High-performance Algorithmic Trading using Machine Learning

Building automated trading strategies with AutoML and feature engineering

Franck Bardol

www.bpbonline.com

First Edition 2025

ISBN: 978-93-65893-892

Dedicated to

*My wife **Vassilina***

*My daughters **Ermance** and **Odile***

About the Author

Franck Bardol is an AI professor, senior consultant, and expert advisor in generative AI and machine learning with over two decades of experience in the intersection of quantitative finance, applied research, and data science. He began his career as a quantitative analyst for hedge funds and proprietary trading desks, where he developed algorithmic trading strategies and predictive models across asset classes. He later expanded his work into industrial and service sectors, applying machine learning to problems in fraud detection, maintenance optimization, customer experience, computer vision and detection of toxic emissions from a factory.

Franck has collaborated with organizations such as Axa, CA-CIB, Equalt alternative, Finaltis Hedge fund, Bouygues, Allianz, Orange Telecom Guinée, LVMH, Banque de France, ITER nuclear fusion project, General Electrics and Airbus. He has also contributed to public policy and ethics through his roles as an independent expert for the European Commission and the French AI Villani Commission.

As an educator, he has designed and delivered AI programs for institutions including the University of Geneva, ISEP, ESME Sudria, and Microsoft AI Campus. His teaching covers supervised and unsupervised learning, generative AI, and data-driven strategy, with a focus on real-world application. He is also a LinkedIn Learning instructor in data science and data marketing, having trained over 50,000 learners through his MOOC courses on the platform. He has delivered keynotes at major events such as Swiss IT Forum, Salon du Trading, and the CEPIC Conference, where he spoke alongside leading international press agencies including AFP, Associated Press, and Xinhua.

Franck holds MSc degrees in artificial intelligence, quantitative modeling, and financial markets, along with a certificate in philosophical ethics. He is the founder of the Paris Machine Learning Meetup, a leading European community of over 8,500 AI professionals and experts.

Acknowledgement

I would like to express my sincere gratitude to all those who contributed to the completion of this book.

I am immensely grateful to BPB Publications for their guidance and expertise in bringing this project to life. Their professionalism and support were invaluable in navigating the complexities of the publishing process.

I would also like to extend my thanks to the editors, technical experts, and the entire team of reviewers. Your thoughtful feedback, careful review, and constructive suggestions have greatly contributed to refining the content and elevating the quality of this manuscript.

Finally, I would like to express my deepest appreciation to my beloved parents, my sister Sylvia, and her husband, Charles Bacon, for their unwavering support and encouragement throughout this journey.

And to the readers who have shown interest in this book — your curiosity and engagement are the true reason this work exists.

Thank you to everyone who played a part in making this book a reality.

Preface

Machine learning has rapidly become a transformative tool in algorithmic trading, offering capabilities that go far beyond traditional methods such as econometrics, technical analysis, stochastic calculus, portfolio optimization, and signal processing. While these established approaches have long been staples in quantitative finance, they often rely on rigid assumptions and handcrafted rules. In contrast, machine learning enables systems to learn directly from data—discovering patterns, adapting to market dynamics, and building predictive models with minimal human intervention. The idea that examples—training sets—could be transformed into models automatically is revolutionary.

This book was written to fill a gap I observed repeatedly: the space between the first steps taken by beginners in machine learning for trading, and the more advanced, often inaccessible, expertise found in academic or institutional settings. Many newcomers begin by trying to classify returns—often unsuccessfully—due to a lack of experience with alternative prediction targets or a deeper understanding of feature engineering. My goal is to bridge that gap by introducing modern machine learning techniques that are both powerful and practical. Whether you're working on your own or within a small team, this book focuses on approaches that are computationally efficient, applicable in real trading contexts, and capable of delivering measurable results.

This book is designed as a hands-on journey through the key techniques of machine learning applied to real-world trading. It starts with the foundations of algorithmic strategy design, then progressively expands into supervised learning, unsupervised models, pattern mining, NLP for financial text, and ends with portfolio construction using advanced ML techniques. The focus is entirely practical—mathematical derivations have been intentionally excluded in favor of code, tools, and examples—making the material accessible without sacrificing technical depth.

