Japan ETF	$22.54B	2.72M
EWZ.US/USD	iShares MSCI Brazil Capped	[none]		NYSEARCA	EWZ	iShares MSCI Brazil Index (ETF)	$8.78B	18M
FRA.IDX/EUR	France 40 Index	multiply EUR/USD	FRX/EUR	NYSEARCA	EWQ	iShares MSCI France Index (ETF)	$802.90M	846100
FXI.US/USD	iShares China Large-Cap ETF	[none]		NYSEARCA	FXI	iShares FTSE/Xinhua China 25 Index (ETF)	$4.45B	14M
GAS.CMD/USD	Natural Gas	[none]		NYSEARCA	UNG	United States Natural Gas Fund, LP	$399.88M	3M
GBP/USD	Pound Sterling vs US Dollar	[none]	GBP/USD	NYSEARCA	FXB	Guggenheim CurrencyShares British Pound Sterling Trust	$202.41M	86520
GDX.US/USD	VanEck Vectors Gold Miners ETF	[none]		NYSEARCA	GDX	VanEck Vectors Gold Miners ETF	$7.33B	47M
GDXJ.US/USD	VanEck Vectors Junior Gold Miners ETF	[none]		NYSEARCA	GDX	VanEck Vectors Junior Gold Miners ETF	$4.61B	15M
GLD.US/USD	SPDR Gold Shares ETF	[none]	XAU/USD	NYSEARCA	GLD	SPDR Gold Trust (ETF)	$36.56B	7M
IVE.US/USD	iShares S&P 500 Value ETF	[none]		NYSEARCA	IVE	iShares S&P 500 Value Index (ETF)	$15.96B	833880
IVW.US/USD	iShares S&P 500 Growth ETF	[none]		NYSEARCA	IVW	iShares S&P 500 Growth Index (ETF)	$21.09B	658639
IWM.US/USD	iShares Russell 2000 ETF	[none]		NYSEARCA	IWM	iShares Russell 2000 Index (ETF)	$43.41B	25M
IYR.US/USD	iShares U.S. Real Estate ETF	[none]		NYSEARCA	IYR	iShares US Real Estate ETF	$3.59B	7M
JNK.US/USD	SPDR Barclays Capital High Yield Bond ETF	[none]		NYSEARCA	JNK	SPDR Barclays High Yield Bond ETF	$11.98B	10M
JPN.IDX/JPY	Japan 225	divide USD/JPY	JPX/JPY	NYSEARCA	EWJ	iShares MSCI Japan ETF	$20.96B	7M
LIGHT.CMD/USD	US Light Crude Oil	[none]	WTI/USD	NYSEARCA	USL	United States 12 Month Oil Fund LP	$90.61M	61462
QQQ.US/USD	PowerShares QQQ ETF	[none]		NASDAQ	QQQ	Powershares QQQ Trust, Series 1 (ETF)	$63.35B	28.95M
SPY.US/USD	SPDR S&P 500 ETF	[none]	SPX/USD	NYSEARCA	SPY	SPDR S&P 500 ETF Trust	$296.19B	71M
TLT.US/USD	iShares 20+ Year Treasury Bond ETF	[none]		NASDAQ	TLT	iShares Barclays 20+ Yr Treas.Bond (ETF)	$6.93B	8537
USA30.IDX/USD	USA 30 Index	[none]		NYSEARCA	DIA	SPDR Dow Jones Industrial Average ETF	$24.82B	3M
USD/CAD	US Dollar vs Canadian Dollar	reciprocal	USD/CAD	NYSEARCA	FXC	Guggenheim CurrencyShares Canadian Dollar Trust	$174.08M	73900
USD/CHF	US Dollar vs Swiss Franc	reciprocal	USD/CHF	NYSEARCA	FXF	Guggenheim CurrencyShares Swiss Franc Trust	$153.36M	22059
USD/JPY	US Dollar vs Japanese Yen	reciprocal	USD/JPY	NYSEARCA	FXY	Guggenheim CurrencyShares Japanese Yen Trust	$108.09M	139080
USO.US/USD	United States Oil	[none]		NYSEARCA	USO	United States Oil Fund LP (ETF)	$2.14B	23M
VXX.US/USD	iPath S&P 500 VIX ST Futures ETN	[none]		NYSEARCA	VXX	iPath S&P 500 VIX Short Term Futures TM ETN		23M
XAG/USD	Spot silver	[none]	XAG/USD	NYSEARCA	SLV	iShares Silver Trust (ETF)	$5.38B	7M
XIV.US/USD	VelocityShares Daily Inverse VIX Short Term ETN	[none]		NASDAQ	XIV	Credit Suisse AG - VelocityShares Daily Inverse VIX Short Term ETN	$1.14B	4.54M
XLF.US/USD	Financial Select Sector SPDR Fund	[none]		NYSEARCA	XLF	Financial Select Sector SPDR Fund (ETF)	$34.70B	63M
XLI.US/USD	Industrial Select Sector SPDR Fund	[none]		NYSEARCA	XLI	Industrial Select Sector SPDR Fund (ETF)	$15.75B	9M
XLP.US/USD	Consumer Staples Select Sector SPDR Fund	[none]		NYSEARCA	XLP	Consumer Staples Select Sector SPDR Fund (ETF)	$8.60B	10M
XOP.US/USD	SPDR S&P Oil & Gas Explor & Prodtn ETF	[none]		NYSEARCA	XOP	SPDR S&P Oil & Gas Exploration & Production ETF	$2.69B	16M

2.1.3 Raw Variables

For each of the 44 securities listed above, one year's data was downloaded through January 26, 2017 to January 26, 2018, containing a microsecond-timestamped index column and four variables per tick (one tick = one trade = one row): ask price, bid price, ask volume, and bid volume.

These data total 14.4GB altogether. The least-traded is EMB.US/USD (iShares J.P. Morgan USD Emerging Markets Bond ETF) whose file size is 13.8MB with 259,131 observations; the most-traded is GBP/USD (Pound Sterling vs US Dollar) at 1.4GB and 22,842,595 observations.

2.1.4 Paritioning

Due to missing observations for some securities, data prior to mid-May 2017 was discarded. The remaining portion is chronologically split into two subsets: training and evaluation.

Training data (00:00:00.000000 May 15, 2017 to 23:59:59.999999 September 14, 2017) is used in exploratory analysis, time-of-day analysis, scaling measurements, principal components analysis (PCA) fitting, and model tuning for selection hyperparameter sets.

Evaluation data (00:00:00.000000 September 15, 2017 to 23:59:59.999999 January 14, 2018) is further subdivided into two equal-sized halves of validation and test data. The validation data serves as a hold-out subset to evaluate the selected hyperparameter sets, as well as in retraining models later for out-of-sample evaluation on the test subset.

2.2 Feature Engineering

The following statements apply identically to each security. That is to say, we do not explicitly generate interaction terms between variables at this stage of the pipeline (although it is theoretically possible).

2.2.1 First-Order Transformations

These first-order transformations are applied on the raw tick data, to generate tick features. We have the arithmetic mean of bid and ask prices as their midpoint, total volume as the sum of bid and ask volume, and (relative) spread as the midpoint-price scaled difference in bid and ask prices.

For each tick,

$$\begin{aligned}
\text{midpoint price} &:= \frac{\text{bid price} + \text{ask price}}{2} \\
\text{total volume} &:= \text{bid volume} + \text{ask volume} \\
\text{spread} &:= \frac{\text{ask price} - \text{bid price}}{\text{midpoint price}}
\end{aligned}$$

Including the four raw variables, there are $4 + 3 = 7$ (first-order) tick features per security. With 44 securities, we have $44 \times 7 = 308$ tick features in total.

2.2.2 Second-Order Transformations

Upon the (first-order) tick features, we can further generate (second-order) bar features. These bars are fixed-duration windows (e.g. periods of 1 minute, 30 minutes) and cover a sequence of ticks. Hence, all the tick features (possibly an empty sequence) within each window are aggregated and mapped into several bar features (where meaningful). Namely, they are the return as the log-ratio of end value to initial value (at the start of a window), volatility defined as min-max range scaled by the arithmetic mean, sample unit-lag serial (or auto-)correlation, sample skewness which is the third standardized moment, and kurtosis which is the fourth standardized moment.

Let X be a sequence (univariate discrete timeseries) of length $n \geq 2$, whose first value is denoted x_1 and the last as x_n. Whence, we have a lagged subsequence $U = (x_1, \ldots, x_{n-1})$ and another unlagged subsequence $V = (x_2, \ldots, x_n)$. Per bar:

$$\begin{aligned}
\text{return}(X) &:= \log(\frac{x_2}{x_1}) \\
\text{volatility}(X) &:= \frac{\max(X) - \min(X)}{\text{mean}(X)} \\
\text{autocorrelation}(X) &:= \text{corr}(U, V) \\
\text{skewness}(X) &:= \frac{m_X^3}{s_X^3} \\
\text{kurtosis}(X) &:= \frac{m_X^4}{s_X^4}
\end{aligned}$$

where m_X^k and s_X are the k-th sample central moment and sample standard deviation of X, respectively.

The bar features are:

1. returns, volatilities, autocorrelations, skewnesses, kurtoses of
   i) bid prices
   ii) ask prices
   iii) midpoint prices
   iv) spreads
   v) bid volumes
   vi) ask volumes
   vii) total volumes
2. arithmetic mean of
   i) spread
   ii) bid volume
   iii) ask volume
   iv) total volume
3. number of ticks per bar

counting $(5 \times 7) + 4 + 1 = 40$ bar features per security, and across all securities $44 \times 40 = 1760$ bar features in total.

2.2.3 Imputation

Not every bar has two non-equal values, so we need to impute values where undefined. Although algorithms such as random forest are able to handle N/A values (if the implementation provides it), for convenience we will impute every undefined (or indeterminate) bar. The procedure is thus defined:

1. Fill with zero for mean volume (bid, ask, or total), number of ticks, return and volatility (on bid price, ask price, midpoint price, spread, bid volume, ask volume, or total volume).
2. Use the last observation carried forward (LOCF) if possible – otherwise, fill with zero – for mean spread.
3. As a last resort, replace with the column median if it has any valid values, or zero otherwise.

We believe this method makes economic sense because no ticks over a time interval indicates that there was no volume, nor ticks, that prices were constant (hence zero return and volatility); the spread invariant; and the median is a robust last-resort filler.

2.2.4 Response Variables

Using predictors lagged by one bar's period, the response (or dependent – in a non-causal sense) variables we aim to predict are i) the return on midpoint price and ii) mean spread, over the next bar. Hence, there are $44 \times 2 = 88$ responses in total.

We abstain from predicting absolute price levels because they are debatably limited in significance, and excessively small sample sizes (e.g. a new high has a sample size of 1) leads to extreme overfitting. Instead, we predict returns and spreads, as from these we can automatically deduce the profit (accounting for cost of crossing the spread) with an active trade executed over the next bar. See section **Evaluation Methodology and Trading Strategy** for detailed proof.

2.3 Exploratory Analysis

In this section, we descriptively analyze the raw data, first-order features, and then second-order features.

2.3.1 Inter-tick Gap Lengths

Securities have different trading times and days, some gaps attributed to lack of volume and others on weekends and holidays. We will deal with the time-of-day in the subsequent subsection, but on the next page we display all gaps between ticks that are at least 1 day (24 hours) in duration. For standardization, all timestamps are indicated in UTC+10.

The first column shows the starting date of the gap, and the second shows a comma-separated list of the gap lengths. We don't display the names of the securities to avoid too much clutter, but if it has a gap, then in each starting date's row it will have an entry in the list.

First, not all securities have gaps beginning on the same days. Second, we observe that the gap lengths starting on each date are uniform (by a design coincidence of global financial markets). Third, some gaps overlap – for example, the 2-day gaps beginning May 21, 2017 intersects with the gaps beginning the next day on May 22, 2017. Taking the union of these time segments, the three days from May 21, 2017 to May 23, 2017 is the combination of U.S. through Europe and Asia weekends (with up to a day's difference due to time zones).

