

CASE STUDIES IN A. I. IN FINANCE

(APS1053)

-SYLLABUS-

COURSE DESCRIPTIVE SUMMARY:

This course is built around a large collection of real-world case studies (67), providing hands-on applications of many topics covered in the courses we've taught. Designed as a seminar, it offers a unique, informal, and collegiate learning atmosphere, fostering both collaboration and accountability. The seminar begins with a series of introductory sessions, delivered by the instructors (and potentially guest speakers), to establish the theoretical foundations necessary for understanding the topics in depth. The 67 case studies cover three areas: Reinforcement Learning Applied to Finance: 14 cases; Structured Data: Machine and Deep Learning Applied to Finance: 37 cases; Unstructured Data: Natural Language or Image Processing Applied to Finance: 9 cases, LLM Trading & Asset Management: 7 cases. A selection of 48 of these 67 cases will be assigned to students, depending partly on their expressed interest. After 4 instructor-led sessions, the course transitions into 8 student-led sessions where participants present assigned case studies. These presentations are an opportunity to apply theoretical knowledge to practical problems and to engage in meaningful discussions with peers. Technically, this course explores the application of advanced artificial intelligence techniques through a large collection of solved case studies in areas such as stock trading and hedging strategies, portfolio construction, options modeling, investor profiling, financial news generation, robotic advisors, and more. Topics include the use of cutting-edge open-source libraries for calculating and evaluating financial indicators, developing custom functions for cross-validation and model selection, reformulating problems using reinforcement learning, implementing Python-based reinforcement learning solutions, and analyzing scenarios where such techniques fall short. Specifically, the Finance topics covered in the 67 cases, are: 1) fighting gender/racial/social class bias 2) default loan probability; 3) fraud detection; 4) stock & crypto trading/statistical arbitrage; 5) portfolio dynamic & hierarchical risk parity asset allocation; 6) derivative pricing for dynamic hedging; 7) liquidity modeling for optimal execution; 8) investor risk tolerance analysis with clustering and recommender systems; 9) yield curve prediction for interest rate modeling; 10) document processing for sentiment and news summarization; 11) satellite image processing to predict economic activity; 12) volatility prediction for market risk modeling (VAR, ES) and corporate governance risk modeling; 13) synthetic data generation; 14) Big Data processing (time series), 15) LLM Trading & Asset Management.

COURSE PREREQUISITES:

No specific prerequisites are required, apart from very basic programming skills, some familiarity with financial terms and a strong interest in the practical applications of AI in solving real-world financial problems.

COURSE STRUCTURE AND CONTENT:

The initial 4 sessions of the seminar will be Instructor-Led Sessions in which the theoretical foundations will be delivered by the lecturers. More specifically, Session 1 will be used for planning and case-assignment. In Session 2 & 3, the issue of Model Selection Bias --why it is important, how to address it, and why it goes

beyond the issue of model tuning, and the well-known tradeoff between Bias and Variance--, will be covered in depth. Regarding Session 4, as 14 of the cases consist of applications of reinforcement learning to finance, the instructors shall spend these covering the theoretical foundations of Reinforcement Learning. After the 4th session, all the remaining ones will be Student-Led Sessions: the subsequent 8 sessions will feature student presentations of assigned case studies. (Approx. 6 presentations of 30 min. each, during each one of the 8 remaining 3 hr. Sessions). In these presentations, students will delve into the theoretical frameworks, financial principles, AI techniques, coding implementations, and opportunities for improvement or extension of each topic. Students will be evaluated based on their assigned case study-related work. The 67 case studies offered in this course cover three important areas of applications of A. I. to Finance; these are: (i) Cases on Reinforcement Learning Applied to Finance (14 cases: 1-9, 23-25, 43, 60) (ii) Cases on Unstructured Data: Natural Language or Image Processing Applied to Finance (10 cases: 26-28, 40, 42, 52-54, 56-57) (iii) Cases on Structured Data: Machine and Deep Learning Applied to Finance (36 cases, i. e., the rest of the cases), LLM Trading & Asset Management (7 cases). As mentioned before, a selection (48) of these 67 cases will be assigned to students, depending on their interest and background.

COURSE TECHNICAL LEARNING OBJECTIVES & OUTCOMES:

Understand the main families of AI methods used in finance.

Students are expected to recognize the main methodological categories and understand their basic logic and use cases.

Connect model type to problem type.

