

# Table of Contents

<b>Preface</b>	<b>xiii</b>
<b>Chapter 1: Machine Learning for Trading – From Idea to Execution</b>	<b>1</b>
<b>The rise of ML in the investment industry</b>	<b>2</b>
From electronic to high-frequency trading	3
Factor investing and smart beta funds	5
Algorithmic pioneers outperform humans	7
ML and alternative data	10
Crowdsourcing trading algorithms	11
<b>Designing and executing an ML-driven strategy</b>	<b>12</b>
Sourcing and managing data	13
From alpha factor research to portfolio management	13
Strategy backtesting	15
<b>ML for trading – strategies and use cases</b>	<b>15</b>
The evolution of algorithmic strategies	15
Use cases of ML for trading	16
<b>Summary</b>	<b>19</b>
<b>Chapter 2: Market and Fundamental Data – Sources and Techniques</b>	<b>21</b>
<b>Market data reflects its environment</b>	<b>22</b>
Market microstructure – the nuts and bolts	23
How to trade – different types of orders	23
Where to trade – from exchanges to dark pools	24
<b>Working with high-frequency data</b>	<b>26</b>
How to work with Nasdaq order book data	26
Communicating trades with the FIX protocol	27
The Nasdaq TotalView-ITCH data feed	27
From ticks to bars – how to regularize market data	35
AlgoSeek minute bars – equity quote and trade data	40
<b>API access to market data</b>	<b>44</b>
Remote data access using pandas	44
yfinance – scraping data from Yahoo! Finance	46



Quantopian	48
Zipline	48
Quandl	50
Other market data providers	50
<b>How to work with fundamental data</b>	<b>51</b>
Financial statement data	51
Other fundamental data sources	56
<b>Efficient data storage with pandas</b>	<b>57</b>
<b>Summary</b>	<b>58</b>
<b>Chapter 3: Alternative Data for Finance – Categories and Use Cases</b>	<b>59</b>
<b>The alternative data revolution</b>	<b>60</b>
<b>Sources of alternative data</b>	<b>62</b>
Individuals	62
Business processes	63
Sensors	63
<b>Criteria for evaluating alternative data</b>	<b>65</b>
Quality of the signal content	65
Quality of the data	67
Technical aspects	68
<b>The market for alternative data</b>	<b>69</b>
Data providers and use cases	70
<b>Working with alternative data</b>	<b>72</b>
Scraping OpenTable data	72
Scraping and parsing earnings call transcripts	77
<b>Summary</b>	<b>80</b>
<b>Chapter 4: Financial Feature Engineering – How to Research</b>	<b>81</b>
<b>Alpha Factors</b>	<b>81</b>
<b>Alpha factors in practice – from data to signals</b>	<b>82</b>
<b>Building on decades of factor research</b>	<b>84</b>
Momentum and sentiment – the trend is your friend	84
Value factors – hunting fundamental bargains	88
Volatility and size anomalies	90
Quality factors for quantitative investing	92
<b>Engineering alpha factors that predict returns</b>	<b>94</b>
How to engineer factors using pandas and NumPy	94
How to use TA-Lib to create technical alpha factors	99
Denoising alpha factors with the Kalman filter	100
How to preprocess your noisy signals using wavelets	104
<b>From signals to trades – Zipline for backtests</b>	<b>106</b>
How to backtest a single-factor strategy	106
Combining factors from diverse data sources	109
<b>Separating signal from noise with Alphalens</b>	<b>111</b>
Creating forward returns and factor quantiles	112
Predictive performance by factor quantiles	113



The information coefficient	115
Factor turnover	117
<b>Alpha factor resources</b>	<b>118</b>
Alternative algorithmic trading libraries	118
<b>Summary</b>	<b>119</b>
<b>Chapter 5: Portfolio Optimization and Performance Evaluation</b>	<b>121</b>
<b>How to measure portfolio performance</b>	<b>122</b>
Capturing risk-return trade-offs in a single number	122
The fundamental law of active management	124
<b>How to manage portfolio risk and return</b>	<b>125</b>
The evolution of modern portfolio management	125
Mean-variance optimization	127
Alternatives to mean-variance optimization	131
Risk parity	134
Risk factor investment	135
Hierarchical risk parity	135
<b>Trading and managing portfolios with Zipline</b>	<b>136</b>
Scheduling signal generation and trade execution	137
Implementing mean-variance portfolio optimization	138
<b>Measuring backtest performance with pyfolio</b>	<b>140</b>
Creating the returns and benchmark inputs	141
Walk-forward testing – out-of-sample returns	142
<b>Summary</b>	<b>146</b>
<b>Chapter 6: The Machine Learning Process</b>	<b>147</b>
<b>How machine learning from data works</b>	<b>148</b>
The challenge – matching the algorithm to the task	149
Supervised learning – teaching by example	149
Unsupervised learning – uncovering useful patterns	150
Reinforcement learning – learning by trial and error	152
<b>The machine learning workflow</b>	<b>153</b>
Basic walkthrough – k-nearest neighbors	154
Framing the problem – from goals to metrics	154
Collecting and preparing the data	160
Exploring, extracting, and engineering features	160
Selecting an ML algorithm	162
Design and tune the model	162
How to select a model using cross-validation	165
How to implement cross-validation in Python	166
Challenges with cross-validation in finance	168
Parameter tuning with scikit-learn and Yellowbrick	170
<b>Summary</b>	<b>172</b>
<b>Chapter 7: Linear Models – From Risk Factors to Return Forecasts</b>	<b>173</b>
<b>From inference to prediction</b>	<b>174</b>