You will learn how to apply quantamental methods by integrating accounting data into predictive models, detect structural changes in time series and extract rules automatically, work with alternative and unstructured data, and engineer features that go far beyond basic OHLC inputs, filter out market noise while preserving signal, and construct volume- or volatility-based bars and leverage recent breakthroughs in AutoML and low-code ML, using tools like H2O and Microsoft FLAML. Each chapter combines clear explanations, ready-to-run code, and use cases that reflect real trading problems and constraints.

Chapter 1: Algorithmic Trading and Machine Learning in a Nutshell - This chapter introduces systematic trading strategies, key players in the industry, and how machine learning fits into modern trading systems. Covers traditional approaches and contrasts them with ML-driven pipelines.

Chapter 2: Data Feed, Backtests, and Forward Testing - This chapter explores how to acquire macroeconomic and fundamental data via APIs, and how to prepare data for machine learning workflows. Introduces forward testing concepts and time-aware data pipelines.

Chapter 3: Optimizing Trading Systems, Metrics, and Automated Reporting - This chapter covers feature engineering, metric selection, model boosting, and creating automatic performance reports using QuantStats and other tools.

Chapter 4: Implement Trading Strategies - This chapter focuses on event-driven strategy implementation using Backtrader. Includes end-to-end ML strategy deployment, risk management, and performance evaluation.

Chapter 5: Supervised Learning for Trading Systems - This chapter covers the classification and regression algorithms relevant for trading. Emphasizes model selection, metric interpretation, and prediction targets.

Chapter 6: Improving Model Capability with Features - This chapter explores advanced feature creation: technical indicators, entropy, PCA, UMAP, tree-based features, and feature selection techniques.

Chapter 7: Advanced Machine Learning Models for Trading - This chapter presents ensemble methods (boosting, bagging, stacking), kernel-based regressors, and online learning strategies adapted to financial time series.

Chapter 8: AutoML and Low-Code for Trading Strategies - This chapter shows how to use AutoML frameworks (H2O, FLAML) to build efficient models without manual tuning. Focuses on workflow automation and reproducibility.

Chapter 9: Unsupervised Learning Methods for Trading - This chapter introduces change point detection and clustering for uncovering hidden patterns and structural shifts in financial series.

Chapter 10: Unsupervised Learning with Pattern Matching - This chapter teaches how to use recurrence plots, distance matrices, and matrix profiles to identify motifs and anomalies in time series data.

Chapter 11: Trading Signals from Reports and News - This chapter combines NLP and embeddings to extract trading signals from unstructured text. Covers GloVe, UMAP, similarity graphs, and HRP-based portfolio construction.

Chapter 12: Advanced Unsupervised Learning, Anomaly Detection, and Association Rules - This chapter explores unsupervised anomaly detection, projection-based clustering, and association rule mining for discovering hidden market structures.

Code Bundle and Coloured Images

Please follow the link to download the
Code Bundle and the ***Coloured Images*** of the book:

https://rebrand.ly/g36hcbb

The code bundle for the book is also hosted on GitHub at **https://github.com/bpbpublications/High-performance-Algorithmic-Trading-using-Machine-Learning**. In case there's an update to the code, it will be updated on the existing GitHub repository.

We have code bundles from our rich catalogue of books and videos available at **https://github.com/bpbpublications**. Check them out!

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Piracy

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at **business@bpbonline.com** with a link to the material.

If you are interested in becoming an author

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit **www.bpbonline.com**. We have worked with thousands of developers and tech professionals, just like you, to help them share their insights with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions. We at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit **www.bpbonline.com**.