We could perform separate analyses based on the inter-day gaps, but that would consume too much time. Instead, in going from the tick to bar features, we intersected the trading days to obtain a "common dates" set where every security is traded, and filtered down to only those days. Consequently, our remaining contains 52 days in the training subset and 80 days (40 validation and 40 test) in the evaluation subset.

date	inter-tick gap length (in days)
2017-05-21	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-05-22	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-05-26	[1]
2017-05-28	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-05-29	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-05-30	[3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]
2017-06-04	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-05	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-06	[3, 3]
2017-06-11	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-12	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-18	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-19	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-25	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-06-26	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-02	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-03	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-05	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
2017-07-09	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-10	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-16	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-17	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-23	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-24	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-30	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-07-31	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-02	[1]
2017-08-06	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-07	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-13	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-14	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-20	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-21	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-27	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-08-28	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-09-03	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-09-04	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-09-05	[3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]
2017-09-10	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
2017-09-11	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]

2.3.2 First-Order Features

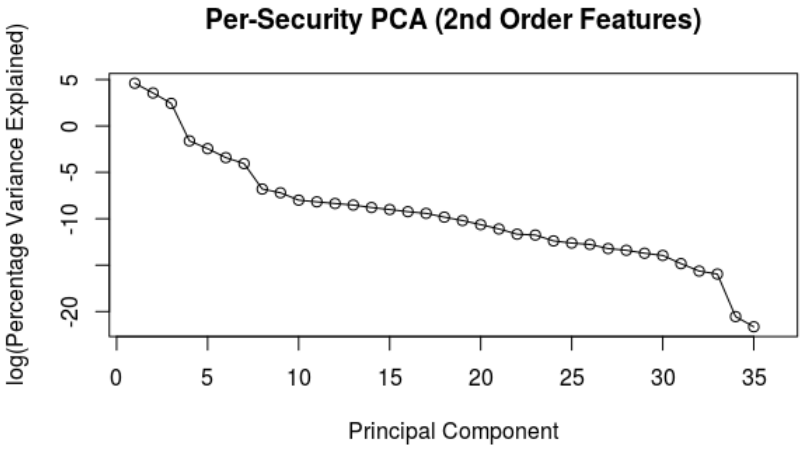
For each of the 44 securities, we perform principal components analysis (PCA) on its 7 first-order features. The following table shows the percentage of variance explained (PVE) by each principal component. The rownames indicate the ticker name (with the **[max]** row showing each column's highest PVE) for the k-th principal component, where k is indicated by the column number.

As we can see, the first two PCs expalin most of the variance. This makes sense because intuitively, the two orthogonal dimensions are price and volume. The first four PCs explain all of the variance (as computational witness, up to 1 millionth of a percentage point); likewise, we began with four raw variables.

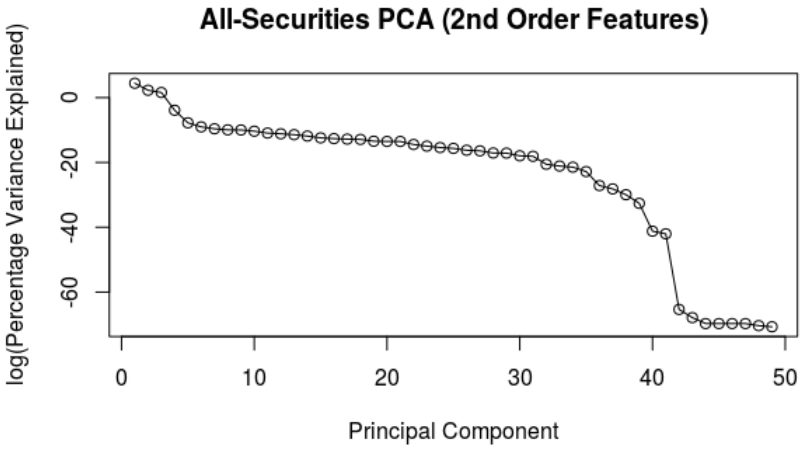
	1	2	3	4	5
[max]	99.96754958%	28.61990029%	8.58597038%	0.00061877%	0.00000000%
USA30IDXUSD	83.42950789%	16.57044521%	0.00004690%	0.00000000%	0.00000000%
USOUSUSD	75.11556055%	24.88443945%	0.00000000%	0.00000000%	0.00000000%
LIGHTCMDUSD	87.90144302%	12.09855698%	0.00000000%	0.00000000%	0.00000000%
FRAIDXEUR	84.21346321%	15.78653433%	0.00000246%	0.00000000%	0.00000000%
XLPUSUSD	76.27237538%	23.72762462%	0.00000000%	0.00000000%	0.00000000%
GBPUSD	89.14462743%	10.85537257%	0.00000000%	0.00000000%	0.00000000%
EEMUSUSD	73.03300086%	26.96699914%	0.00000000%	0.00000000%	0.00000000%
GDXUSUSD	73.57314975%	26.42685025%	0.00000000%	0.00000000%	0.00000000%
JPNIDXJPY	75.89429039%	24.10570397%	0.00000564%	0.00000000%	0.00000000%
TLTUSUSD	81.39829256%	18.60170744%	0.00000000%	0.00000000%	0.00000000%
EWZUSUSD	78.90478881%	21.09521119%	0.00000000%	0.00000000%	0.00000000%
IYRUSUSD	85.45802374%	14.54197626%	0.00000000%	0.00000000%	0.00000000%
XLFUSUSD	75.41227758%	24.58772242%	0.00000000%	0.00000000%	0.00000000%
EFAUSUSD	72.25032467%	27.74967533%	0.00000000%	0.00000000%	0.00000000%
XOPUSUSD	84.84888020%	15.15111980%	0.00000000%	0.00000000%	0.00000000%
VXXUSUSD	74.26585487%	25.73414509%	0.00000005%	0.00000000%	0.00000000%
EMBUSUSD	98.83244970%	1.16755030%	0.00000000%	0.00000000%	0.00000000%
IVEUSUSD	97.76218267%	2.23781733%	0.00000000%	0.00000000%	0.00000000%
JNKUSUSD	71.38009971%	28.61990029%	0.00000000%	0.00000000%	0.00000000%
GASCMDUSD	86.36975760%	13.63024240%	0.00000000%	0.00000000%	0.00000000%
USDCHF	82.28220828%	17.71779172%	0.00000000%	0.00000000%	0.00000000%
EURUSD	76.11773127%	23.88226874%	0.00000000%	0.00000000%	0.00000000%
SPYUSUSD	82.05351005%	17.94648994%	0.00000000%	0.00000000%	0.00000000%
DEUIDXEUR	86.42023734%	13.57975034%	0.00001232%	0.00000000%	0.00000000%
XLIUSUSD	76.86362177%	23.13637823%	0.00000000%	0.00000000%	0.00000000%
CHEIDXCHF	99.96754958%	0.03244704%	0.00000338%	0.00000000%	0.00000000%
BRENTCMDUSD	83.30724074%	16.69275891%	0.00000036%	0.00000000%	0.00000000%
FXIUSUSD	73.21972953%	26.78027047%	0.00000000%	0.00000000%	0.00000000%
COPPERCMDUSD	83.99969955%	16.00030045%	0.00000000%	0.00000000%	0.00000000%
USDJPY	75.16513972%	24.83486029%	0.00000000%	0.00000000%	0.00000000%
QQQUSUSD	78.37269934%	21.62730066%	0.00000000%	0.00000000%	0.00000000%
XIVUSUSD	87.71401307%	12.28598691%	0.00000001%	0.00000000%	0.00000000%
DVYUSUSD	99.59393536%	0.40606464%	0.00000000%	0.00000000%	0.00000000%
XAGUSD	81.51267620%	18.48732380%	0.00000000%	0.00000000%	0.00000000%
GDXJUSUSD	82.32144780%	17.67855220%	0.00000000%	0.00000000%	0.00000000%
AUSIDXAUD	70.77923181%	20.63417904%	8.58597038%	0.00061877%	0.00000000%
USDCAD	74.96650370%	25.03349630%	0.00000000%	0.00000000%	0.00000000%
EWJUSUSD	73.27247948%	26.72752052%	0.00000000%	0.00000000%	0.00000000%
IWMUSUSD	84.66347307%	15.33652693%	0.00000000%	0.00000000%	0.00000000%
IVWUSUSD	97.75487840%	2.24512159%	0.00000001%	0.00000000%	0.00000000%
GLDUSUSD	82.58518464%	17.41481536%	0.00000000%	0.00000000%	0.00000000%
ESPIDXEUR	86.70429825%	13.29565167%	0.00005008%	0.00000000%	0.00000000%
AUDUSD	77.88687331%	22.11312669%	0.00000000%	0.00000000%	0.00000000%
EUSIDXEUR	87.35863153%	12.64136644%	0.00000204%	0.00000000%	0.00000000%

2.3.3 Second-Order Features

Similarly, we extracted PVEs by PCA on each security's set of 40 second-order features. The table would be too wide to fit on this paper, but we can plot the maximum PVE per PC number. The highest number of PCs found was 35. Below is a plot with log(PVE) on the vertical axis.



And we can perform another PCA after concatenating all 1760 second-order features and computing the PVE per PC in the entire matrix. The total number of PCs was 49 (which is less than 3% of the number of second-order features), although only 21 PCs had PVEs above 10^{-6}.



From both plots, we can see (minding the dependent variable's log-scale) that PVE decreases "exponentially" over the range of PCs.

2.4 Time-of-Day (ToD)

Typically, equities are traded during daylight market hours of the local region. US ETFs are traded during regular American stock exchange hours. Commodities and currencies tend to be traded for longer periods each day.

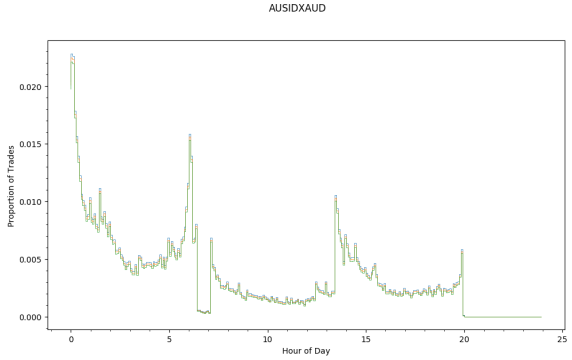
Since the availability of ticks varies depending on the time-of-day, directly concatenating different securities' feature matrices would yield a relatively sparse matrix with much missing data. Our ToD analysis will make sense of these diverse trading hours.

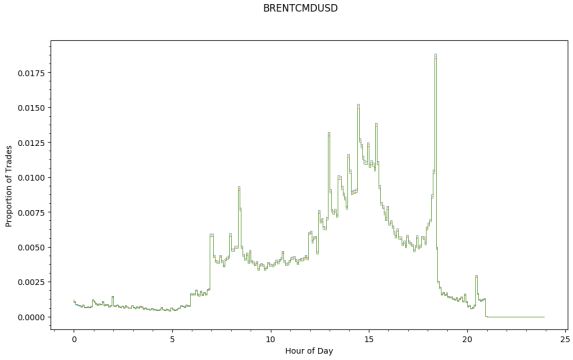
2.4.1 Histograms

For visualization purposes, suppose we aggregate each continuous 5-minute collection of ticks into "bins". Taller bins indicate more trading activity (often at the start and end of the day), while shorter bins indicate periods of relative inactivity. The green, orange, and blue (piecewise) lines are the lower bound, mean, and upper bound proportions based on 95% Jeffreys confidence intervals.

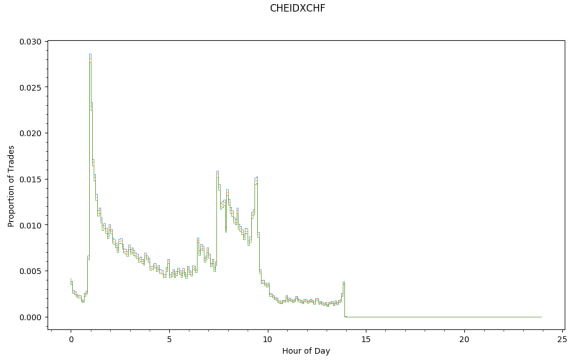
A pattern that frequently emerges is a U-curve (or parabola), within the market hours of a region. For securities that trade over multiple geographic regions, there may be multiple U-curves. We will exhibit some archetypal histograms of trading activity (not exactly in terms of volume, but number of trades).

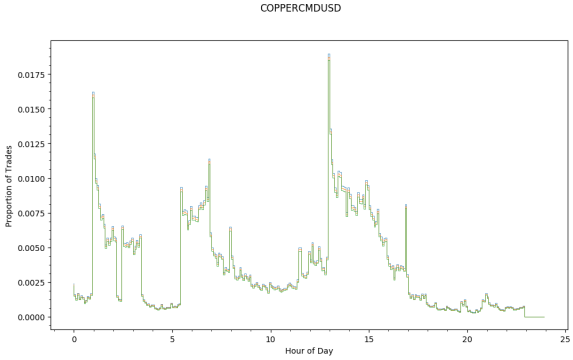
Again, we note that the ToD on the horizontal axis is in UTC+10 (Australian hours), as we follow the sunrise from East to West of Greenwich geographically, while scanning from left to right on the plots; this way, no local stock exchange hours are broken across different days, and all opening to closing hours are covered between 00:00 and 24:00.