<b>The baseline model – multiple linear regression</b>	<b>175</b>
How to formulate the model	175
How to train the model	176
The Gauss–Markov theorem	179
How to conduct statistical inference	180
How to diagnose and remedy problems	181
<b>How to run linear regression in practice</b>	<b>184</b>
OLS with statsmodels	184
Stochastic gradient descent with sklearn	186
<b>How to build a linear factor model</b>	<b>187</b>
From the CAPM to the Fama–French factor models	188
Obtaining the risk factors	189
Fama–Macbeth regression	191
<b>Regularizing linear regression using shrinkage</b>	<b>194</b>
How to hedge against overfitting	194
How ridge regression works	195
How lasso regression works	196
<b>How to predict returns with linear regression</b>	<b>197</b>
Preparing model features and forward returns	197
Linear OLS regression using statsmodels	203
Linear regression using scikit-learn	205
Ridge regression using scikit-learn	208
Lasso regression using sklearn	210
Comparing the quality of the predictive signals	212
<b>Linear classification</b>	<b>212</b>
The logistic regression model	213
How to conduct inference with statsmodels	215
Predicting price movements with logistic regression	217
<b>Summary</b>	<b>219</b>
<b>Chapter 8: The ML4T Workflow – From Model to Strategy Backtesting</b>	<b>221</b>
<b>How to backtest an ML-driven strategy</b>	<b>222</b>
<b>Backtesting pitfalls and how to avoid them</b>	<b>223</b>
Getting the data right	224
Getting the simulation right	225
Getting the statistics right	226
<b>How a backtesting engine works</b>	<b>227</b>
Vectorized versus event-driven backtesting	228
Key implementation aspects	230
<b>backtrader – a flexible tool for local backtests</b>	<b>232</b>
Key concepts of backtrader's Cerebro architecture	232
How to use backtrader in practice	235
backtrader summary and next steps	239
<b>Zipline – scalable backtesting by Quantopian</b>	<b>239</b>



Calendars and the Pipeline for robust simulations	240
Ingesting your own bundles with minute data	242
The Pipeline API – backtesting an ML signal	245
How to train a model during the backtest	250
Instead of How to use	254
<b>Summary</b>	<b>254</b>
<b>Chapter 9: Time-Series Models for Volatility Forecasts and Statistical Arbitrage</b>	<b>255</b>
<b>Tools for diagnostics and feature extraction</b>	<b>256</b>
How to decompose time-series patterns	257
Rolling window statistics and moving averages	258
How to measure autocorrelation	259
<b>How to diagnose and achieve stationarity</b>	<b>260</b>
Transforming a time series to achieve stationarity	261
Handling instead of How to handle	261
Time-series transformations in practice	263
<b>Univariate time-series models</b>	<b>265</b>
How to build autoregressive models	266
How to build moving-average models	267
How to build ARIMA models and extensions	268
How to forecast macro fundamentals	270
How to use time-series models to forecast volatility	272
<b>Multivariate time-series models</b>	<b>276</b>
Systems of equations	277
The vector autoregressive (VAR) model	277
Using the VAR model for macro forecasts	278
<b>Cointegration – time series with a shared trend</b>	<b>281</b>
The Engle-Granger two-step method	282
The Johansen likelihood-ratio test	282
<b>Statistical arbitrage with cointegration</b>	<b>283</b>
How to select and trade comoving asset pairs	283
Pairs trading in practice	285
Preparing the strategy backtest	288
Backtesting the strategy using backtrader	292
Extensions – how to do better	294
<b>Summary</b>	<b>294</b>
<b>Chapter 10: Bayesian ML – Dynamic Sharpe Ratios and Pairs Trading</b>	<b>295</b>
<b>How Bayesian machine learning works</b>	<b>296</b>
How to update assumptions from empirical evidence	297
Exact inference – maximum a posteriori estimation	298
Deterministic and stochastic approximate inference	301
<b>Probabilistic programming with PyMC3</b>	<b>305</b>
Bayesian machine learning with Theano	305