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Table of Contents

1. Algorithmic Trading and Machine Learning in a Nutshell 1
Introduction 1
Structure 1
Objectives 2
Systematic algorithmic trading 2
Emblematic players in systematic trading 3
Implementing the first statistical arbitrage system 5
Discretionary vs. systematic trading 7
Main types of algorithmic strategies 8
Momentum and trend-following strategies 8
Statistical arbitrage 8
Pair trading 8
Major steps to implement mean-reversion strategy 13
Other algorithmic strategies 14
Understanding machine learning 15
Machine learning in trading 17
Momentum strategy with machine learning 17
Meta-strategy using machine learning 21
Meta strategy in action 22
Meta trading strategy 22
Conclusion 25
References 25

2. Data Feed, Backtests, and Forward Testing 27
Introduction 27
Structure 27
Objectives 28
Data feed 28
Macroeconomic and fundamental data 28
World Bank API 29
Federal Reserve Economic Data API 31
Quandl economic data API 32

Fundamental data 33
Tiingo Fundamental data API 34
Financial modeling prep fundamental data API 36
Quantamental investing model 37
Training phase 43
Optimize hyperparameters 46
Open-High-Low-Close price 47
Introduce variable bars 48
Report, studies, and unstructured data 49
Conclusion 50

3. Optimizing Trading Systems, Metrics, and Automated Reporting 51
Introduction 51
Structure 51
Objectives 52
Designing a trading system 52
Preparing data 52
Undertaking feature engineering 53
Feature correlation 53
Detecting and eliminating correlated features 53
Missing values and target column 56
Splitting and scaling 56
Features selection 58
Improving trading systems 60
Boosting models with adaptive parameters search 60
Ensembling modeling 63
Equity curve 65
Selecting metrics for trading systems 67
Ordering criteria for strategy evaluation metrics 67
List of statistics for evaluating a trading strategy 67
Automated trading reports 70
Financial functions for Python 70
Common financial risk metric 73
QuantStat analytical library 73
quantstats.stats 74
Quantstats.plot 74

Quantstats.report ... 75
Conclusion ... 76
References ... 76

4. Implement Trading Strategies ... 77
Introduction ... 77
Structure ... 77
Objectives ... 78
Backtesting and trading libraries ... 78
Criteria for choosing the right library and API ... 79
Implementing a strategy with BackTrader ... 80
Guide to backtesting with BackTrader ... 80
Coding a machine learning strategy with BackTrader ... 81
Import libraries and set-up ... 82
Define strategy in an event-driven environment ... 83
Set-up trading logic ... 84
Create Cerebro engine ... 85
Run backtest ... 86
Risk management ... 88
Adding stop-loss ... 88
Adding take-profit with limit order ... 89
Set broker commission and slippage ... 90
Analyze results ... 91
Trade statistics ... 92
Multi-asset support and advanced features ... 94
Adding multiple data feeds ... 94
Conclusion ... 97
Multiple choice questions ... 97
Answers ... 99
References ... 99

5. Supervised Learning for Trading Systems ... 101
Introduction ... 101
Structure ... 101
Objectives ... 102
Role of supervised learning in trading ... 102

Rise of ML in trading ... *102*
Supervised learning for predictive trading's foundation ... 102
Advantages of supervised learning in trading ... 104
Supervised learning landscape ... 105
Core principles of supervised learning ... 105
Role of labeled data in supervised learning ... *105*
Learning process ... *108*
Main supervised learning algorithms ... 109
Classification algorithms ... *109*
Logistic regression ... *110*
Decision trees ... *110*
Random forest ... *111*
Support Vector Machines ... *112*
k-nearest neighbors ... *112*
Naïve Bayes ... *113*
Gradient Boosting Machines ... *114*
AdaBoost (adaptive boosting) ... *114*
XGBoost ... *115*
LightGBM ... *115*
Regression algorithms ... *116*
Linear regression ... *117*
Ridge regression ... *118*
Lasso regression ... *118*
ElasticNet ... *119*
Decision tree regressor ... *119*
Random forest regressor ... *120*
Gradient boosting regressor ... *121*
K-nearest neighbors regressor ... *122*
Bayesian ridge regression ... *123*
Selection criteria for trading applications ... 124
Classification metrics ... *124*
Confusion matrix ... *124*
Accuracy score ... *126*
Precision score ... *127*
Recall score ... *128*
F1-score ... *130*