Students are expected to learn which classes of models are appropriate for which financial tasks.

Run and reproduce applied AI workflows in Python.

Students must work with Jupyter notebooks, rerun cases using fresh data, and present the associated code. That implies practical competence in executing and understanding applied workflows, not merely discussing them in theory.

Modify existing code and perform small extensions.

Students must extend the notebook code by one experiment. This points to an implicit objective of basic research-style or development-style competence: taking an existing implementation and pushing it one step further.

Evaluate models critically, especially with respect to bias and overfitting.

The prominence given to model selection bias in the lecture sequence suggests that a core technical objective is to avoid naive model evaluation and to understand the dangers of misleading performance claims in finance.

Understand the theoretical foundations behind the case studies.

The student presentations are explicitly supposed to explain theoretical frameworks and coding implementations, while the first sessions establish conceptual foundations. So students are implicitly expected to know enough theory to explain why a method works, not just how to execute it.

MARKING SCHEME:

Grades will be based on 4 assignments. One AI-Agent homework worth 10% of the final grade and three Case homework each worth 30% of the final grade.

DELIVERABLES:

Homework on AI-Agent: A mark down compendium knowledge base with associated evaluation metrics. The homework consists of executing all the steps in the workflow to create an AI-Agent curated and epistemologically structured knowledge base or compendium (not a summary) of an investment channel consisting of thousands of video transcripts. The steps include the writing of declarative prompts for an LLM and the execution of the corresponding Python programs, including compendium evaluation programs, all of them AI-generated and a number of them AI-executed as well. Students are expected to sign up for and buy DeepSeek API tokens to do this homework. (about CAD \$15).

Case Homework: Students are expected to prepare 2 thirty minute long presentations and 1 written case 2000 word report that summarizes and criticizes the case. Depending on the number of students, this may be

changed to 3 thirty minute long presentations and NO written case 2000 word report. The presentations shall be documented with: an (at least) 30 slide power-point slide deck, an accompanying script for the power-point slide deck, a Jupiter notebook with its associated data (students must run the case notebook with fresh data, possibly a subset of the original data), and any code extending the notebook (students are expected to extend the case notebook code by one small experiment). The student must be prepared to answer questions about both the theory and the case code.

REFERENCES:

Financial Theory Books:

- Nyholm, Ken: Strategic Asset Allocation in Fixed-Income Markets, 2008.
- Benninga, Simon: Financial Modeling, 2008.
- Aronson, David: Evidence Based Technical Analysis, 2006.

Prompt Engineering Books:

- Clocksin and Mellish, Programming in Prolog, 1981.
- El Amri, AYmen, LLM Prompt Engineering for Developers, 2023.

Source Books for the 60 Finance Case Studies (all downloadable from U of T library):

- Tatsat, Hariom: Machine Learning and Data science Blueprints for Finance, 2021.
- Hilpisch, Yves: Reinforcement Learning for Finance, 2025.
- Karasan, Abdullah: Machine Learning for Financial Risk Management with Python, 2022.
- Patel, Ankur: Hands-On Unsupervised Learning Using Python, 2019.
- Kaabar, Sofien: Deep Learning for Finance, by Kaabar, 2024.
- Klaas, Jannes: Machine Learning for Finance, 2019.
- Jansen, Stephen: Machine learning for algorithmic trading, 2020
- Jesse, Hamlet: Generative AI for Trading and Asset Management, 2025
- Github (see below).

--COURSE LAYOUT--

SESSIONS 1-4. PLANNING MEETING + INSTRUCTOR'S LECTURES.

Session 1

- Planning: case-studies will be selected and assigned.

Session 2

- Introduction to a stock trading system
- Parts of a trading system:
- What is an indicator?

- What is a trading algorithm or trading rule?
- What is a trading system output?
- What is a financial metric?
- How is a trading system evaluated?
- Model Selection Bias: financial metrics are not enough.
- Why model selection bias important for a trading system, how to address it.
- Why the Model Selection Bias goes beyond the issue of model tuning and the well-known tradeoff between Bias and Variance.