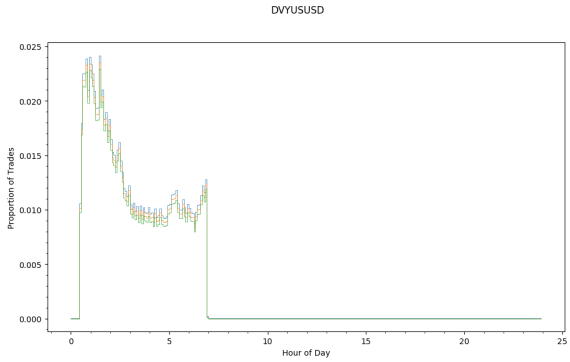


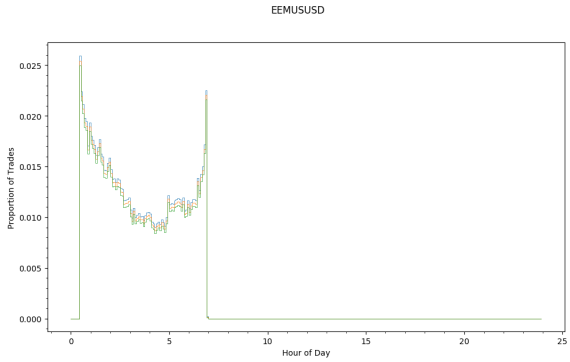
Australia's ASX 200 (AUS.IDX/AUD) shows its trading during European, American, and Australian market hours. We see three distinct U-curves. Brent crude oil (BRENT.CMD/USD) trading is slow at the start of the day, and then picks up in the middle of the day. There is a 3-hour break beginning at 21:00.



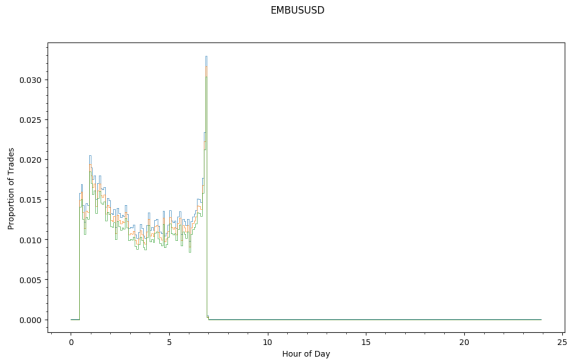


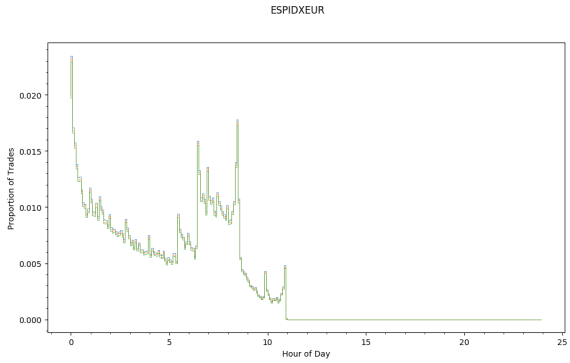
On left is the Switzerland Stock Market index (CHE.IDX/CHF), which tracks a basket of 20 blue chip stocks. It looks different still from the previous plots. There is a 10-hour break at the end of the day (starting from 14:00). The plot on the right shows copper (COPPER.CMD/USD), which apparently has a few spikes of activity throughout the day.



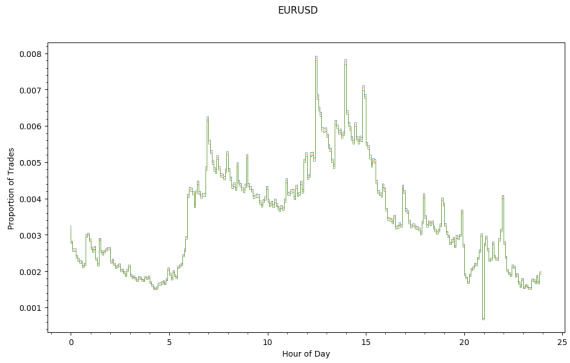


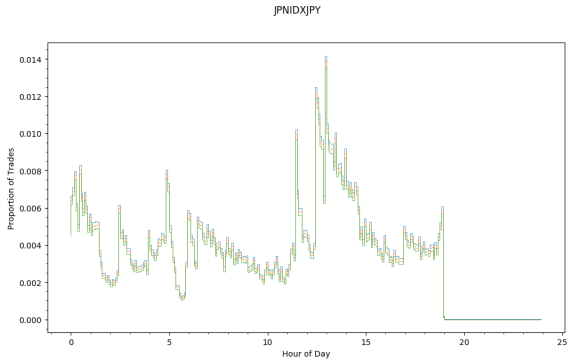
The iShares Select Dividend ETF (DVY.US/USD) is traded on NASDAQ during the US's hours. It follows a U-curve trend although there is less interest near the end of the day. One possible explanation is that fund managers who invest in dividend-yielding companies are mostly long-term value investors rather than opportunistic day traders. The iShares MSCI Emerging Markets ETF (EEM.US/USD) offers a higher risk-reward profile (with a basket of securities across multiple developing countries). It has a relatively more symmetric U-curve than that of aforementeiond dividend investors.



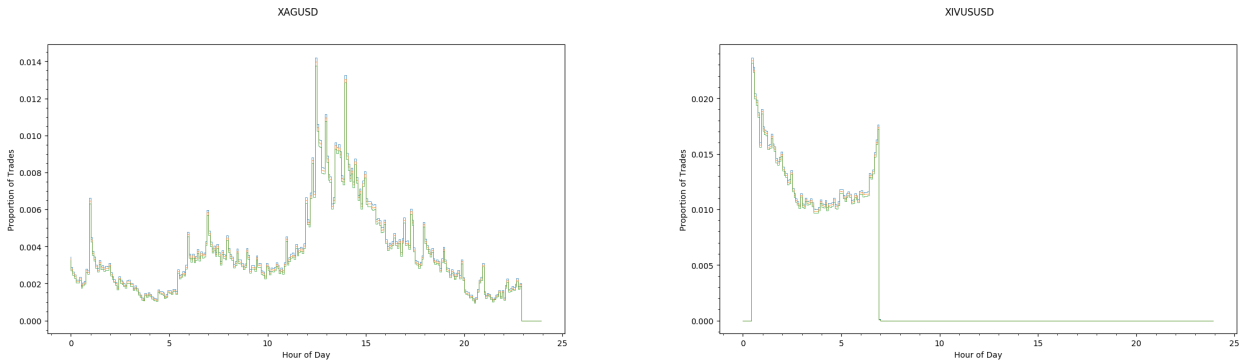


Here is yet another, the iShares J.P. Morgan USD Emerging Markets Bond ETF (EMB.US/USD). This fixed income basket has heavier trading during the end of the day (roughly twice as many trades in the last 5 minutes prior to the closing bell than any other 5 minutes). Spain's IBEX 35 Index (ESP.IDX/EUR) has a trading pattern distinguished from those of US markets.





The Euro-US Dollar (EUR/USD) exchange seems fairly evenly-distributed in activity throughout periods of the day, although somewhat heavier in the middle; there are no dormant periods except a short 5-minute cooldown at the 21st hour. Japan's Nikkei 225 (JPN.IDX/JPY) trades for 19 hours of the day, followed by a short 5-hour sleep.



Above on the left is the silver spot market (XAG/USD). It only sleeps for one hour per night. VelocityShares Daily Inverse VIX Short Term ETN (XIV.US/USD) is an exchange-traded note on the CBOE's VIX index of volatility, trading during regular US hours. Its "skewed" U-curve with highly active opening hours and (moderately) active closing hours is more typical of US stocks. Although it is barely visible, there is a small residual of after-hours trading after the closing bell (and this is regular on American exchanges).

Hopefully, it is clear now that securities have different trading behavior; their activities are sometimes, but not always simultaneous.

2.4.2 Jaccard Complement

Define Θ to be the finite collection of $24 \times 60 \div 5 = 288$ five-minute bins (00:00,00:05) to (23:55,24:00). Then if $f, g : \Theta \to [0, 1]$ are two functions of trading activity (proportions of ticks) for X and Y from the collection of securities Ω, we can refer to whether a security has any trades at all during any fixed but arbitrary 5-minute bin $\theta \in \Theta$ by two functions $a(\theta) := I(f(\theta))$ and $b(\theta) := I(g(\theta))$ respectively for X and Y, where $I : \mathbb{R} \to \{0, 1\}$ is the (strictly) positive indicator function ($I(p) = 1$, if $p > 0$ else $I(p) = 0$ for $p \in \mathbb{R}$).

The Jaccard index of overlap for discrete bins is the indicator proportion among all non-empty bins in the intersection over the union between two securities [20]. We can form the (not necessarily strict) subcollections of bins over which X and Y are traded as $B_X := \{\theta \in \Theta \mid a(\theta) = 1\}$ and $C_X := \{\theta \in \Theta \mid b(\theta) = 1\}$.

$$J(X, Y) := \frac{|B_X \cap C_X|}{|B_X \cup C_X|} = \frac{\sum\limits_{\theta \in \Theta} a(\theta) \cdot b(\theta)}{\sum\limits_{\theta \in \Theta} \max(a(\theta), b(\theta))}$$

where $|A|$ of a (finite) collection A denotes its cardinality.

Then the Jaccard complement, $1 - J(X, Y)$, is a dissimilarity metric on the distance between two securities by the degree of difference between their ToD patterns. The Jaccard index and complements are real numbers between 0 and 1 inclusively, i.e. $0 \leq J(X, Y), 1 - J(X, Y) \leq 1$.

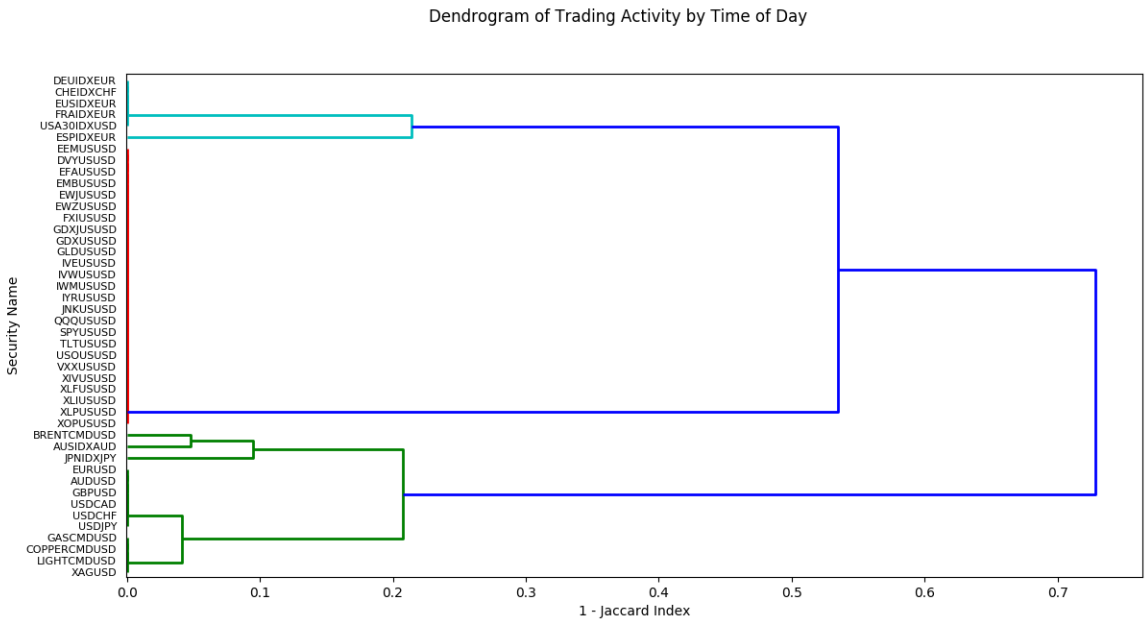
$$1 - J(X, Y) := \frac{|B_X \setminus C_X| + |C_X \setminus B_X|}{|B_X \cup C_X|} = \frac{\sum\limits_{\theta \in \Theta} a(\theta) \cdot (1 - b(\theta)) + (1 - a(\theta)) \cdot b(\theta)}{\sum\limits_{\theta \in \Theta} \max(a(\theta), b(\theta))}$$

2.4.3 Dendrogram

We can visualize clusters using a dendrogram. At each height level of Jaccard complement (on the horizontal axis), all securities (labelled at the leaves) with distance lower than the height threshold are merged into one node. At the top of the tree, all nodes are merged into the tree's root.

Upon first glance, a few European securities are combined early. US stocks all have shared trading hours. The forex pairs are traded 24/7. Commodities trade for most hours of the day except near the end, when there is a short break.

Since the latter's hours go almost full cycle around the clock, they are merged at a low level (climbing from the computer science-style upside-down tree's leaves), which then merge with European stock indices, and finally with America.