Regression metrics ... *130*
Mean Squared Error ... *130*
Root Mean Squared Error ... *131*
Mean Absolute Error ... *132*
Conclusion ... 132

6. Improving Model Capability with Features ... 133
Introduction ... 133
Structure ... 133
Objectives ... 134
Basic features based on OHLC ... 134
Lag values as features ... *134*
Arithmetic-based features ... *135*
Feature discretization ... *137*
Technical features ... 138
Implementing technical indicators with Python ... *139*
Trend detection ... *139*
Momentum indicators ... *140*
Volatility measurement ... *141*
Volume and money flow ... *141*
Volatility and dollar bars-based features ... 142
Memory-preserved features ... 144
Statistical and information-based features ... 147
Hurst exponent ... *147*
Introduction to entropy ... *150*
Using entropy as a feature ... *152*
Machine learning-based features ... 153
PCA-based features ... *153*
UMAP-based features ... *154*
Decision tree leaf based features ... *155*
Feature selection ... 156
Mutual information as a feature selection method ... *157*
Recursive feature elimination ... *158*
Feature selection with PCA and UMAP ... *159*
Combine best features with set operators ... *159*
Conclusion ... 160

7. Advanced Machine Learning Models for Trading 161
Introduction.. 161
Structure.. 161
Objectives 162
Understanding advanced machine learning 162
Introduction to ensembling methods 163
Advanced ensembling method for classification.................................... 163
Statistical foundations of ensembling 164
Fight overfitting with bagging methods... 166
Predict volatility patterns with the bagging algorithm 167
Achieving high accuracy with boosting methods.................................... 171
XG-Boost.. 172
Comparing bagging and boosting.. 175
Voting methods... 175
Advanced regression methods 175
Defeating overfitting 176
Step-by-step guide for advanced regressions.. 176
LASSO method.. 177
Kernel methods 180
Online learning methods.. 182
Implement an online regression with Scikit-multiflow........................... 183
Conclusion.. 184

8. AutoML and Low-Code for Trading Strategies.. 185
Introduction.. 185
Structure.. 185
Objectives 186
Introduction to AutoML and low-code 186
Global macro modelling 187
Macroeconomic features 187
Hand-crafted algorithm as a benchmark 190
LASSO 190
Core components of AutoML 192
Feature engineering and selection.. 192
Feature engineering with Feature-engine 192
Feature engineering and selection with autofeat 193

Algorithm selection *194*
AutoML with FLAML *195*
AutoML with H2O *197*
Hyperparameters optimization *201*
Model evaluation *202*
Results compared to manual approaches 202
Conclusion 202

9. Unsupervised Learning Methods for Trading 205
Introduction 205
Structure 205
Objectives 206
Introduction to change point detection 206
Defining change points *206*
Types of CPD *207*
Methods and algorithms *207*
Challenges, limitations, and applications of CPD 207
Online CPD with ChangeFinder library 208
Offline CPD with ruptures library 210
Introduction to kernel CPD methods *211*
Detecting change points *211*
Comparing kernel CPD to traditional methods *211*
Drawback of kernelized methods *212*
Application of kernel CPD *212*
Assessing the quality of segmentation *216*
Clustering sequences 218
Data preparation for clustering *218*
Applying UMAP algorithm *219*
Clustering sequences with KMeans *219*
Interpreting clustering *221*
Conclusion 223