The PyMC3 workflow: predicting a recession	305
<b>Bayesian ML for trading</b>	<b>317</b>
Bayesian Sharpe ratio for performance comparison	317
Bayesian rolling regression for pairs trading	320
Stochastic volatility models	323
<b>Summary</b>	<b>326</b>
<b>Chapter 11: Random Forests – A Long-Short Strategy for Japanese Stocks</b>	<b>327</b>
<b>Decision trees – learning rules from data</b>	<b>328</b>
How trees learn and apply decision rules	328
Decision trees in practice	330
Overfitting and regularization	336
Hyperparameter tuning	338
<b>Random forests – making trees more reliable</b>	<b>345</b>
Why ensemble models perform better	345
Bootstrap aggregation	346
How to build a random forest	349
How to train and tune a random forest	350
Feature importance for random forests	352
Out-of-bag testing	352
Pros and cons of random forests	353
<b>Long-short signals for Japanese stocks</b>	<b>353</b>
The data – Japanese equities	354
The ML4T workflow with LightGBM	355
The strategy – backtest with Zipline	362
<b>Summary</b>	<b>364</b>
<b>Chapter 12: Boosting Your Trading Strategy</b>	<b>365</b>
<b>Getting started – adaptive boosting</b>	<b>366</b>
The AdaBoost algorithm	367
Using AdaBoost to predict monthly price moves	368
<b>Gradient boosting – ensembles for most tasks</b>	<b>370</b>
How to train and tune GBM models	372
How to use gradient boosting with sklearn	374
<b>Using XGBoost, LightGBM, and CatBoost</b>	<b>378</b>
How algorithmic innovations boost performance	379
<b>A long-short trading strategy with boosting</b>	<b>383</b>
Generating signals with LightGBM and CatBoost	383
Inside the black box - interpreting GBM results	391
Backtesting a strategy based on a boosting ensemble	399
Lessons learned and next steps	401
<b>Boosting for an intraday strategy</b>	<b>402</b>
Engineering features for high-frequency data	402
Minute-frequency signals with LightGBM	404
Evaluating the trading signal quality	405



<b>Summary</b>	<b>406</b>
<b>Chapter 13: Data-Driven Risk Factors and Asset Allocation with Unsupervised Learning</b>	<b>407</b>
<b>Dimensionality reduction</b>	<b>408</b>
The curse of dimensionality	409
Linear dimensionality reduction	411
Manifold learning – nonlinear dimensionality reduction	418
<b>PCA for trading</b>	<b>421</b>
Data-driven risk factors	421
Eigenportfolios	424
<b>Clustering</b>	<b>426</b>
k-means clustering	427
Hierarchical clustering	429
Density-based clustering	431
Gaussian mixture models	432
<b>Hierarchical clustering for optimal portfolios</b>	<b>433</b>
How hierarchical risk parity works	433
Backtesting HRP using an ML trading strategy	435
<b>Summary</b>	<b>438</b>
<b>Chapter 14: Text Data for Trading – Sentiment Analysis</b>	<b>439</b>
<b>ML with text data – from language to features</b>	<b>440</b>
Key challenges of working with text data	440
The NLP workflow	441
Applications	443
<b>From text to tokens – the NLP pipeline</b>	<b>443</b>
NLP pipeline with spaCy and textacy	444
NLP with TextBlob	448
<b>Counting tokens – the document-term matrix</b>	<b>449</b>
The bag-of-words model	450
Document-term matrix with scikit-learn	451
Key lessons instead of lessons learned	455
<b>NLP for trading</b>	<b>455</b>
The naive Bayes classifier	456
Classifying news articles	457
Sentiment analysis with Twitter and Yelp data	458
<b>Summary</b>	<b>462</b>
<b>Chapter 15: Topic Modeling – Summarizing Financial News</b>	<b>463</b>
<b>Learning latent topics – Goals and approaches</b>	<b>464</b>
Latent semantic indexing	465
How to implement LSI using sklearn	466
Strengths and limitations	468
<b>Probabilistic latent semantic analysis</b>	<b>469</b>
How to implement pLSA using sklearn	470