2.4.4 Coverage Counts

From another perspective, we can view coverage defined as how many securities altogether are traded during the same hours as each. For example, during US stock market hours every security is actively traded (so it is covered by all). On the other end, the forex pairs' hours are only covered by each other (and their own). Below are the full lists, in alphabetical order,

[**coverage count**]: [**list of securities**]
44: DVY.US/USD, EEM.US/USD, EFA.US/USD, EMB.US/USD, EWJ.US/USD, EWZ.US/USD, FXI.US/USD, GDXJ.US/USD, GDX.US/USD, GLD.US/USD, IVE.US/USD, IVW.US/USD, IWM.US/USD, IYR.US/USD, JNK.US/USD, QQQ.US/USD, SPY.US/USD, TLT.US/USD, USO.US/USD, VXX.US/USD, XIV.US/USD, XLF.US/USD, XLI.US/USD, XLP.US/USD, XOP.US/USD
19: ESP.IDX/EUR,
18: CHE.IDX/CHF, DEU.IDX/EUR, EUS.IDX/EUR, FRA.IDX/EUR, USA30.IDX/USD
13: JPN.IDX/JPY
12: AUS.IDX/AUD
11: BRENT.CMD/USD
10: COPPER.CMD/USD, GAS.CMD/USD, LIGHT.CMD/USD, XAG/USD
6: AUD/USD, EUR/USD, GBP/USD, USD/CAD, USD/CHF, USD/JPY

2.4.5 Cluster Visualization

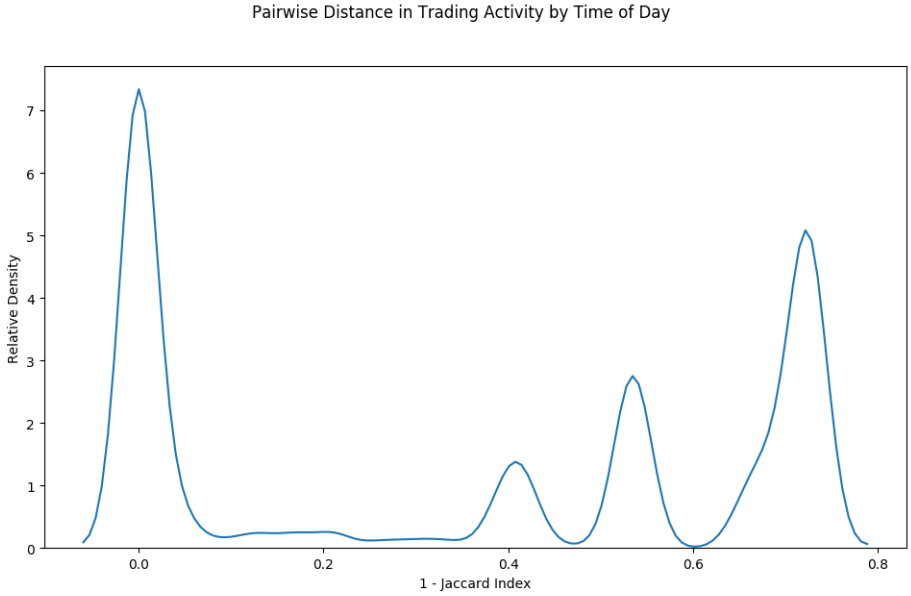
Another technique utilized was constructing multiple graphs where edges indicate either coverage (directed) or shared ToD (undirected) at different thresholds. Then related securities will be clustered in connected components.

Although for interpretation, it is quite useful for smaller networks, in dense graphs with many vertices, the image can become large and difficult to decipher so we omit its inclusion. In larger networks, such graphs can show overall patterns, but the fine-grained and specific information becomes difficult to discern so there

is usually less readable text. In this application, the dendrogram and coverage lists more than suffice for our analysis, with greater level of detail available in the historgrams.

2.4.6 Pairwise Distances

Last and least, we observe a (kernel-smoothed) density plot of the Jaccard complements $\{1-J(X,Y) \mid X,Y \in \Omega\}$; there are (44 choose 2) = 946 pairs of securities. As expected, many pairs on the US market have zero distance; there are a few smaller modes farther right, but these are fairly distant.



2.5 Preprocessing

2.5.1 Filtering by Time-of-Day (ToD)

Because US market hours covers a large proportion of overall trading activity, and because Robinhood's accessible liquidity is mostly limited to regular intraday US trading hours, we will restrict ourselves to 00:30 to 07:00 in the UTC+10 timestamp (which translates 09:30 to 16:00 in New York's EST). In the subsequent stages of our analysis, all remaining data is discarded

2.5.2 Predictor Scaling and Dimensionality Reduction

First, we compute the sample mean and standard deviation of the training subset (of second-order features). Then, we use these same measurements to normalize each column across the entire datset (including training and evaluation subset).

Then, we extract 100 principal components via singular value decomposition. The PCA model is fitted on the training subset, in the same manner as with scaling, and then used to transform the second-order features in every subset.

Unfortunately, PCA makes interpretation of predictor importance more opaque to interpretation. The necessity is justified by performance optimizations considering the timeline of this project.

That poses the question of why we are doing PCA at all. First, with excessively large number of predictors d compared to sample size n, overfitting and slow computation are major practical issues. Hence, dimensionality reduction is necessary to counter the curse of dimensionality. Second, orthogonalization will be useful later in measuring predictor importance and in our additive model, as multicollinearity inflates variances (hence estimated coefficients are unstable) while feature selection becomes difficult when predictors are highly correlated.

2.5.3 Response Scaling

The response matrix (of the 44 securities' two variables each) needs to be scaled before using `sklearn` due to numerical instability. In its underlying implementation, `pandas` is built on C code with limited double precision representation.

To illustrate, consider a small y on the order of 10^{-6}. When subtracted by another small $y' \approx y$ to yield an even smaller difference, the squared residual gets rounded down to 0 and suggest to a random forest algorithm that no split is needed. This is a legitimate issue factually encountered.

Thus, we first scaled the response matrix before fitting models, and then after prediction, we transformed our model's output back into the original space.

3 Model Tuning and Hyperparameter Selection

For each response variable, a separate set of hyperparameters (per algorithm) and PCs (where relevant) are selected because each is differently related to different markets. It is possible that no single hyperparamter set works well for all response variables, and the predictors for one response are nearly useless for another.

The moving window method is common. However, we will fix the lookback horizon to 52 days for sake of a larger sample size, in great need of variance reduction. If the model changes completely from one day to the next, then it is probably severely overfitting, and one struggles to imagine how much signal (as opposed to merely noise) has been detected; robust models that yield consistent performance out-of-sample are highly desired.

3.1 Hyperparameter Sets

In this subsection we discuss our usage of three algorithms to generate fitted models. Although elastic net tends to be moderately parsimonious due to regularization upon the additive model assumption, k-nearest neighbors and random forest can vary from the most simple ($k = n - 1$ and decision stump, respectively) to highly complex ($k = 1$ and trees with up to n leaves). In a colloquial sense, model complexity depends as much on to which family it belongs, as location within its family. We declare no strict ordering of algorithms.

Although some algorithms have over a dozen configurable hyperparameters, we need not fret over all of those available in `sklearn` as 1) some arguments passed to the class constructors are functions of the others so there are actually fewer degrees of freedom than presented to us, 2) a number of the hyperparameters do not have meaningfully-large effects on the actual models generated, and 3) most of the default values are reasonably well-chosen. We have utilized multicore processing wherever possible.

3.1.1 Elastic Net (EN)

`sklearn`'s implementation formulates the objective as follows:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(\tfrac{1}{n}||y - X\beta||_2^2 + 2\alpha\lambda||\beta||_1 + \alpha(1-\lambda)||\beta||_2^2)$$

where $\alpha \in \mathbb{R}_0^+$, $0 \leq \lambda \leq 1$.

Intuitively, α controls the overall degree of regularization, where $\alpha = 0$ corresponds to ordinary least-squares (OLS) regression and larger α yields a more parsimonious fit (with reduced coefficient magnitudes, possibly zero). λ is our scaling parameter, where in particular (assuming $\alpha > 0$), $\lambda = 0$ yields ridge regression with L1 penalty and $\lambda = 1$ yields LASSO with L2 penalty.

In logarithmic sequence, we experimented with 241 values of α from 10^{-12} to 10^{12}; and in a linear sequence, 50 values of λ from 0.02 to 0.98; for $241 \times 50 = 12,050$ hyperparameter sets in combination.

3.1.2 Random Forest (RF)

There are two free hyperparameters. First, is the minimum number of observations (sample size) required per leaf node, effectively limiting the depths of the trees (assuming they are reasonably balanced); we consider values of ν (minimum observations per leaf) from the list (1,2,5,10,20,50,100,200,500,1000) for 1-minute bars and (1,2,5,10,20,50,100,200,500) for 30-minute bars. Second, and this is not inherent to the RF algorithm, we vary the d (number of PCs) from (1,2,5,10,20,50,100). In total, we have $7 \times 13 = 91$ hyperparameter sets.

The number of (bagged) trees is fixed at 500; more would be preferable, if we had more time. Also, with d PCs, we fix $\sqrt{d} \leq \sqrt{100} = 10$ PCs to be (randomly) considered when looking for the best split; this is a balanced rule of thumb due to computational constraints.

3.1.3 Stepwise k-Nearest Neighbors (SkNN)

To use all PCs as predictors in kNN would occlude the signal with noise as most of them – one may reasonably presume – are useless. Instead we proceed in forward stepwise selection of predictors. Each step, we include the two best PCs (if adding any yield better fits on the validation subset) with one according to each of the **Measures of Fit** (see later subsection). In this project, we made 3 steps (to select 6 PCs) per response variable.

At each step, we vary the number of nearest neighbors k from RF's corresponding lists for the minimum number of observations per leaf (1,. . . ,1000) and (1,. . . ,500) depending on the bar period; we also consider two weight functions (to be abbreviated w), "uniform" (weighting each nearest neighbor equally) and "distance" (by inverse distance, so that closer neighbors are more heavily weighted than those farther away). These total $10 \times 2 = 20$ hyperparameter sets (per stage).

3.2 Computational Time Analysis

A run of each algorithm has asymptotic time complexities as well as a constant factor multiplier attributed to its specific hyperparameter set, implementation, and hardware. The latter two are fixed for us, so we will discuss only asymptotic and hyperparameter-related performance.

3.2.1 Elastic Net (EN)

For (predictors) $d < n$ (number of observations), asymptotic time complexity is $O(d^2 N)$ (in the same class as linear regression). `sklearn`'s coordinate descent algorithm takes longer to converge for smaller values of α (and for $\alpha = 0$, it is recommended to use OLS instead).

3.2.2 Random Forest (RF)

Let M be the number of (binary decision) trees, and $N = (1 - e^{-1})n \approx 0.632n$ is the bagged sample size (the constant factor does not make a difference in complexity class). At each split, suppose we consider each of $\sqrt{d}$ PCs and sort $Nlog(N)$ observations; with N observations, the maximum tree depth is $\frac{N}{\nu}$, where ν is the minimum observations per leaf node; therefore, RF of M trees belongs to $O(M\sqrt{d}\frac{N^2}{\nu}logN)$.

As we have fixed M, we can vary d with which time complexity is directly related to its square root, and time is inversely related to ν (smaller ν cause slower fitting, while computation finishes quicker with larger ν).

3.2.3 Stepwise k-Nearest Neighbors (SkNN)

For S stages of kNN using k-d tree (with $d << n$) nearest neighbors search, the overall time complexity is $O(Sdklogn)$. The runtime depends linearly with respect to k but is independent of the weight function.

3.3 Quality of Fit

Although eventually we will benchmark profitability in dollars terms (conditional upon predictions exceeding a upper tail cutoff), for now we will measure quality of fit on all observations in the validation subset for sake of achieving lower standard errors.

3.3.1 RMSE ($\rightarrow R^2$)

From $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$ and $R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$, clearly R^2 is inversely related to (decreases monotonically with) $RMSE$.

RMSE as a measure of error (and R^2 of performance) carries psychological analogues with volatility aversion, since large deviations from the mean are more severely penalized. To be risk-neutral, we could substitute L2 with L1 loss; but we did not, in order to remain consistent with `sklearn`'s built-in L2 loss.