Session 3

- Reinforcement Learning:
 - Deep Q Learning,
 - Policy Gradient

Session 4

- What is a Large Language Model (LLM)
- What is an “AI agent”
- Types of “AI agents”
 - Pipelines
 - Loops
- How to program an “AI agent” with a LLM as programmer
 - Declarative vs Procedural Prompts
 - Making a prompt more deterministic
 - Programming iteration loop using a metric
- Application/Case: Building a curated investment knowledge base using an “AI agent”

SESSIONS 5-12: 48 CASE PRESENTATIONS, 6 CASES PER SESSION

Session 5

6 Cases on Reinforcement Learning: stock trading, derivatives pricing and trading, asset allocation

Session 6

6 Cases on Reinforcement Learning: stock trading, derivatives pricing and trading, asset allocation

Session 7

2 Cases on Reinforcement Learning: stock trading, derivatives pricing and trading, asset allocation

4 Cases on Unstructured Data: document and image processing to obtain signals of economic activity, market sentiment

Session 8

6 Cases on Unstructured Data: document and image processing to obtain signals of economic activity, market sentiment; RAG extension of the case on building a curated investment knowledge base using an “AI agent”

Sessions 9, 10, 11, 12

24 Cases on Structured Data: stock trading, derivatives pricing and trading, asset allocation, liquidity, volatility, interest rate modelling, synthetic data generation, risk management, fraud detection, credit allocation, costumer/investor analysis and management

LIST OF CASES TO SELECT FOR SEMINAR PRESENTATION

BOOK 1: Reinforcement Learning for Finance by Hilpisch: UofT:

https://librarysearch.library.utoronto.ca/view/action/uresolver.do?operation=resolveService&package_service_id=46202946440006196&institutionId=6196&customerId=6195&VE=true

THE BASICS:

Case Study 1: Chapter 1: Learning through interaction [p.3]: The first chapter focuses on learning through interaction with four major examples: probability matching, Bayesian updating, RL, and DQL.

Case Study 2: Chapter 2: Deep Q Learning [p.19]: The second chapter introduces concepts from dynamic programming (DP) and discusses DQL as an approach to approximate solutions to DP problems. The major theme is the derivation of optimal policies to maximize a given objective function through taking a sequence of actions and updating the optimal policy iteratively. DQL is illustrated on the basis of a DQL agent that learns to play the CartPole game from the Gymnasium Python package.

Case Study 3: Chapter 3: Financial Q Learning [p.37]: The third chapter develops a first Finance environment that allows the DQL agent from Chapter 2 to learn a financial prediction game. Although the environment formally replicates the API of the CartPole game, it misses some important characteristics that are needed to apply RL successfully.

DATA AUGMENTATION:

Case Study 4: Chapter 4: Simulated Data [p.51]: The fourth chapter is about data augmentation based on Monte Carlo simulation (MCS) approaches, and it discusses the addition of noise to historical data and the simulation of stochastic processes.

Case Study 5: Chapter 5: Generated Data [p.67]: The fifth chapter introduces generative adversarial networks (GANs) to synthetically generate time series data that has statistical characteristics that are similar to those of historical time series data on which a GAN was trained.

FINANCIAL APPLICATIONS:

Case Study 6: Chapter 6: Algorithmic Trading [p.85]: Building on the example from Chapter 3, this chapter applies DQL to the problem of algorithmic trading based on the prediction of the next price movement's direction.

Case Study 7: Chapter 7: Dynamic Hedging [p.105]: The seventh chapter is about learning optimal dynamic hedging strategies for an option with European exercise in the Black-Scholes-Merton (1973)

model. In other words, delta hedging or dynamic replication of the option is the goal.

Case Study 8: Chapter 8: Dynamic Asset Allocation [p.129]: This chapter applies DQL to three canonical examples in asset management: one risky asset and one risk-free asset, two risky assets, and three risky assets. The problem is to dynamically allocate funds to the available assets to maximize a profit target or a risk-adjusted return (Sharpe ratio).

Case Study 9: Chapter 9: Optimal Execution [p.167]: The ninth chapter is about the optimal liquidation of a large position in a stock. Given a certain risk aversion, the total execution costs are to be minimized. This use case differs from the others in that all actions are tightly connected with each other through an additional constraint. The chapter also introduces an additional RL algorithm in the form of an actor-critic implementation.

BOOK 2: Machine Learning Blueprints for Finance by Tatsat UofT:

https://librarysearch.library.utoronto.ca/permalink/01UTORONTO_INST/14bjeso/alma991107061413606196

Asterisk: (*) Beginner level (Do not select if you took APS1052)

*Case Study 10: Stock Price Prediction [p.95]:**
Blueprint for Using Supervised Learning Models to Predict a Stock Price

Case Study 11: Derivative Pricing [p.114]:
Blueprint for Developing a Machine Learning Model for Derivative Pricing (See Part III of Benninga in Nyholm minicourse).