10. Unsupervised Learning with Pattern Matching 225
Introduction 225
Structure 225
Objectives 226

Understanding unsupervised learning 226
Unsupervised learning techniques 226
Relevance of unsupervised learning to algo trading 227
Importance of unsupervised learning 228
Introduction to recurrence plot and Distance Matrix 228
Understanding recurrence plot 228
Capabilities of an RP 229
Computing a recurrence plot 229
Examples of RPs and DM 230
Linear function 230
Piecewise function 231
Periodic function 231
Periodic linear function 232
Stock price 233
Trading with recurrence plot and Distance Matrix 234
Introduction to Recurrence Quantification Analysis 234
RP as a feature for trading system 236
Baseline based approach 236
Data transformation 237
Labelling 239
Splitting and flattening 239
Train a machine learning algorithm 240
Evaluate the baseline 240
Backtest the baseline algorithm 241
Improve the baseline version 241
Pattern matching 242
Fast pattern matching with matrix profile 243
Find similarities and anomalies with matrix profile 244
Search for pattern with matrix profile 247
Conclusion 251

11. Trading Signals from Reports and News 253
Introduction 253
Structure 253
Objectives 254
Introduction to trading from news and reports 254

Data collection 255

Source of financial news 255

Extracting news data 256

Scrapping BizToc platform 256

Sentiment analysis with Yahoo! RSS feed 257

Reading Wikipedia pages with API 260

Text preprocessing 263

Cleaning 263

Tokenizing 264

Introduction to word embeddings 264

Creating company embeddings with GloVe 266

Visualizing word embeddings with UMAP 267

Building a similarity matrix 268

Building a clustermap 271

Uncovering hidden connections through similarity analysis 273

Building a tree graph 274

Portfolio construction 275

Advantages of allocation based on similarity matrix 275

Maximizing asset diversity through dissimilarity ranking 276

Comparison with equal weight portfolios 278

Hierarchical Risk Parity allocation 278

HRP as an unsupervised algorithm 279

Connecting HRP with clustermap 279

Building the HRP portfolio 279

Weight allocation based on Sharpe metric 282

Conclusion 285

12. Advanced Unsupervised Learning, Anomaly Detection, and Association Rules . 287

Introduction 287

Structure 287

Objectives 288

Introduction to anomaly detection in finance 288

Challenges in detecting financial anomalies 289

Overview of unsupervised approaches to anomaly detection 289

Statistical methods 290

Machine learning approach 294

Sequence clustering and projection for anomaly detection 296
Feature engineering.. *297*
Clustering algorithm.. *298*
Non-linear projection and visualizations... *299*
Anomaly detection for trend reversals *302*
Clustering-based strategy with association rule... 304
Association rules *304*
Data preparation *305*
Search for association rules *307*
Understanding the lift metric in association rule learning *307*
Conclusion... 309
Note to the reader *310*

Appendix: APIs and Libraries for each chapter 311

Index.................................... 313-320

Chapter 1
Algorithmic Trading and Machine Learning in a Nutshell

Introduction

This chapter provides an overview of algorithmic trading. It covers the basics of algorithmic trading strategies.

It explains the reasons why ML is being introduced in trading and the potential consequences of its use. This chapter discusses the use of **machine learning** (**ML**) in algorithmic trading, from momentum to statistical arbitrage strategies. It explores how ML can detect trends and mean-reversion patterns for trading and other innovative applications, such as meta-learning.

Structure

In this chapter, we will cover the following topics:

- Systematic algorithmic trading
- Discretionary vs. systematic trading
- Main types of algorithmic strategies
- Understanding machine learning
- Machine learning in trading
- Meta-strategy using machine learning

Objectives

By the end of this chapter, you will have a robust understanding of algorithmic trading, its inception, the driving forces behind its development, and its diverse applications. Moreover, you can differentiate and describe key algorithmic strategies, from momentum to statistical arbitrage and **high-frequency trading** (**HFT**), recognizing the distinguishing elements and identifying the various participants in the space.

This chapter aims to provide a comprehensive foundation in algorithmic trading and machine learning applications, empowering you to build upon this knowledge in real-world applications.

Systematic algorithmic trading

The evolution of the financial markets and investment industry has led to the development of various sophisticated trading methodologies. One such method that has emerged and seen considerable growth over the years is systematic, algorithmic trading. Algorithmic trading[1] has captured over 50% of the trading volume in US markets today. The reasons for this proliferation are manifold, with the key drivers being the ability to process large amounts of information rapidly and the elimination of human errors and emotions from the trading process. This approach eliminates emotional biases and subjectivity from trading decisions, providing objectivity.