3.3.2 Correlation (Kendall's τ)

As a nonparametric correlation coefficient, Kendall's τ measures the difference in proportions of concordant and discordant pairs. First, ranks are computed on two vectors of values – actual (A) and predicted (B), for whose k-th element's ranks we denote a_k and b_k. Then for each pair of distinct observational indices i and j ($i \neq j$), we construct the following sets by the membership of (i, j):

1. A if $a_i = a_j$ (tie in A)
2. B if $b_i = b_j$ (tie in B)
3. C if $(a_i - a_j)(b_i - b_j) > 0$ (concordant)
4. D if $(a_i - a_j)(b_i - b_j) < 0$ (discordant)

and define:

$$\tau := \frac{|C| - |D|}{\sqrt{(|A| + |B| + |C|)(|A| + |B| + |D|)}}$$

By virtue of being rank-based, τ tends to more be robust to extreme outliers (compared to Pearson's ρ measuring linear correlation). The disadvantage relative to $RMSE$ in comparing models based on correlation between actual and predicted values is that it does not account for differences in scale; if the actual values are (1,2,3), then we obtain the same correlation regardless of whether we predict (0.1,0.2,0.3) or (20,40,60).

3.4 Predictor Importance

3.4.1 Hypothesis

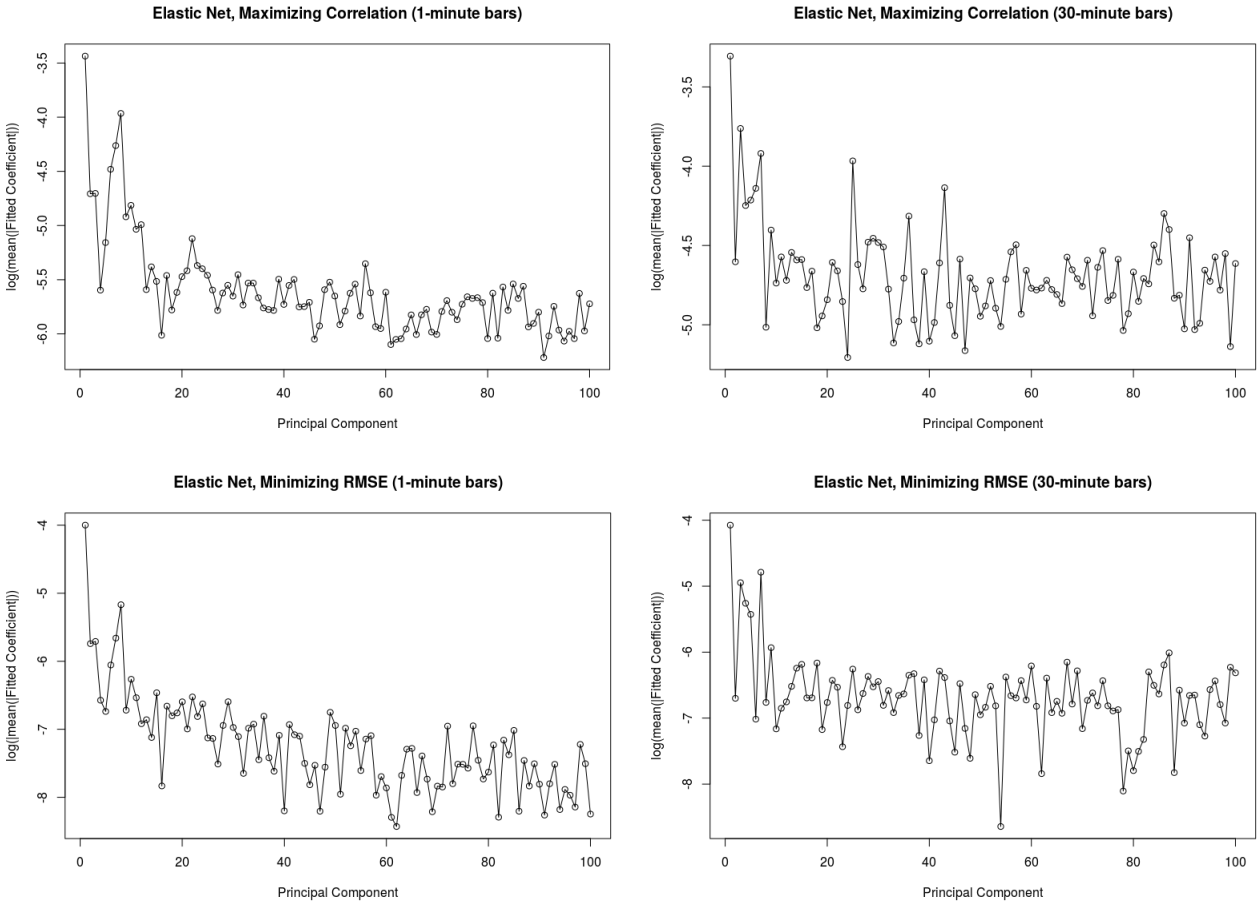
We hypothesize that predictor importance scores will following a decreasing trend with respect to the PC number (so the 1st PC is expected to be highly important, while the 100th is less so).

3.4.2 Procedure

The two algorithms that lend most naturally to measuring predictor importance are EN, whence we can compare the absolute value of coefficients (since the predictor matrix was re-scaled after PCA); and RF, whence we examine the mean decrease in variance by splitting on each predictor.

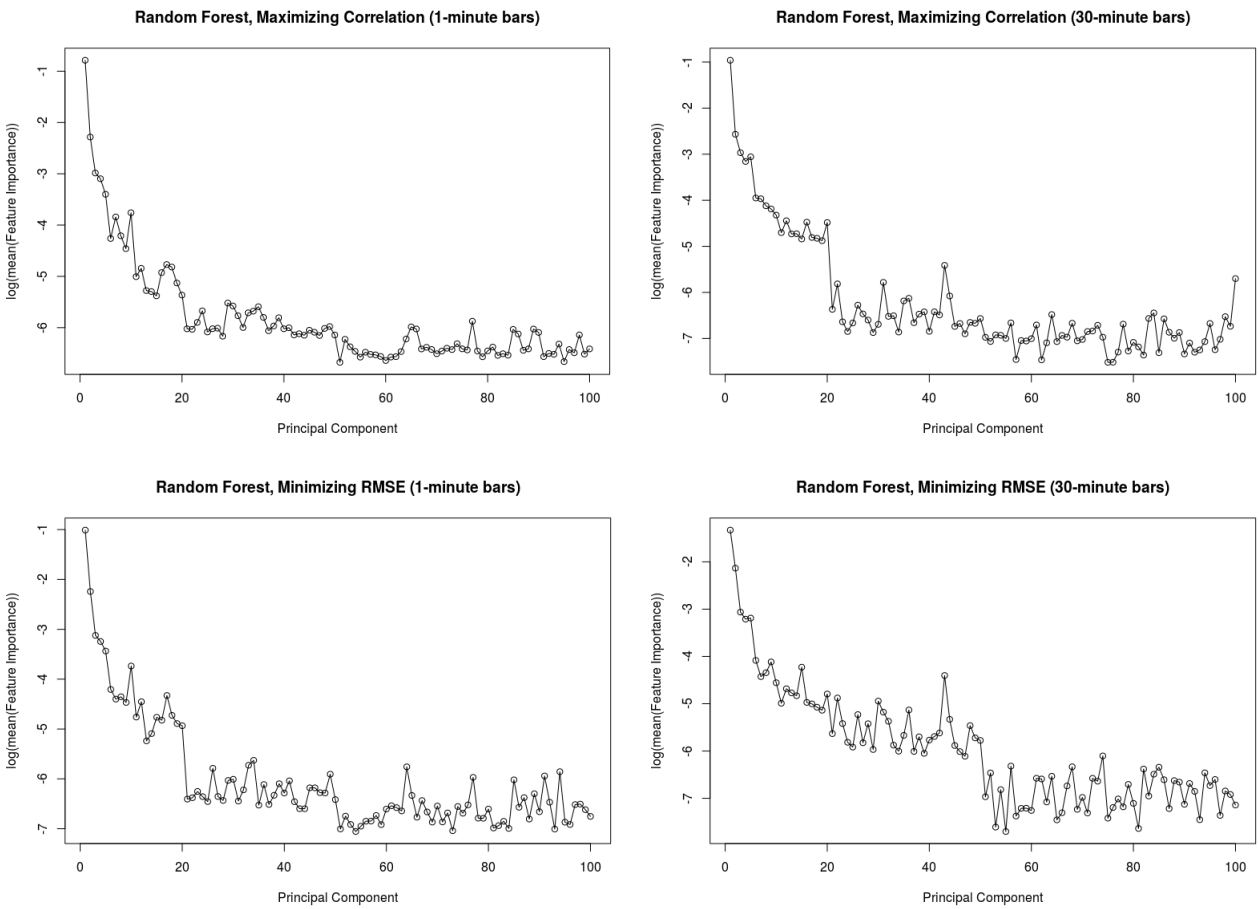
PC selection from SkNN is another framework for measuring predictor importance, although the frequencies with which PCs are selected do not correspond to intuitive magnitudes regarding their relative importances, and the greedy paradigm is pessimistically challenged on grounds of local suboptimality (a detraction which does not apply as extensively to RF when we have 500 bagged trees with randomly selected predictors).

3.4.3 Results



In the quadrant of plots above, we show the relationship between log-scaled average (over each response variable's regression equation) of $|\beta_j|$ for the j-th PC using EN after its hyperparameters have been optimized to the two measures of fit, with 1-minute and 30-minute bars.

In the plot of each individual scenario, we verify that our hypothesis was mostly correct (albeit, the average fitted coefficient magnitudes do not decrease monotonically from every PC to the next). Another sight worthy of mention is that under the RMSE-minimization objective, the magnitudes decrease further than with the correlation-maximizing objective (as evident by the y-axis ticks).



The hypothesis holds true for RF as well, with more monotonically decreasing predictor importance scores; this is especially evident when comparing trends over the first 20 PCs. Also, there is a less pronounced difference in dropoffs between the correlation and RMSE objectives.

Across both algorithms, we did not notice any markedly consistent differences going from left to right, comparing the "curves" for 1-minute versus 30-minute bars. Recalling that the vertical axis is converted to a (natural) log scale, much of the variance can be explained by the first 10 PCs.

3.5 Hyperparameter Selection

This subsection presents an analysis on average quality of fit (over all response variables) as a function of the hyperparameter sets per algorithm (for 1-minute and 30-minute bars).

A few questions ought to be posed (and we will make an effort to answer them):

1. For each algorithm, are there hyperparameter values that suit all response variables, or do the selected values vary according to the response (this may also suggest insignificance)?
2. How are the (empirical, discrete) distributions shaped for each hyperparameter's selected values?
3. Do the selected hyperparameters correlate between the two objectives (τ and RMSE)?
4. Which predictors are selected at each step of SkNN?

To avoid the pretense of being an oracle, we will adopt the $\hat{h}$ notation to manifest that the selected hyperparameters are not truly "optimal" in the sense of h, but rather, estimated from our (limited) sample. We abstain from promising validity of formal inference (e.g. "evidence against H_0 significant at the .05 level") because when assumptions are severely violated, it would be doctrinally unsound; moreover, if we cherry-pick significant differences, we would need to somehow properly account for the family-wise error rate of significance mining across multiple unknown distributions.

Figures are rounded where appropriate, and cells in some tables are left blank because their interpretation would be either multi-valued (many modes in the case of ties), ill-adapted (e.g. taking the arithmetic mean of a logarithmic sequence), or useless (kurtosis on observed vector of bernoulli variables).

3.5.1 Elastic Net (EN)

1-minute bars	τ: $log_{10}(\hat{\alpha})$	$RMSE$: $log_{10}(\hat{\alpha})$	τ: $\hat{\lambda}$	$RMSE$: $\hat{\lambda}$
[first quartile]	-2.7	-1.6	2%	11%
[arithmetic mean]	-2.6	-1.1	52%	59%
[third quartile]	-1.8	-1.1	66%	82%
[standard deviation]	2.1	0.6	39%	39%
[skewness]	-1.3	0.1	-0.2	-0.5
[kurtosis]	0.9	-0.7	-1.7	-1.6
inter-objective correlation	6.2%		0.5%	

With 1-minute bars, most of the selected $\hat{\alpha}$ were in the middle of our search range; the RMSE objective penalized coefficient magnitudes more harshly. In juxtaposition, $\hat{\lambda}$ seemed rather inconsistent. Each vector of selected hyperparameters were not heavily skewed, though they were all platykurtic (so I omitted tail quantiles). The $\hat{\alpha}$ values selected between the two objectives (τ and RMSE) were not strongly correlated; ditto for $\hat{\lambda}$.

30-minute bars	τ: $log_{10}(\hat{\alpha})$	$RMSE$: $log_{10}(\hat{\alpha})$	τ: $\hat{\lambda}$	$RMSE$: $\hat{\lambda}$
[first quartile]	-1.3	-0.9	21%	14%
[arithmetic mean]	-1.2	-0.6	57%	61%
[third quartile]	-1.0	-0.8	68%	84%
[standard deviation]	1.1	0.5	35%	39%
[skewness]	-1.7	1.0	-0.4	-0.6
[kurtosis]	3.5	-0.2	-1.4	-1.4
inter-objective correlation	8.7%		21.0%	

Since for EN we used the same space of hyperparameters to choose from between both bar durations, it is fair to compare with the selections from above. As a discrete proxy for "stochastic dominance" (at Q1, mean, and Q3), the 30-minute bars' $\hat{\alpha}$ were higher; sample sizes were smaller by a factor of 30, and the coefficients were more heavily penalized than before. The selected hyperparameters are less spread out (lower SD), and they were more correlated between objectives (especially for $\hat{\lambda}$).