Case Study 12: Investor Risk Tolerance and Robo-Advisors [p.125]:
Blueprint for Modeling Investor Risk Tolerance and Enabling a Machine Learning-Based Robo-Advisor

*Case Study 13: Yield Curve Prediction [p.141]:**
Blueprint for Using Supervised Learning Models to Predict the Yield Curve (See Part IV of Benninga in Nyholm minicourse)

*Case Study 14: Fraud Detection [p.153]:**
Blueprint for Using Classification Models to Determine Whether a Transaction Is Fraudulent

*Case Study 15: Loan Default Probability [p.166]:**
Blueprint for Creating a Machine Learning Model for Predicting Loan Default Probability

*Case Study 16: Bitcoin Trading Strategy [p.179] **
Blueprint for Using Classification-Based Models to Predict Whether to Buy or Sell in the Bitcoin Market

Case Study 17: Portfolio Management: Finding an Eigen Portfolio [p.202]:
Blueprint for Using Dimensionality Reduction for Asset Allocation (See Part II of Benninga in Nyholm minicourse)

Case Study 18: Yield Curve Construction and Interest Rate Modeling [p.217]:
Blueprint for Using Dimensionality Reduction to Generate a Yield Curve (See Part IV of Benninga in Nyholm minicourse)

*Case Study 19: Bitcoin Trading: Enhancing Speed and Accuracy [p.227]:**
Blueprint for Using Dimensionality Reduction to Enhance a Trading Strategy

Case Study 20: Clustering for Pairs Trading [p.243]:
Blueprint for Using Clustering to Select Pairs

*Case Study 21: Portfolio Management: Clustering Investors [p.259]:
Blueprint for Using Clustering for Grouping Investors*

*Case Study 22: Hierarchical Risk Parity [p.267]:
Blueprint for Using Clustering to Implement Hierarchical Risk Parity (See Part II of Benninga in Nyholm minicourse).*

*Case Study 23: Reinforcement Learning–Based Trading Strategy [p.298]:
Blueprint for Creating a Reinforcement Learning–Based Trading Strategy*

*Case Study 24: Derivatives Hedging [p.316]:
Blueprint for Implementing a Reinforcement Learning–Based Hedging Strategy (See Part III of Benninga in Nyholm minicourse).*

*Case Study 25: Portfolio Allocation [p.334]
Blueprint for Implementing a Reinforcement Learning–Based Portfolio Allocation (See Part II of Benninga in Nyholm minicourse)*

*Case Study 26: NLP and Sentiment Analysis–Based Trading Strategies [p.362]:
Blueprint for Building a Trading Strategy Based on Sentiment Analysis*

*Case Study 27: Chatbot Digital Assistant [p.383]:
Blueprint for Creating a Custom Chatbot Using NLP*

*Case Study 28: Document Summarization
Blueprint for Using NLP for Document Summarization [p.394]:*

BOOK 3: Machine Learning for Financial Risk Management with Python by Karasan UofT:
https://librarysearch.library.utoronto.ca/view/action/uresolver.do?operation=resolveService&package_service_id=46202946440006196&institutionId=6196&customerId=6195&VE=true

Case Study 29: Chapter 3 [p.57]: Deep Learning for Time Series Modeling: introduces the deep learning tools for time series modeling. Recurrent Neural Network and Long Short Term Memory are two approaches by which we are able to model the data with time dimension. Besides, this chapter gives us an impression about the applicability of deep learning models to time series modeling.*

*Case Study 30: Chapter 4 [p.75]:*Machine Learning for Volatility Prediction: Increased integration of financial markets has led to a prolonged uncertainty in financial markets, which in turn stresses the importance of volatility. Volatility is used in measuring the degree of risk, which is one of the main engagements of the area of finance. The fourth chapter deals with the novel volatility modeling based on Support Vector Regression, Neural Network, Deep Learning, and Bayesian approach. For the sake of comparison of the performances, traditional ARCH and GARCH-type models are also employed.*

Case Study 31: Chapter 5 [p.119]: Machine learning for Modeling Market Risk : employs machine learning-based models to boost estimation performance of the traditional market risk models, namely Value-at-Risk (VaR) and Expected Shortfall (ES). VaR is a quantitative approach for the potential loss of fair value due to market movements that will not be exceeded in a defined period of time and with a defined confidence level. Expected Shortfall, on the other hand, focuses on the tail of the distribution referring to big and unexpected losses. VaR model is developed using denoised covariance matrix and ES is developed incorporating liquidity dimension of the data.