Historically, trading was primarily discretionary, which involved human decision-making and intuition. However, it became apparent over time that this approach has inherent limitations, particularly in processing vast amounts of data and acting rapidly on market opportunities. Systematic algorithmic trading solved these challenges, introducing a new speed, scalability, and efficiency paradigm.

It was introduced in the 1970s when highly computerized trading systems emerged in the American financial markets.

The systematic aspect comes from the use of explicitly formulated investment rules. These rules express the conduct to be followed. Consequently, the writing and formulation of relevant rules becomes a strategic differentiator between investors, and we will see throughout this book how to achieve this. This book aims to explore methods for generating trading rules using self-learning algorithms.

Before going any further, let us take a moment to illustrate this. Here is an example of a trading rule:

```
“Buy Microsoft share if
the Volume exceeds the previous day’s volume and
the closing price is higher the opening”
```

1 https://analyzingalpha.com/algorithmic-trading-history

This rule constitutes a trading strategy, which, when followed, is called systematic or algorithmic trading[2].

The algorithmic trading concept involves applying quantitative models to create, back-test, and implement trading strategies. This approach enables the execution of large orders exceptionally quickly, often resulting in significant financial gains.

Speed and scalability are a natural consequence of using a computer (via programming language) to process these systematic trading rules and send the resulting buy/sell orders to the financial markets. The rule encompasses all the necessary information to make informed investment decisions in each context.

The decision involves selecting the most suitable course of action from the options, such as buy, sell, do nothing, reduce exposure, lighten a portfolio, hedge a financial risk, or protect an investment. The usefulness of a rule is precisely to choose among these actions.

Now let us look at some of the key players in this business and a brief history.

Emblematic players in systematic trading

The narrative of algorithmic trading began in the late 1970s, deeply rooted in quantitative methodologies. *Ed Thorp*[3], a math professor turned hedge fund manager, was one of the pioneers, utilizing his expertise in blackjack strategies to make a lasting impact on Wall Street. His strategies were well-suited for Wall Street, leaving a lasting mark on trading history. He introduced quantitative methods into finance, establishing the foundation for systematic trading. He is considered the first quantitative analyst in history.

Here is how it started[4].

In the late 1970s, the prevailing theory of efficient markets, which posits that financial markets reflect all available information, thus rendering it impossible to consistently achieve higher than average profits, was subject to increasing skepticism. Influential figures like Ed Thorp, renowned for his successful application of probabilistic strategies in blackjack, and *Jerome Baesel*, a distinguished mathematician at *UCI University* and colleague at *Princeton-Newport Partners*, harbored strong beliefs in the existence of market inefficiencies. Their conviction was further buoyed by empirical evidence, including the consistently successful investment strategies of *Warren Buffett*, suggesting that savvy players could indeed beat the market. Thus, the stage was set for the era of systematic trading and the advent of new tools to exploit these inefficiencies.

While at Princeton-Newport Partners, they embarked on a groundbreaking project: studying the impact of various indicators and characteristics on the historical returns of

2 To be quite precise, there is a fine distinction between systematic trading and algorithmic trading. Unlike algorithmic trading, systematic trading offers no discretionary alternative to the trader or manager who applies it. In this book, we will deal mainly with systematic strategies.

3 http:/www.fortunesformula.com/EdwardThorpBio.html

4 Ed Thorp, a mathematician on Wall Street, Statistical Arbitrage, part I, https://www.valuewalk.com/1850840

securities. This audacious endeavor involved analyzing factors like P.E. ratios, book-to-price ratios, and company size, and was met with a wave of criticism from the academic world. Yet, they pressed on undeterred.

Then, in a twist of fate, one of their researchers stumbled upon a game-changing idea: statistical arbitrage.