Strengths and limitations	471
<b>Latent Dirichlet allocation</b>	<b>471</b>
How LDA works	471
How to evaluate LDA topics	473
How to implement LDA using sklearn	475
How to visualize LDA results using pyLDAvis	475
How to implement LDA using Gensim	476
<b>Modeling topics discussed in earnings calls</b>	<b>478</b>
Data preprocessing	478
Model training and evaluation	479
Running experiments	480
<b>Topic modeling for with financial news</b>	<b>481</b>
<b>Summary</b>	<b>482</b>
<b>Chapter 16: Word Embeddings for Earnings Calls and SEC Filings</b>	<b>483</b>
<b>How word embeddings encode semantics</b>	<b>484</b>
How neural language models learn usage in context	485
word2vec – scalable word and phrase embeddings	485
Evaluating embeddings using semantic arithmetic	487
<b>How to use pretrained word vectors</b>	<b>489</b>
GloVe – Global vectors for word representation	489
<b>Custom embeddings for financial news</b>	<b>491</b>
Preprocessing – sentence detection and n-grams	492
The skip-gram architecture in TensorFlow 2	493
Visualizing embeddings using TensorBoard	496
How to train embeddings faster with Gensim	497
<b>word2vec for trading with SEC filings</b>	<b>499</b>
Preprocessing – sentence detection and n-grams	500
Model training	501
<b>Sentiment analysis using doc2vec embeddings</b>	<b>503</b>
Creating doc2vec input from Yelp sentiment data	503
Training a doc2vec model	504
Training a classifier with document vectors	505
Lessons learned and next steps	507
<b>New frontiers – pretrained transformer models</b>	<b>507</b>
Attention is all you need	508
BERT – towards a more universal language model	509
Trading on text data – lessons learned and next steps	511
<b>Summary</b>	<b>511</b>
<b>Chapter 17: Deep Learning for Trading</b>	<b>513</b>
<b>Deep learning – what's new and why it matters</b>	<b>514</b>
Hierarchical features tame high-dimensional data	515
DL as representation learning	516
How DL relates to ML and AI	517
<b>Designing an NN</b>	<b>518</b>



A simple feedforward neural network architecture	519
Key design choices	520
How to regularize deep NNs	522
Training faster – optimizations for deep learning	523
Summary – how to tune key hyperparameters	525
<b>A neural network from scratch in Python</b>	<b>526</b>
The input layer	526
The hidden layer	527
The output layer	528
Forward propagation	529
The cross-entropy cost function	529
How to implement backprop using Python	529
<b>Popular deep learning libraries</b>	<b>534</b>
Leveraging GPU acceleration	534
How to use TensorFlow 2	535
How to use TensorBoard	537
How to use PyTorch 1.4	538
Alternative options	541
<b>Optimizing an NN for a long-short strategy</b>	<b>542</b>
Engineering features to predict daily stock returns	542
Defining an NN architecture framework	542
Cross-validating design options to tune the NN	543
Evaluating the predictive performance	545
Backtesting a strategy based on ensembled signals	547
How to further improve the results	549
<b>Summary</b>	<b>549</b>
<b>Chapter 18: CNNs for Financial Time Series and Satellite Images</b>	<b>551</b>
<b>How CNNs learn to model grid-like data</b>	<b>552</b>
From hand-coding to learning filters from data	553
How the elements of a convolutional layer operate	554
The evolution of CNN architectures: key innovations	558
<b>CNNs for satellite images and object detection</b>	<b>559</b>
LeNet5 – The first CNN with industrial applications	560
AlexNet – reigniting deep learning research	563
Transfer learning – faster training with less data	565
Object detection and segmentation	573
Object detection in practice	573
<b>CNNs for time-series data – predicting returns</b>	<b>577</b>
An autoregressive CNN with 1D convolutions	577
CNN-TA – clustering time series in 2D format	581
<b>Summary</b>	<b>589</b>
<b>Chapter 19: RNNs for Multivariate Time Series and Sentiment Analysis</b>	<b>591</b>
<b>How recurrent neural nets work</b>	<b>592</b>