Case Study 32: Chapter 7 [p.193]: Gaussian Mixture for Liquidity Modeling: is used to model the liquidity, which is thought to be a neglected dimension in risk management. This model allows us to incorporate different aspects of the liquidity proxies in this chapter so that we will be able to capture the effect of liquidity on financial risk in a more robust way.

Case Study 33. Chapter 9 [p.255]: *Machine learning for A Corporate Governance Risk Measure Stock Market Crash*: introduces a brand new approach in modeling the corporate governance risk: Stock Price Crash. Many studies find an empirical link between stock price crash and corporate governance. This chapter, using Minimum Covariance Determinant model, attempts to unveil the relationship between the components of corporate governance risk and stock price crash.

Case Study 34. Chapter 10 [p.281] *Machine learning for Synthetic Data Generation and the Hidden Market Model in Finance*: makes use of synthetic data to estimate different financial risks. The aim of this chapter is to highlight the emergence of synthetic data that helps us to minimize the impact of limited historical data. So, synthetic data allows us to have large enough and high-quality data, which improves the quality of the model.

BOOK 4: Hands-on Unsupervised Learning Using Python by Patel:

UofT:https://librarysearch.library.utoronto.ca/view/action/uresolver.do?operation=resolveService&package_service_id=46202946440006196&institutionId=6196&customerId=6195&VE=true

Case Study 35: Chapter 4 [p. 97]: *Anomaly Detection: Credit Card Fraud Detection with PCA et.al.*

Case Study 36: Chapter 8: *Hands-On Autoencoder* [p. 179]: *Anomaly Detection: Credit Card Fraud Detection with Auto-Encoder*

Case Study 37: Chapter 10: *Recommender Systems using Restricted Boltzmann Machines* [p. 231]. Focus on Collaborative Filtering.

BOOK 5: Deep Learning for Finance by Kaabar: UofT:

UofT:https://librarysearch.library.utoronto.ca/view/action/uresolver.do?operation=resolveService&package_service_id=46202946440006196&institutionId=6196&customerId=6195&VE=true

Case Study 38: Chapter 9 [p. 249]: *Deep Learning for Times Series Prediction II: Fractional Differentiation*. Should also present, on the same topic: *Advances on Financial Machine Learning* by Marcos Lopez de Prado, Chapter 5: *Fractionally Differentiated Features* , UofT: https://librarysearch.library.utoronto.ca/view/action/uresolver.do?operation=resolveService&package_service_id=46202946440006196&institutionId=6196&customerId=6195&VE=true

Case Study 39: Chapter 11 [p.277]: *Advanced Techniques and Strategies: Using COT Data to Predict Long-Term Trends, Algorithms 1, 2, 3, Putting it All Together.*

BOOK 6: Machine Learning for Finance by Klaas: UofT:

UofT:https://librarysearch.library.utoronto.ca/view/action/uresolver.do?operation=resolveService&package_service_id=46202946440006196&institutionId=6196&customerId=6195&VE=true

Case 40 Chapter 3 [p.71] – Chapter 3 – Utilizing Computer Vision

Explains how convolutional neural networks (CNNs) process images, from basic filters to advanced architectures. The chapter demonstrates training image classifiers, using pretrained models, data augmentation, and extends beyond classification to tasks such as facial recognition and object detection

Case 41 Chapter 4 [p.115]– Understanding Time Series*

Covers the analysis and modeling of temporal data, a core financial use case. It discusses stationarity, autocorrelation, ARIMA models, Kalman filters, and neural approaches (CNNs, RNNs, LSTMs), showing how classical and modern methods complement each other in forecasting. Time series, Big Data.

Case 42 Chapter 5 [p.167] – Parsing Textual Data with Natural Language Processing

Introduces NLP techniques for financial text using spaCy and Keras. Topics include named entity recognition, rule-based systems, text classification, word embeddings, attention mechanisms, and sequence-to-sequence models, with applications such as news analysis and language translation.

Case 43 Chapter 6 [p.227] – Using Generative Models

Explores models that generate data, including autoencoders, variational autoencoders, and GANs. The chapter shows how these models can be used for anomaly detection, fraud detection, data augmentation, and active learning to reduce labeling costs.