This concept hinged on a single indicator that ranked stocks from best to worst and offered short-term forecasts of their performance relative to one another. They discovered intriguing recurrent patterns by examining the percentage change in price over a recent period, such as the last two weeks. The stocks that experienced significant gains tended to falter in the subsequent weeks, while those that suffered losses often rebounded.

With this newfound insight, they devised a system called MUD, which cleverly stood for most up, most down stocks. Through extensive computer simulations, they were astounded to find that buying the top-performing decile of stocks while short-selling the bottom-performing decile could yield an annualized return of around 20 percent.

At the end of this paragraph, we will return to this system and propose an implementation.

Concurrently with *Ed Thorp*, innovators such as *Richard Olsen* and *Michael Stumm* launched digital forex trading platforms, further preparing the ground for the adoption of algorithmic methods.

Among the other pioneers of algorithmic trading, the most famous is a mathematician specializing in transmission codes and how to break them. Armed with this knowledge of code breaker, he founded Renaissance in 1982. It is the best-known systematic hedge fund globally for its success[5] and the aura of secrecy surrounding its strategies.

Renaissance Technologies was not alone on this new frontier. Other noteworthy hedge funds, including *D.E. Shaw* and *Citadel*, were also at the forefront of the algorithmic trading movement. They were early adopters of systematic algorithmic trading and have reaped substantial rewards from their endeavors.

For instance, D.E. Shaw manages assets worth over $50 billion, with trading systems powered by algorithms consistently delivering market-beating returns. Similarly, Renaissance Technologies, with around $130 billion in assets under management, and Citadel, with assets exceeding $34 billion, have realized remarkable performance from their algorithmic trading operations.

These entities and their significant successes exemplify the substantial potential inherent in algorithmic trading. However, the nuances and variations in algorithmic trading strategies are vast, with each type possessing unique attributes and considerations. We will delve deeper into these strategies in the subsequent sections.

5 "Renaissance's flagship Medallion fund is famous for the best track record on Wall Street, returning more than 66 percent annualized before fees over a 30-year span from 1988 to 2018".
Source Wikipedia, https://en.wikipedia.org/wiki/Renaissance_Technologies

Let us take a moment to implement the idea, which is probably the ancestor of modern statistical arbitrage systems: buy the poor performers and sell the best. This forms the basis of trading systems based on the **statistical properties** of mean reversion of financial asset prices.

Implementing the first statistical arbitrage system

The sequence is as follows, starting with installing the **Yahoo finance library (yfinance)** if necessary:

```
try :
    import yfinance as yf
except ModuleNotfoundError as e:
    !pip install -q yfinance


import yfinance as yf
```

Next, we request a long history of daily quotes for 30 tickers traded on the New York Stock Exchange. Tickers are randomly picked:

```
# Define the stock tickers
tickers = ['AAP', 'AXP', 'BA', 'CAT', 'CSCO', 'CVX', 'DIS', 'GS', 'HD', \
            'IBM', 'INTC', 'JNJ', 'JPM', 'KO', 'MCD', 'MMM', 'MRK', 'MSFT',
'NKE'\
            'PFE', 'PG', 'TRV', 'UNH', 'VZ', 'WBA', 'WMT', 'XOM', 'MMM']
# Download historical price data from Yahoo Finance
data = yf.download(tickers, start='1990-01-01', end='2023-07-07' , interval
='1d')
```

Compute returns and drop missing values with the help of the Pandas **`dropna()`** function :

```
# Calculate the percentage change in price over a recent period (e.g., last
two weeks)
ret = data['Adj Close'].pct_change(periods=10)
# Drop `Nan` values
ret.dropna(inplace=True)
```

Then, following the logic outlined by Ed Thorp, we start by sorting the returns (**`rank`** function) to determine the best (**`top_decile`**) and worst performers (**`bottom_decile`**) for each period of history.

```
# Rank the stocks based on their percentage change
ranked_ret = ret.rank(axis=1, ascending=False)


# Select the top-performing and bottom-performing deciles of stocks
```