3.5.2 Random Forest (RF)

1-minute bars	τ: $\hat{d}$	RMSE: $\hat{d}$	τ: $\hat{\nu}$	RMSE: $\hat{\nu}$
[first quartile]	2	2	18	200
[geometric mean]	7	8	140	873
[third quartile]	28	20	1000	5000
[standard deviation]	39	34	1624	3274
[skewness]	1.3	1.6	1.7	1.2
[kurtosis]	-0.2	1.1	1.7	0.2
inter-objective correlation	44.4%		11.5%	

Consistent with our hypothesis in the **Predictor Importance** subsection (as a sanity check), the selected number of PCs $\hat{d}$ mostly fell below 20; that these values are positively skewed is an artifact of the range of values with which we experimented. Much lower than the correlation for $\hat{d}$ at 44%, that of RMSE was only 12%. Evinced by the geometric mean $\hat{\nu}$ in the hundreds, we see that RF is really trying hard to reduce variance (such numbers are well-above the threshold above which small-sample statistical methods are typically employed on i.i.d. normal variables).

30-minute bars	τ: $\hat{d}$	RMSE: $\hat{d}$	τ: $\hat{\nu}$	RMSE: $\hat{\nu}$
[first quartile]	1	2	5	20
[geometric mean]	6	8	23	51
[third quartile]	20	50	100	100
[standard deviation]	30	32	71	148
[skewness]	1.9	1.5	0.9	2.0
[kurtosis]	2.4	0.9	-0.5	2.7
inter-objective correlation	20.3%		5.2%	

Opposite to the case in changing from 1-minute to 30 minute bars in EN, our inter-objective correlations for RF were lower with a longer bar duration, trading off sample size for robustness (despite a marginally smaller hyperparameter search range for $\hat{\nu}$). The geometric means of $\hat{d}$ remained nearly identical, while we have lower $\hat{\nu}$ as expected (albeit, almost mesokurtic when minimizing RMSE).

3.5.3 Stepwise k-Nearest Neighbors (SkNN)

We have coded the weight function as $w = 0$ for "uniform" and $w = 1$ for "distance". If the reader may forgive us for the abuse of notation in the right-most columns, we use $x_s, x_s + 1 (s = 1, 2, 3)$ to denote the selected predictor in the stepwise selection procedure; at each step s, the τ-selected PC has index x_{2s-1} and RMSE selects the x_{2s}-th PC.

1-minute bars	τ: $\hat{k}$	RMSE: $\hat{k}$	τ : $\hat{w}$	RMSE: $\hat{w}$	$\hat{x}_1$	$\hat{x}_2$	$\hat{x}_3$	$\hat{x}_4$	$\hat{x}_5$	$\hat{x}_6$
[mode]					1	9	2	2	10	
[first quartile]	10	500			16	8	13	7	12	12
[geometric mean]	56	507			26	4	25	18	29	20
[arithmetic mean]			39%	50%	43	27	42	35	46	35
[third quartile]	500	1000			72	43	72	61	84	57
[standard deviation]	334	316			31	28	33	31	35	34
[skewness]	1.4	0.0			0.3	1.2	0.3	0.6	0.2	1.0
[kurtosis]	0.6	-1.4			-1.2	0.1	-1.4	-1.0	-1.6	-0.7
inter-objective correlation	20.3%		5.2%							

The RMSE objective chooses 10x more nearest neighbors than τ, though the $\hat{k}$ vectors are weakly correlated. Selected weight functions were almost evenly divided between "uniform" and "distance", and the selections are barely correlated at all between objectives. The PCs selected were mostly low-indexed, though the trend with respect to S is hardly monotonic.

30-minute bars	τ: $\hat{k}$	RMSE: $\hat{k}$	τ : $\hat{w}$	RMSE: $\hat{w}$	$\hat{x}_1$	$\hat{x}_2$	$\hat{x}_3$	$\hat{x}_4$	$\hat{x}_5$	$\hat{x}_6$
[mode]					10	1	21	3	10	10
[first quartile]	5	20			28	10	21	15	18	10
[geometric mean]	16	39			39	21	35	29	30	25
[arithmetic mean]			38%	53%	54	41	47	43	43	35
[third quartile]	50	50			79	68	76	71	72	48
[standard deviation]	136	66			30	31	29	30	30	27
[skewness]	2.6	4.0			-0.2	0.1	0.1	0.3	0.3	0.4
[kurtosis]	5.7	22.5			-1.2	-1.4	-1.3	-1.2	-1.5	-1.5
inter-objective correlation	-0.8%		-7.7%							

The hyperparamters selected between the two objectives were uncorrelated for $\hat{k}$ and even marginally negative for the $\hat{w}$. Both objectives recommended fewer nearest neighbors, although the distributions of $\hat{k}$ are leptokurtic (especially in RMSE minimization). Where the same predictor is selected at later stages, it means that we did not find that adding any new predictor would improve the fitted model. Overall, later PCs were selected than in the previous scenario with 1-minute bars.

3.6 Performance Metrics

From a bird's eye view, we compare the algorithms against each other. The arithmetic mean on each metric is an average over all response variables. The last two rows per table represent for what percentage of the responses an algorithm scored the best on each metric. We use R^2 here in place of RMSE for cognitive consonance with τ (generally, higher performance scores are better).

3.6.1 1-Minute Bars

1-minute bars	EN	SkNN	RF
arithmetic mean(τ)	1.3%	2.6%	1.9%
arithmetic mean(R^2)	0.8%	2.5%	1.6%
percent_argmax(τ)	11.4%	76.1%	12.5%
percent_argmax(R^2)	8.0%	63.6%	28.4%

3.6.2 30-Minute Bars

30-minute bars	EN	SkNN	RF
arithmetic mean(τ)	3.0%	12.1%	6.1%
arithmetic mean(R^2)	0.8%	4.6%	2.2%
percent_argmax(τ)	0.0%	98.9%	1.1%
percent_argmax(R^2)	0.0%	98.9%	1.1%

3.6.3 Summary

For both 1-minute and 30-minute bars, SkNN's performance was superior, followed by RF, and then EN. Per response variable, the same ordering applies; this disparity is more pronounced with longer duration bars, with SkNN taking a clear lead above the other two algorithms.

Concidentally, one might (haphazardly) conjecture that our ranking in these terms also corresponds to the actual extent of predictor selection. This intuition accords with our observations earlier in viewing the variance and importance of PCs; the relationship holds typically though not always, as occasionally the most useful signal can be hidden in higher-order PCs. SkNN is restricted to at most 6 PCs and often refuses to add more in the 2nd and 3rd steps of (forward) stepwise selection. RF chooses which predictors to use at each split (from a random sample of $\sqrt{d} \leq 10$ PCs, where each subset is equally likely to be chosen). Empirically, we found EN (partway between LASSO and ridge regression) to perform the worst.

Yet another plausible explanation could be the existence and degree of nonlinearities modelled – it would be strenuous to make any conclusive claims in terms of recommendations on methodology since we have neither sufficient evidence nor proof of causation.

4 Trading Strategy and Performance Evaluation

4.1 Trading Strategy

We describe an active, compounding, long-only (or hold cash, but no short-selling) strategy that, if any trades are deemed sufficiently attractive at the end of each bar, chooses one security (with the highest forecasted profit) to enter a long position of q shares ($q = \left\lfloor \frac{I_t}{a_t} \right\rfloor$ where I_t is our present amount of capital in bar t, $1 \leq t \leq n$, and a_t is the ask price), hold until the end of the next bar, and then sell all q shares at the bid. For consideration, fix an algorithm and objective (henceforth "AO") pair from {EN, SkNN, RF} $\times$ {minimize RMSE, maximize correlation}.

How attractive is "sufficient"? That will be a metaparameter of the trading strategy, and we name it the "cutoff profitability", $\pi_0 \in \mathbb{R}$. Intuitively, we should to restrict $\pi_0 > 0$.

4.1.1 Forecasted Profit

In each bar, for each security, we predict its two response variables (return on midpoint price and return on spread). Let a_1 and b_1 denote the ask and bid prices at the end of the first bar (the present time), a_2 and b_2 similarly for the end of the next bar (from now until the bar's duration later); $m_1 := \frac{a_1+b_1}{2}$ is the present midpoint price, m_2 the future; $s_1 := \frac{b_1-a_1}{2}$ the spread (in absolute magnitude) and likewise for s_2; and $\hat{\delta}_y := (\hat{y}_2 - \hat{y}_1)$ where y is either m or s.

The (actual) profit can be formulated:

$$\pi = b_2 - a_1 = (m_2 - \tfrac{s_2}{2}) - (m_1 + \tfrac{s_1}{2}) = (m_2 - m_1) - \tfrac{s_2+s_1}{2}$$

If we hold until the end of a bar, the forecasted profit can be expressed in terms of the presently known s_1 along with our forecasted returns on midpoint price $\hat{\delta}_m$ and spread $\hat{\delta}_s$:

$$\hat{\pi} = \hat{\delta}_m - \tfrac{2s_1+\hat{\delta}_s}{2}$$

4.1.2 Security Rotation

Therefore, at the end of each bar, we forecast the profit over the next bar. Denoting the forecasted profitability of security j as $\hat{\pi}_j$ ($j = 1, \ldots, 44$), then we buy security indexed $j^* :=$argmax$(\hat{\pi}_j)$ if max$(\hat{\pi}_j)$ $> \pi_0$; otherwise, we hold cash.

4.1.3 Fee Structure

There are two costs we include in our evaluation. First, active trading incurs the cost of crossing the spread; each time we trade, we pay the arithmetic mean spread $\frac{s_2+s_1}{2}$ which is differenced from our return on the midpoint price $m_2 - m_1$ as explained in the previous subsection.

Second, the SEC charges \$23.10 per \$1,000,000 of principal (rounded up to the nearest penny) [34]. The latter is around 0.2 basis points (pips), which is negligent compared to the former; even highly liquid forex majors usually have a spread of 1-2 pips in non-volatile conditions.

4.1.4 Other Costs

FINRA also charges a TAF fee at \$0.000119 per share traded [11]; since this is fairly cheap compared to the price of stocks, and many liquid securities have a variety of ETFs with different prices, we will exclude it from our analysis.

In reality, we cannot assume that orders will be executed at instanteously. It takes time to send the order, have it directed by one's broker, and for the exchange to process it. The price we receive (for which there is also a delay) and the price at which it executes can be different, hence live trading involves slippage (which may be positive or negative depending on "luck", though in the long run it is expected to erode the profits of an otherwise good strategy, especially at higher frequencies) [2].

Third, when we engage in sufficiently high frequency and large volume trading, our interactions with the market affect other participants' behavior. Hence, a more realistic evaluation of profitability would include either a market impact model or involve live testing (typically with smaller amounts of investment at first, and then gradually scaling up).

4.1.5 Benchmark Model

We will benchmark against a "naive copycat" strategy which satisfies two properties: 1) it trades as often as the actual AO (being in the market for the same proportion of the time), and 2) it trades each of the 44 securities with the same preference proportions (in equal relative frequency) as AO. But it differs in that it does not know which security to hold, so it buys and holds a portfolio containing all of them with the overall weight determined by the actual AO's frequencies.

More precisely, suppose that some AO has a position in the market $100p$ percent of the time. Let R be the (row) vector of each security's total return over the entire period (during all trading hours in our filtered dataset), so each security has total return

$$r_j = \frac{P_n}{P_0} \in [-100\%, \infty)$$

where P_0 and P_n are the initial and ending prices, for each $j = 1, ..., 44$.

Then let Φ be the (column) vector of AO's trading proportions (in the manner of partitioning a unit, i.e. $0 \leq \phi_j \leq 1$ and $\sum_{j=1}^{44} \phi_j = 1$). We will denote the benchmark model's total "base return" by r_β (once again, apologies for the abuse of notation):

$$r_\beta := pR\Phi = p \sum_{j=1}^{44} \phi_j r_j$$

4.2 Evaluation

In this subsection, we will first briefly motivate performance metrics for comparing AOs using the described trading strategy. Then for each evaluation dataset, we examine performance in three frameworks in order to show cost contributions. Let "AOC" denote AO-cutoff metaparameter pair (AO, π_0).