Case 44 Chapter 7 [p.277] – Reinforcement Learning for Financial Markets

Presents reinforcement learning concepts and algorithms (Q-learning, actor-critic methods) and connects them to economic theory. Practical examples illustrate how RL can be applied to trading, portfolio construction, and sequential decision-making under uncertainty.

Case 45 Chapter 9 [p.365]– Fighting Bias

Examines how bias and unfairness arise in machine learning systems, especially in regulated financial contexts. It introduces legal considerations, observational and causal fairness, interpretability, and provides a practical checklist for developing and monitoring fair models. In more detail: you learn about fairness in machine learning in different aspects. First, we discuss legal definitions of fairness and quantitative ways to measure these definitions. We then discuss technical methods to train models to meet fairness criteria. We also discuss causal models. We learn about SHAP as a powerful tool to interpret models and find unfairness in a model. Finally, we learn how fairness is a complex systems issue and how lessons from complex systems management can be applied to make models fair.

Case 46 Chapter 10 [p.401] – Bayesian Inference and Probabilistic Programming

Introduces Bayesian reasoning and probabilistic programming using PyMC3. The chapter explains priors, posteriors, MCMC methods, and demonstrates how probabilistic models can be used to quantify uncertainty, such as inferring volatility from financial data.

BOOK 7: Machine learning for algorithmic trading : predictive models to extract signals from market and alternative data for systematic trading strategies with Python, 2nd edition

By Stephen Jansen UofT:

https://librarysearch.library.utoronto.ca/nde/fulldisplay?query=Machine%20Learning%20for%20Algorithmic%20Trading%20Jansen&tab=Everything&search_scope=UTL_AND_CI&searchInFulltext=false&vid=01UTORONTO_INST:UTORONTO_NDE&lang=en&docid=alma991107212149706196&adaptor=Local%20Search%20Engine&context=L&isFrbr=false&isHighlightedRecord=false&state=

Case 47 Chapter 9 [p.255], Time-Series Models for Volatility Forecasts and Statistical Arbitrage, covers univariate and multivariate time series diagnostics and models, including vector autoregressive models as well as ARCH/GARCH models for volatility forecasts. It also introduces cointegration and shows how to use it for a pairs trading strategy using a diverse set of exchange-traded funds (ETFs).

Case 48 Chapter 10 [p.295], Bayesian ML – Dynamic Sharpe Ratios and Pairs Trading, presents probabilistic models and how Markov chain Monte Carlo (MCMC) sampling and variational Bayes facilitate approximate inference. It also illustrates how to use PyMC3 for probabilistic programming to gain deeper insights into parameter and model uncertainty, for example, when evaluating portfolio performance.

Case 49 Chapter 11, Random Forests – A Long-Short Strategy for Japanese Stocks [p.327], shows how to build, train, and tune nonlinear tree-based models for insight and prediction. It introduces tree-based ensembles and shows how random forests use bootstrap aggregation to overcome some of the weaknesses of decision trees. We then proceed to develop and backtest a long-short strategy for Japanese equities.*

Case 50 Chapter 12, Boosting Your Trading Strategy [p.365], introduces gradient boosting and demonstrates how to use the libraries XGBoost, LightBGM, and CatBoost for high-performance training and prediction. It reviews how to tune the numerous hyperparameters and interpret the model using SHapley Additive exPlanation (SHAP) values before building and evaluating a strategy that trades US

equities based on LightGBM return forecasts.

Case 51 Chapter 13 [p.407], *Data-Driven Risk Factors and Asset Allocation with Unsupervised Learning*, shows how to use dimensionality reduction and clustering for algorithmic trading. It uses principal and independent component analysis to extract data-driven risk factors and generate eigenportfolios. It presents several clustering techniques and demonstrates the use of hierarchical clustering for asset allocation.

Case 52 Chapter 14 [p.439], *Text Data for Trading – Sentiment Analysis*, demonstrates how to convert text data into a numerical format and applies the classification algorithms from Part 2 for sentiment analysis to large datasets.

Case 53 Chapter 15, *Topic Modeling– Summarizing Financial News* [p.463], uses unsupervised learning to extract topics that summarize a large number of documents and offer more effective ways to explore text data or use topics as features for a classification model. It demonstrates how to apply this technique to earnings call transcripts sourced in Chapter 3 and to annual reports filed with the Securities and Exchange Commission (SEC).