Unfolding a computational graph with cycles	594
Backpropagation through time	594
Alternative RNN architectures	595
How to design deep RNNs	596
The challenge of learning long-range dependencies	597
Gated recurrent units	599
<b>RNNs for time series with TensorFlow 2</b>	<b>599</b>
Univariate regression – predicting the S&P 500	600
How to get time series data into shape for an RNN	600
Stacked LSTM – predicting price moves and returns	605
Multivariate time-series regression for macro data	611
<b>RNNs for text data</b>	<b>614</b>
LSTM with embeddings for sentiment classification	614
Sentiment analysis with pretrained word vectors	617
Predicting returns from SEC filing embeddings	619
<b>Summary</b>	<b>624</b>
<b>Chapter 20: Autoencoders for Conditional Risk Factors and Asset Pricing</b>	<b>625</b>
<b>Autoencoders for nonlinear feature extraction</b>	<b>626</b>
Generalizing linear dimensionality reduction	626
Convolutional autoencoders for image compression	627
Managing overfitting with regularized autoencoders	628
Fixing corrupted data with denoising autoencoders	628
Seq2seq autoencoders for time series features	629
Generative modeling with variational autoencoders	629
<b>Implementing autoencoders with TensorFlow 2</b>	<b>630</b>
How to prepare the data	630
One-layer feedforward autoencoder	631
Feedforward autoencoder with sparsity constraints	634
Deep feedforward autoencoder	634
Convolutional autoencoders	636
Denoising autoencoders	637
<b>A conditional autoencoder for trading</b>	<b>638</b>
Sourcing stock prices and metadata information	639
Computing predictive asset characteristics	641
Creating the conditional autoencoder architecture	643
Lessons learned and next steps	648
<b>Summary</b>	<b>648</b>
<b>Chapter 21: Generative Adversarial Networks for Synthetic Time-Series Data</b>	<b>649</b>
<b>Creating synthetic data with GANs</b>	<b>650</b>
Comparing generative and discriminative models	651
Adversarial training – a zero-sum game of trickery	651
The rapid evolution of the GAN architecture zoo	652



GAN applications to images and time-series data	653
<b>How to build a GAN using TensorFlow 2</b>	<b>655</b>
Building the generator network	655
Creating the discriminator network	656
Setting up the adversarial training process	657
Evaluating the results	660
<b>TimeGAN for synthetic financial data</b>	<b>660</b>
Learning to generate data across features and time	661
Implementing TimeGAN using TensorFlow 2	663
Evaluating the quality of synthetic time-series data	672
Lessons learned and next steps	678
<b>Summary</b>	<b>678</b>
<b>Chapter 22: Deep Reinforcement Learning – Building a Trading Agent</b>	<b>679</b>
<b>Elements of a reinforcement learning system</b>	<b>680</b>
The policy – translating states into actions	681
Rewards – learning from actions	681
The value function – optimal choice for the long run	682
With or without a model – look before you leap?	682
<b>How to solve reinforcement learning problems</b>	<b>682</b>
Key challenges in solving RL problems	683
Fundamental approaches to solving RL problems	683
<b>Solving dynamic programming problems</b>	<b>684</b>
Finite Markov decision problems	684
Policy iteration	687
Value iteration	688
Generalized policy iteration	688
Dynamic programming in Python	689
<b>Q-learning – finding an optimal policy on the go</b>	<b>694</b>
Exploration versus exploitation – $\epsilon$ -greedy policy	695
The Q-learning algorithm	695
How to train a Q-learning agent using Python	695
<b>Deep RL for trading with the OpenAI Gym</b>	<b>696</b>
Value function approximation with neural networks	697
The Deep Q-learning algorithm and extensions	697
Introducing the OpenAI Gym	699
How to implement DDQN using TensorFlow 2	700
Creating a simple trading agent	704
How to design a custom OpenAI trading environment	705
Deep Q-learning on the stock market	709
Lessons learned	711
<b>Summary</b>	<b>711</b>
<b>Chapter 23: Conclusions and Next Steps</b>	<b>713</b>
<b>Key takeaways and lessons learned</b>	<b>714</b>



Data is the single most important ingredient	715
Domain expertise – telling the signal from the noise	716
ML is a toolkit for solving problems with data	717
Beware of backtest overfitting	719
How to gain insights from black-box models	719
<b>ML for trading in practice</b>	<b>720</b>
Data management technologies	720
ML tools	722
Online trading platforms	722
<b>Conclusion</b>	<b>723</b>
<b>Appendix: Alpha Factor Library</b>	<b>725</b>
<b>Common alpha factors implemented in TA-Lib</b>	<b>726</b>
A key building block – moving averages	726
Overlap studies – price and volatility trends	729
Momentum indicators	733
Volume and liquidity indicators	741
Volatility indicators	743
Fundamental risk factors	744
<b>WorldQuant's quest for formulaic alphas</b>	<b>745</b>
Cross-sectional and time-series functions	745
Formulaic alpha expressions	747
<b>Bivariate and multivariate factor evaluation</b>	<b>749</b>
Information coefficient and mutual information	749
Feature importance and SHAP values	750
Comparison – the top 25 features for each metric	750
Financial performance – Alphas	752
<b>References</b>	<b>753</b>
<b>Index</b>	<b>769</b>