In the first framework (FW1), we choose π_0 to maximize relative total return without the cost of spread (i.e. tuned for simulated passive trading with priority in queue, being always first in line), but with fees paid to the SEC; second (FW2), we use the π_0 optimized for maximal relative total return as before, but then calculate performance including cost of spread and the SEC's fees (i.e. tuned for passive trading, traded actively); third (FW3), we choose π_0 to directly maximize total profit including the cost of spread and SEC fees (i.e. tuned for active trading, actively traded).

To be clear, recall that we tuned the model on the training subset, with hyperparameters selected by evaluation on the validation subset. We also use the validation subset to compare profit as dependent on AOCs. Finally, we use one selected AOC to re-evaluate profit on the test subset.

4.2.1 Performance Metrics

- Number of Trades (T): This is the total number of bars in which any security is bought and held (as opposed to being cash neutral). With n as the total number of bars, we have $0 \leq T \leq n$. An AOC with small $T \approx 0$ has not made full use of the signal and will have statistics with high standard error. The optimal AOC will have $T \approx qn$, where q is the proportion of all bars which have profitable trades. Assuming that $q << 1$, an AOC with $T \approx n$ has many false positives (trades forecasted to be profitable, but are actually not. Because trades with close to zero profitabiliy are expensive (as we pay transaction costs each trade), it is better to "under-trade" than trade excessively frequently when $\hat{\pi}_t \approx 0$ in bar t; a realistic desirable AOC should have $0 << T << n$.

- Annualized Base Return (ABR): Since each evaluation subset (validation and test) are two months long, a total base return of r_β over two months becomes compounded into a total return of $ABR := (1 + r_\beta)^6 - 1$.

- Annualized Relative Return (ARR): This is the annualized total return of the strategy subtracted by the annualized base return. So if total return over two months is r_Σ, then annualized relative return is $ARR := (1 + r_\Sigma)^6 - ABR$. ARR is not synonymous with "excess" return, which is the difference relative to a "risk-free" investment.

- Relative Return per Trade (RRT): This is the simple fraction $RRT := \frac{r_\Sigma - r_\beta}{T}$. RRT is different from $(r_\Sigma - r_\beta)^{1/T}$.

- Maximum Drawdown (MD): With I_t dollars of capital at time t, $MD := \min\{I_t/I_0 - 1, 0 \leq t \leq n\}$. Assuming no borrowing, $MD \geq -100\%$.

- Sharpe Ratio (SAR) [32]: Denoting $s_r^2 = \frac{1}{T-1}\sum_{t=1}^{T}(r_t - \bar{r})^2$ for sample variance of the AOC's returns, we define $SAR := RRT/s_r$.

- Sortino Ratio (SOR) [28]: Let $R_{<0} := \{r_t \mid r_t < 0, 1 \leq t \leq T\}$. The version of the Sortino ratio we use shall be defind as $SOR := \frac{RRT}{s_{R_{<0}}}$, where $s_{R_{<0}}$ can be intuitively interpreted as "downside sample standard deviation".

In the tables below, the first column indicates our metaparameter tuning and trading framework (in the first row of each subtable), followed by rows for performance metric. The other columns are AOC pairs.

As a heads up, we will leave blank cells where there are no valid values (e.g. if $T = 0$, or SAR and SOR with $T \leq 1$). Unless there is a tie, the "best" value in each row of a table for the validation subset will be shaded blue and the worst in red.

4.2.2 1-Minute Bars (Validation)

FW1 (passive → passive)	(EN, τ)	(EN, R^2)	(SkNN, τ)	(SkNN, R^2)	(RF, τ)	(RF, R^2)
number of trades (T)	22	37	726	9	5340	121
annualized base return (ABR)	-0.01%	0.00%	0.79%	0.00%	15.10%	-0.03%
annualized relative return (ARR)	9.15%	1.35%	1.52%	0.66%	17.45%	2.00%
relative return per trade (RRT)	0.0668%	0.0060%	0.0003%	0.0122%	0.0005%	0.0027%
maximum drawdown (MD)	-0.00%	-0.00%	-0.56%	-0.00%	-0.58%	-0.00%
Sharpe ratio (SAR)	0.33	0.32	0.01	0.42	0.01	0.16
Sortino ratio (SOR)	0.70	0.58	0.01	0.84	0.01	0.28

FW2 (passive → active)	(EN, τ)	(EN, R^2)	(SkNN, τ)	(SkNN, R^2)	(RF, τ)	(RF, R^2)
number of trades (T)	22	36	724	8	5335	121
annualized base return (ABR)	-0.01%	0.00%	0.79%	0.00%	15.10%	-0.03%
annualized relative return (ARR)	4.09%	-5.50%	-99.18%	-3.16%	-100.00%	-24.24%
relative return per trade (RRT)	0.0304%	-0.0261%	-0.0761%	-0.0667%	-0.0171%	-0.0374%
maximum drawdown (MD)	-0.00%	-0.94%	-54.98%	-0.58%	-89.04%	-4.53%
Sharpe ratio (SAR)	0.15	-1.06	-0.53	-0.47	-0.23	-1.50
Sortino ratio (SOR)	0.32	-1.33	-0.53	-0.42	-0.32	-1.69

FW3 (active → active)	(EN, τ)	(EN, R^2)	(SkNN, τ)	(SkNN, R^2)	(RF, τ)	(RF, R^2)
number of trades (T)	22	1	724	0	69	0
annualized base return (ABR)	-0.01%	0.00%	0.79%		0.39%	
annualized relative return (ARR)	4.09%	0.05%	-99.18%		-16.76%	
relative return per trade (RRT)	0.0304%	0.0088%	-0.0761%		-0.0436%	
maximum drawdown (MD)	-0.00%	-0.00%	-54.98%		-2.95%	
Sharpe ratio (SAR)	0.15		-0.53		-0.55	
Sortino ratio (SOR)	0.32		-0.53		-0.76	

The validation subset contains n =15,560 1-minute bars in total. Switching to a different objective (τ or R^2) changes the selected π_0. Although (RF, τ) is yields the highest ABR in FW1, its risk-adjusted performance (SAR and SOR) are not impressive, and its ARR in FW2 and FW3 are negative due to the costs of crossing the spread in active trading. EN finds positive ARR in FW3, but it trades too infrequently ($T = 22$) to be of significant interest to us. Even when optimizing π_0 for active trading (in FW3), one tends to lose money due to the cost of spread.

4.2.3 30-Minute Bars (Validation)

FW1 (passive → passive)	(EN, τ)	(EN, R^2)	(SkNN, τ)	(SkNN, R^2)	(RF, τ)	(RF, R^2)
number of trades (T)	39	137	382	16	440	438
annualized base return (ABR)	0.34%	4.46%	3.93%	0.56%	11.93%	-2.32%
annualized relative return (ARR)	10.68%	18.19%	-1.84%	4.63%	19.51%	125.27%
relative return per trade (RRT)	0.0437%	0.0206%	-0.0008%	0.0474%	0.0069%	0.0331%
maximum drawdown (MD)	-0.00%	-0.04%	-1.73%	-0.51%	-1.62%	-0.00%
Sharpe ratio (SAR)	0.26	0.21	-0.01	0.16	0.04	0.12
Sortino ratio (SOR)	0.53	0.45	-0.01	0.19	0.05	0.16

FW2 (passive → active)	(EN, τ)	(EN, R^2)	(SkNN, τ)	(SkNN, R^2)	(RF, τ)	(RF, R^2)
number of trades (T)	39	135	382	16	440	438
annualized base return (ABR)	0.34%	4.32%	3.80%	0.56%	11.93%	-2.56%
annualized relative return (ARR)	-5.43%	-9.57%	-79.74%	-7.20%	-63.18%	12.56%
relative return per trade (RRT)	-0.0237%	-0.0123%	-0.0612%	-0.0774%	-0.0349%	0.0046%
maximum drawdown (MD)	-1.00%	-1.98%	-22.74%	-1.18%	-13.44%	-0.07%
Sharpe ratio (SAR)	-0.14	-0.13	-0.39	-0.26	-0.20	0.02
Sortino ratio (SOR)	-0.23	-0.24	-0.49	-0.39	-0.25	0.02

FW3 (active → active)	(EN, τ)	(EN, R^2)	(SkNN, τ)	(SkNN, R^2)	(RF, τ)	(RF, R^2)
number of trades (T)	4	0	26	0	1	305
annualized base return (ABR)	0.05%		0.39%		0.03%	-0.91%
annualized relative return (ARR)	-0.62%		-8.80%		1.27%	15.59%
relative return per trade (RRT)	-0.0261%		-0.0586%		0.2098%	0.0080%
maximum drawdown (MD)	-0.15%		-1.54%		-0.00%	-0.00%
Sharpe ratio (SAR)	-0.25		-0.28			0.02
Sortino ratio (SOR)	-0.36		-0.30			0.03

With 30-minute bars, we have $n = 480$. SkNN performs better (or at least not as bad) in most ABR, ARR, and RRT rows. (RF, R^2) looks spectacularly appealing in FW1 and FW2, with a nice ARR of 15.59% in FW3 (though its SAR and SOR are mediocre). Time after time, we witness the enormous costs of active trading.

4.2.4 30-Minute Bars (Test)

selected → passive trading	(RF, R^2)
number of trades (T)	422
annualized base return (ABR)	2.51%
annualized relative return (ARR)	-27.65%
relative return per trade (RRT)	-0.0124%
maximum drawdown (MD)	-5.29%
Sharpe ratio (SAR)	-0.08
Sortino ratio (SOR)	-0.09

selected → active trading	(RF, R^2)
number of trades (T)	422
annualized base return (ABR)	2.51%
annualized relative return (ARR)	-76.67%
relative return per trade (RRT)	-0.0510%
maximum drawdown (MD)	-21.12%
Sharpe ratio (SAR)	-0.32
Sortino ratio (SOR)	-0.36

We only show the selected AOR (RF, R^2). The test subset also has $n = 480$ bars in total. Note that in the active trading case, T has increased from 305 to 422; ergo, we are trading more frequently than hoped – and it appears that some extra unwanted trades have made our ARR quite negative (if it had vanished, then our absolute return would be positive because $ABR > 0$). Overfitting has cost us an arm and a leg.

5 Algorithmic Trading System (ATS)

5.1 Robinhood API

5.1.1 Requests

Robinhood's API offers a secure connection via the HTTPS protocol [17, 29]. Token authentication is necessary for account operations and helps prevent against man-in-the-middle attacks.

Since there is no publicly available web socket or information on maximum poll rates, one could either directly contact Robinhood Financial (preferred) or would need to experiment (and perhaps occasionally change one's IP address or use VPNs) to determine the threshold.

In theory, one could find the maximum threshold rate by a variant on binary search. We define a (piecewise constant) function $h : \mathbb{N} \to [0, 1]$ by $h(u) := 1$ if one gets throttled at a frequency u (requests per unit time) and otherwise $h(u) := 0$. We could memoize h in a hash table. Then we call (any equivalent of) C++'s `lower_bound` algorithm from `STL`, in search for the value $c = 1$ (technically, the discrete representation of any real number $c \geq 1$ would work). In logarithmic time, `lower_bound` finds $u^* :=\min\{u \mid u_0 \leq u < u_1, h(u) \geq c\}$ for some prespecified search range marked by $u_0, u_1 \in \mathbb{N}$. To find an appropriate u_0, one could start at some high frequency such as 10 Hz and halve the frequency until one finds that $h(u_0) = 0$; necessarily, $u_0 < u^* (\implies u_0 \leq u^*)$. Then one one starts at $u_1 := u_0$ and doubles the frequency until one finds that $h(u_1 + 1) = 1$; it is true that $u^* \leq u_1 + 1 (\implies u^* < u_1)$. Thus, we run (an implementation of) `lower_bound` to find u^*, whence the maximum (discrete) threshold we want is $u^* - 1$.

5.1.2 Order and Risk Management

The purpose of executing our ATS' strategy demands three heavily used API calls, with URI relative to `api.robinhood.com/`:

URI	Method	Comments
orders/	GET	check order status, including number of shares filled
orders/	POST	send new order to buy or sell (limit orders available)
quotes/	GET	get current {ask, bid} × {price, size}

To control maximum (theoretical – barring exchange meltdown) loss, we can use stop-loss orders. Essentially, when we send our buy order at some price p_t as one is wont to do, we simultaneously send an accompanying limit sell order that does not trigger immediately as it has an execution price (e_t) lower than a stop price (s_t) which is lower than the current ask price (a_t); that is, $e_t < s_t < a_t$. Then if the stock's market price falls below the stop price s_t at time t', the stop-loss order automatically executes (without network latency from our server, since it is stored on the exchange) and liquidates our current position at no lower than $\max(e_t, b_{t'})$ where $b_{t'}$ is the new bid price. Of course, another way to hedge our bet is to buy a put option.