Case 54 Chapter 16, *Word Embeddings for Earnings Calls and SEC Filings* [p.483], uses neural networks to learn state-of-the-art language features in the form of word vectors that capture semantic context much better than traditional text features and represent a very promising avenue for extracting trading signals from text data.

Case 55 Chapter 17, *Deep Learning for Trading* [p.513]*, introduces TensorFlow 2 and PyTorch, the most popular deep learning frameworks, which we will use throughout Part 4. It presents techniques for training and tuning, including regularization. It also builds and evaluates a trading strategy for US equities.

Case 56 Chapter 18, *CNNs for Financial Time Series and Satellite Images* [p.551], covers convolutional neural networks (CNNs) that are very powerful for classification tasks with unstructured data at scale. We will introduce successful architectural designs, train a CNN on satellite data (for example, to predict economic activity), and use transfer learning to speed up training. We'll also replicate a recent idea to convert financial time series into a two-dimensional image format to leverage the built-in assumptions of CNNs.

Case 57 Chapter 19, *RNNs for Multivariate Time Series and Sentiment Analysis* [p.591], shows how recurrent neural networks (RNNs) are useful for sequence-to-sequence modeling, including for univariate and multivariate time series to predict. It demonstrates how RNNs capture nonlinear patterns over longer periods using word embeddings introduced in Chapter 16 to predict returns based on the sentiment expressed in SEC filings.

Case 58 Chapter 20, *Autoencoders for Conditional Risk Factors and Asset Pricing* [p.625], covers autoencoders for the nonlinear compression of high- dimensional data. It implements a recent paper that uses a deep autoencoder to learn both risk factor returns and factor loadings from the data while conditioning the latter on asset characteristics. We'll create a large US equity dataset with metadata and generate predictive signals.

Case 59 Chapter 21, *Generative Adversarial Networks for Synthetic Time-Series Data* [p.649], presents one of the most exciting advances in deep learning. Generative adversarial networks (GANs) are capable of learning to reproduce synthetic replicas of a target data type, such as images of celebrities. In addition to images, GANs have also been applied to time-series data. This chapter replicates a novel approach to generate synthetic stock price data that could be used to train an ML model or backtest a strategy, and also evaluate its quality.

Case 60 Chapter 22, *Deep Reinforcement Learning – Building a Trading Agent* [p.679], presents how

reinforcement learning (RL) permits the design and training of agents that learn to optimize decisions over time in response to their environment. You will see how to create a custom trading environment and build an agent that responds to market signals using OpenAI Gym.

BOOK 8: Generative AI for Trading and Asset Management, 2025

Case 61 Ch. 5 — Deep Autoregressive Models for Sequence Modeling

Logistic regression/FVSN, MADE, causal masked networks/WaveNet, RNNs, Transformers (attention, positional encoding, multi-head attention, encoder), time-series transformers (Chronos, Lag-Llama)

Case 62 Ch. 6 — Deep Latent Variable Models

Factor analysis, probabilistic PCA, Gaussian mixture models (regime detection), deep latent variable models, variational inference, VAEs, TimeVAE

Case 63 Ch. 7 — Flow Models

Linear/nonlinear/coupling flows (NICE, Real-NVP), autoregressive flows, continuous normalizing flows, modeling financial time series

Case 64 Ch. 8 — Generative Adversarial Networks

GAN training, theory, Wasserstein GAN/WGAN-GP, extending GANs for time series

Case 65 Ch. 9 — Leveraging LLMs for Sentiment Analysis in Trading

Fed press-conference sentiment analysis; Whisper speech-to-text; audio segmentation; BERT/FinBERT fine-tuning; experiment results

Case 65 Ch. 10 — Efficient Inference

Scaling LLMs; knowledge distillation; model quantization; fine-tuning (traditional FT, PEFT: BitFit, Adapters, Prompt-tuning, LoRA/QLoRA); aligning with human preferences

Case 67

GITHUB:

"StockTime: A Time Series Specialized Large Language Model Architecture for Stock Price Prediction" (Wang et al., 2024) <https://arxiv.org/abs/2409.08281>

QuantAgent multi-agent framework (Wang et al., 2024 — <https://arxiv.org/abs/2509.09995>)

INSTRUCTORS:

Sabatino Costanzo-Alvarez:

Holds a Masters in Economics and Finance from Brandeis University as well as a Magister Scientiarum, a Magister Philosopharum and a Ph.D. in Mathematics from Yale University, where in 1990 achieved a significant breakthrough by solving an important mathematical conjecture which had remained unsolved for more than 3 decades. Taught Mathematics of Finance at Boston University as an Associated Professor for 5 years and later co-founded the Boston Trading Group LLC, designed the trading systems used in the

firm's daily Futures Trading Operations and acted as head trader of the team. Holds the licenses "Registered Representative NYSE/NASDAQ" (Series 7), "Registered Financial Advisor", "Registered Uniform State Law Securities Agent", "Registered Managed Futures Fund Representative" in the U.S. and "Canadian Securities Course" & "Conduct and Practices" in Canada, as well as products training at Morgan Stanley in Boston, and later at Merrill Lynch in New York. Chaired the Advanced Management Program for Senior Executives (PAG), an Executive MBA at the IESA Institute, where he taught Financial Engineering and Investment Management as an Associate Professor, and tutored over 70 MBA dissertations. Acted as Head of Research at Econo Invest C.A., one of the largest Investment Firms in Latin America, leading the Investment Strategy Team in charge of generating and executing the U.S. & E.U. investment strategies for Commodities, Fixed Income Instruments and Equities for the firm (published weekly in Bloomberg), as well as generating and maintaining the Sovereign Fixed Income Indexes of Brazil, Colombia, Mexico, Peru, Chile, Uruguay and Venezuela to be used in the design of international financial products. Acted as an Investment Advisor for the International Wealth Management Groups at Morgan Stanley (Boston), Merrill Lynch (NY) and the Royal Bank of Canada(Toronto), and is now a Senior Partner at the Toronto boutique Investment Firm Inter Alea, where he provides state-of-the-art mathematical modeling solutions to portfolio and risk management problems for a select group of corporate and high net worth private clients, designing and managing their investment portfolios based on their specific risk & return requirements. He taught Portfolio Management, Statistics & Mathematical Modelling and Business Mathematics Courses at the Pilon School of Business, and is the founder and advisor of the Sheridan Students Trading and Investment Association. He is a Lecturer at the U of T Graduate School, where he is teaching Portfolio Management, Blockchain Technology, Cryptocurrencies and Artificial Intelligence applied to Finance.

Rosario Lorenza Trigo-Ferre:

Holder of a B. A. in Philosophy (Magna Cum Laude) from Yale University -where she also received training in Math & Physics-, a Ph.D. in Generative Linguistics from Massachusetts Institute of Technology (MIT) and a M. Sc. in Management of Information Systems from Boston University ("Beta Gamma Sigma Honors" award), she was a Professor at Boston University for 8 years. While a Programmer Analyst at Boston University, she designed and developed an application for the management of accounts trading stock and currency futures and co-designed financial applications under the direction of Professor Zvie Bodie at B.U. Co-founder and Trader at the Boston Trading Group and Certified Programmer Analyst in e-commerce by the University Computer Careers Program, she generated the trading signals for currencies and metals futures used in the BTG's market operations; developed an application maximizing the efficiency of trading system for currency and metal futures, and designed a client-server application for the management and operation of trading accounts. Has designed and developed many multi- tiered e-commerce applications dynamically generated from databases. Project leader and senior programmer analyst at IngeDigit, designed and developed internet applications for banking accounts management & operation, and for international transactions between banking accounts and credit cards. She was a Professor at the Department of Production and Technical Innovation of the IESA Institute, the top -only US accredited- Venezuelan Business School, where taught courses in Information Systems, Simulation in Finance, Operations and Database Marketing. She is the author of many scientific papers in refereed journals and a Permanent Consultant for an international development bank (C.A.F, The Andean Region Development Bank), where she has designed the financial models used to evaluate the profitability, coverage and socio-economic impact of projects like the inclusion of fiber-optic cable in highways in Colombia and Peru. These models led to the enactment of new laws making such inclusion mandatory in the Andean region. Also designed the financial models used to evaluate the profitability of projects in satellite technology in Argentina (specifically the ARSAT program) by estimating the future regional demand for transponders and the impact of the project in the input-output matrix of the country, and is now a Partner at the boutique Investment Firm InterAlea, where she designs, develops, tests and implements trading and risk management strategies based on the entropy analysis of price signals, executed on stock quote-data processed through SQL-Server. She is a Lecturer at the U of T Graduate School, where she is teaching Portfolio Management, Blockchain Technology, Cryptocurrencies and Artificial Intelligence applied to Finance.