To be extra safe (e.g. in the case of Internet connection loss), we can set up multiple servers from different locations and ISPs that are connected in a fully-connected network topology. When one server is disconnected, the others will detect the failure and either elect one of the remaining (for instance, with lowest packet loss or RTT ping) as substitute; or if the resulting bus factor is too low, then we send an order to urgently liquidate all positions on Robinhood [21]. In any case, the event should be logged, and a warning ought to be raised for more pressing emergencies.

5.2 Web Server

5.2.1 Smartphone Remote Control

It is possible to control the ATS (as a multi-threaded, online server) by mobile smartphone. The ATS (which can be first tested on `localhost` using `ngrok`) connects to both Robinhood and Twilio's web server

(via API), and the latter is able to transmit instant messages (SMS) between the ATS and one or more recipients' phones (with authorized numbers) [3].

$$\text{US stock exchanges} \longleftrightarrow \text{HFT flow traders} \longleftrightarrow \text{Robinhood} \longleftrightarrow \text{ATS} \longleftrightarrow \text{Twilio} \longleftrightarrow \text{user's phone}$$

Upon executing an order (by receiving an HTTP response from Robinhood), the ATS can make a request to Twilio which will send an instant message to the user. The user can text replies to Twilio number linked with their account whose configurations has the ATS's web address stored; Twilio sends another HTTP request to the ATS, which asynchronously processes the request (equipped with its internal database), then responds to Twilio; Twilio sends a text message back to the user.

5.2.2 Additional Services

Although users can use the official Robinhood app to check the markets along with all other account information, for simplicity of synchronization (i.e. ease of information management and avoiding network race conditions), it might be more convenient to trade by texting.

By default, Robinhood alerts users with a push notification the price of a stock currently held has changed more than 5% since the open, we may wish to set up more elaborate alert systems. As an example of a more specific rule, if a "major" relevant news event (with many tweets and overrepresented keywords appearing in news sources such as Bloomberg) occurs and the price changes by more than the three times the historical daily standard standard deviation over the past month, then the user receives a special alert that a trade will be entered unless they press cancel within 15 seconds [22]. Alternatively, according to dynamically changing market conditions, the ATS can switch between different strategies or adapt its risk management.

It would be ideal for a retail trader to operate their ATS on a cloud hosting service, such as Amazon's AWS or Google's GCP, allowing easier remote colocation and even the possibility of a high-uptime server farm over multiple geographic regions [1]. The user's personal computer is freed from ATS obligations.

6 Conclusion

6.1 Reflections

6.1.1 Data and Preprocessing

There are free tick data sources limited in depth of book and volume of historical data. One way of dealing with multiple tick series is to aggregate their features over regular bars, which at finer granularities necessitates imputation. Global securities across asset classes have different trading times (and intraday volume patterns), which requires due treatment (as we have done methodologically by histogram and dendrogram anaysis, followed by restricting the trading window).

On our dataset, the rapidly decreasing trend of PVE per PC (with PCA) corresponded with that of predictor importance, and under computational constraints as well as for the sake of parsimony it can pay off to limit the number of PCs. It is useful to scale features before applying PCA and convenient to rescale afterward (especially where numerical stability is an issue). Prior knowledge on fundamental economic relationships helps.

6.1.2 Hyperparameter Selection

Among the hyperparameters we experimented with in selecting model families, some of the more vital to tune are EN's λ, RF's ν, and k in kNN. On the other hand, λ in EN and the weight function in kNN are relatively supplementary. The number of trees in RF and steps in SkNN can be increased until marginal improvement converges approximately to zero. Despite RF being able to pick predictors at each split, when many predictors are insignificant, RF can perform even better if they are filtered beforehand.

Within the context of financial markets, even seemingly similar assets of the same class can select different hyperparameters. Even if one presupposes that multiple securities are similar (by individual properties as well as causal connections in asset markets), it is worth checking to make sure.

6.1.3 Performance Evaluation

To have multiple, sufficiently large data subsets is essential to guard against the trap of overfitting on highly noisy data. Recommended sample sizes for high-frequency financial data greatly exceeds those of i.i.d. datasets with strong signals.

We recommend accounting for and measuring as many known trading costs as possible early in the tuning pipeline because our anecdotes demonstrate that assumed proxies can easily break down from one stage to the following. Forecasting near and extrapolation beyond tails can place higher robustness pressures on estimators, and in-sample statistics can be completely disparate from their out-of-sample counterparts. Examining various performance metrics helps elucidate more holistic modelling diagnosis and subject matter interpretation.

6.2 Future Work

6.2.1 Environment

Buying a GPU, more RAM, and a solid state hard drive can accelerate data science iterations. Even though a person could supposedly multi-task, some decisions about what to do afterward depend on results currently being computing.

The implementations in `sklearn` are not terribly efficient. There are some packages in R whose algorithms offer more hyperparamater arguments and run much faster in practice on larger datasets.

6.2.2 Level II Data

The data acquired in this project only included top-of-book quotes. There is typically much deeper liquidity further down, and different features can be generated from a thicker orderbook.

6.2.3 Time-of-Day (ToD) Extension

If we fix a security whose movement we wish to predict, then we can include all securities whose trading patterns cover the former's ToD. In a directed coverage graph, we can look at all the securities with edges pointing to the response's node. The classic algorithm for an activity-on-vertex (AoV) network is suitable.

Different derivatives markets – such as those of the CME Group – offer more trading hours. Advanced portfolio hedging strategies arise from combinations of spot, futures, options, and swaps.

6.2.4 Response Variable(s)

More response variables extends to a larger data matrix. This is common in factor-based models used for stock selection, on which hundreds to thousands of stocks are compared. There marginal gains to adding more securities decreases though, especially if additional securities are highly correlated or structurally similar to those already in the portfolio.

The disadvantage of applying one model to all securities is that we could end up overlooking valuable idiosyncrasies. Still, we could further refine our set of securities under consideration by analyzing financial fundamentals and heuristics such as skewness of returns and volatility-to-spread ratios. It is vastly more straightforward to scrutinize one particular response, signal, or relationship in detail than to data mine over combinatorially many, then struggle to extract specific information below the surface, and concisely summarize all insights.

6.2.5 Problem Formulation

Instead of forecasting a real-valued quantity, we could classify whether a value (i.e. a security's price) will increase or decrease; a several-bins approach is viable too. Binary classification would intuitively lend to other perspectives, such as confusion matrix and ROC curve analyses. Of course with classification, we could specify specify a loss matrix. We did not use L1 loss in regression this time for computational reasons, but it is more robust than L2 and consistent with risk-neutral preferences.

6.2.6 Decomposition

We did not conduct any deseasonalization. From our discussion on ToD, we pointed out that there are archetypal U-curves in trading activity plots for many US equity markets. Spectral analysis and partial autocorrelation plots are also great tools. Four components should be kept: level, trend, seasononality, and residuals. We estimate the former three and predict the last.

6.2.7 Ensemble Models

It may happen that we have a number of weak learners, but by combining their outputs, the aggregate prediction surpasses that of any individual model. For example, any arbitrarily-complicated periodic function can be expressed as the infinite series of sine and cosine terms. More pathologically, consider a true model which is the horizontal x-axis ($y = 0$); the Weierstrass function $W(x)$ would be a terrible fit, but adding $W(x)$ to the negative of the Weierstrass function, $-W(x)$, gives us exactly what we want.

6.2.8 Evaluation

Some simple vector autoregressive models can be used as alternative benchmarks. There are other dimensions along which we can view performance and error metrics, such as grouping by geographic region or conditioning upon time-of-day.

7 References

[1] Ahuja, S. P., Furman, T. F., Roslie, K. E., & Wheeler, J. T. (2013). Empirical Performance Assessment of Public Clouds Using System Level Benchmarks. International Journal of Cloud Applications and Computing (IJCAC), 3(4), 81-91.

[2] Aldridge, I. (2013). High-frequency trading: a practical guide to algorithmic strategies and trading systems (Vol. 604). John Wiley & Sons.

[3] Alagappan, R., & Das, S. Uncovering Twilio: Insights into Cloud Communication Services.

[4] Bailey, D., Borwein, J., Lopez de Prado, M., & Zhu, Q. (2014). Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-sample performance.

[5] Benjamin, R., & Wigand, R. (1995). Electronic markets and virtual value chains on the information superhighway. Sloan Management Review, 36(2), 62.

[6] Bowen, D., Hutchinson, M., & O'Sullivan, N. (2010). High frequency equity pairs trading: transaction costs, speed of execution and patterns in returns.

[7] Chordia, T., Roll, R., & Subrahmanyam, A. (2008). Liquidity and market efficiency. Journal of Financial Economics, 87(2), 249-268.

[8] Dimson, E., & Mussavian, M. (1998). A brief history of market efficiency. European financial management, 4(1), 91-103.

[9] Domingos, P. (2012). A few useful things to know about machine learning. Communications of the ACM, 55(10), 78-87.

[10] Dukascopy Bank SA. (n.d.). Historical Data Feed. Retrieved from https://www.dukascopy.com/swiss/english/marketwatch/historical/

[11] Financial Industry Regulatory Authority. (n.d.). Section 1 — Member Regulatory Fees. Retrieved from http://finra.complinet.com/en/display/display_main.html?rbid=2403&element_id=4694

[12] Fischetti, T. (2015, May 31). Lessons learned in high-performance R. Retrieved from http://www.onthelambda.com/2015/05/31/lessons-learned-in-high-performance-r/

[13] Fry, J., & Cheah, E. T. (2016). Negative bubbles and shocks in cryptocurrency markets. International Review of Financial Analysis, 47, 343-352.

[14] Giuseppe. (2017, August 6). Duka - Dukascopy historical data downloader. Retrieved from https://github.com/giuse88/duka

[15] HistData.com. (n.d.). Data Files: Detailed Specification. Retrieved from http://www.histdata.com/f-a-q/data-files-detailed-specification/

[16] Karolyi, G. A., & Stulz, R. M. (2003). Are financial assets priced locally or globally?. Handbook of the Economics of Finance, 1, 975-1020.

[17] Kelly, J. (2018, April 24). Python Framework to make trades with Robinhood Private API. Retrieved from https://github.com/Jamonek/Robinhood

[18] Lo, A. W. (2004). The adaptive markets hypothesis: Market efficiency from an evolutionary perspective.

[19] MacKenzie, D. (2014). A sociology of algorithms: High-frequency trading and the shaping of markets. Unpublished paper.

[20] Michener, C. D., & Sokal, R. R. (1957). A quantitative approach to a problem in classification. Evolution, 11(2), 130-162.

[21] Monperrus, M. (2017, April). Principles of antifragile software. In Companion to the first International Conference on the Art, Science and Engineering of Programming (p. 32). ACM.

[22] Mullins, B., Rothfeld, M., McGinty, T., & Strasburg, J. (2013). Traders pay for an early peek at key data. Wall Street Journal.

[23] Poleszczuk, J. (2015, September 26). Java, C , MATLAB, Julia or Python for cellular automatons? Speed comparison. Retrieved from https://computecancer.wordpress.com/2015/09/26/java-c-matlab-or-python-for-cellular-automatons-speed-comparison/

[24] Robinhood. (n.d.). Retrieved from https://www.robinhood.com/options/

[25] Robinhood Crypto Trading Is Here. (n.d.). Retrieved from http://blog.robinhood.com/news/2018/2/21/robinhood-crypto-trading-is-here

[26] Robinhood - Disclosure Library. (n.d.). Retrieved from https://robinhood.com/legal/

[27] Robinhood Help Center - Order Types. (n.d.) Retrieved from https://support.robinhood.com/hc/en-us/articles/208650386-Order-Types

[28] Rollinger, T. N., & Hoffman, S. T. (2013). Sortino: A 'Sharper' Ratio. Chicago, IL: Red Rock Capital. http://www.redrockcapital.com/assets/RedRock_Sortino_white_paper.pdf.

[29] Robinson, S. (2018, April 18). Unofficial Documentation of Robinhood Trade's Private API. Retrieved from https://github.com/sanko/Robinhood

[30] Saffi, P. A., & Sigurdsson, K. (2010). Price efficiency and short selling. The Review of Financial Studies, 24(3), 821-852.

[31] Sarantis, N. (2001). Nonlinearities, cyclical behaviour and predictability in stock markets: international evidence. International Journal of Forecasting, 17(3), 459-482.

[32] Sharpe, W. F. (1994). The sharpe ratio. Journal of portfolio management, 21(1), 49-58.

[33] Timmermann, A., & Granger, C. W. (2004). Efficient market hypothesis and forecasting. International Journal of forecasting, 20(1), 15-27.

[34] U.S. Securities and Exchange Commission. (2018, April 17). Fee Rate Advisory #3 for Fiscal Year 2018 [Press release]. Retrieved from https://www.sec.gov/news/press-release/2018-67