New York University
Tandon School of Engineering
Finance and Risk Engineering Department
1 Metrotech Center, Suite 1001
Brooklyn, NY 11201

Machine Learning in Financial Engineering

COURSE FRE-GY 7773, FALL 2025

Professor Andrey Itkin

Email: aitkin@nyu.edu

Office hours: By appointment (happy to meet, discuss, support, zoom). Email preferred.

COURSE OVERVIEW

This course provides an in-depth exploration of machine learning applications in market forecasting and modeling for various asset classes. We begin with foundational techniques such as regressions and "shallow" machine learning models (e.g., Support Vector Machines, Random Forests) before advancing to "deep" layered models (e.g., Long Short-Term Memory Networks).

For each model, we examine:

- *Input variables and architecture:* The rationale behind their design.
- *Learning evaluation*: How model performance is tracked during training.
- *Performance assessment:* The metrics used by machine learning scientists and market traders to gauge effectiveness.

This course is hands-on providing a comprehensive foundation in modern ML through practical code examples:

- *Modern Python & OOP:* All code is built with contemporary Python and Object-Oriented Programming for scalable, maintainable software.
- *Production-Ready Practices:* Learn to structure and manage projects using packages that mimic a real-world production environment.
- *Data Manipulation Mastery:* Work with real datasets using both Polars and Pandas for efficient data analysis and preparation.
- *Full ML Stack Coverage:* Implement models with the industry's leading tools, including Scikit-learn, TensorFlow, and PyTorch.
- *Software Engineering for ML:* Go beyond algorithms to master the design patterns, proven tricks, and practical recipes for building effective ML systems.

By the end of the course, students will be able to:

- Identify key characteristics of machine learning models for market forecasting.
- Assess whether a model is likely to be useful and efficiently structured in terms of inputs and outputs.

Key Topics Covered

- Training and testing workflow: Scaling, cross-validation pipelines.
- Optimization techniques: Stochastic and mini-batch gradient descent.
- Financial metrics: Profitability and risk analysis.
- Feature engineering: Techniques tailored for financial data.
- Models:
    a. Traditional & shallow learning: Multivariate regression, logistic regression, SVMs, PCA, decision trees, random forests.
    b. Clustering & probabilistic models: k-means, hierarchical clustering, Gaussian mixtures.
    c. Deep learning: MLPs, LSTMs, auto-encoder neural networks.
    d. Applications:
    e. Financial time-series forecasting.
    f. Deep learning calibration of financial models
    g. Investment portfolio optimization and spread trading.
    h. Reinforcement learning for trading and pricing .

Students will be required to program in Python to complete assignments and a research project. Basic knowledge in linear algebra, probability and statistics is required. The project will require reading and understanding research articles. The course helps prepare students for quant or research related positions.

SYLLABUS

Week 1: Introduction

- What is Machine Learning and how it is related to Artificial Intelligence?
- Differences between ML and Statistical Modeling
- Core paradigms of ML: Supervised, Unsupervised and Reinforcement Learning
- ML in Finance: main applications
- Differences between ML in Finance and ML for tech
- Setting test python environment for class examples and home assignments.

Week 2: Foundations of Machine Learning

- Generalization and bias-variance tradeoff
- Under-fitting and overfitting
- Regularization
- Linear Regression as a ML algorithm
- Probabilistic ML models: Maximum Likelihood, Maximum a-posteriory probabilities
- Hyperparameters and cross-validation

Week 3: Linear regression and classification

- Linear regression models
- Regression use case: implied volatility of options
- The bias-variance decomposition
- Regularization
- Bayesian linear regression
- Discriminative classification models: logistic regression and Bayesian logistic regression
- Generative classification models

Week 4: Decision tree models

- Building Decision Trees
- Classification and Regression Trees (CART)
- Random Forest Trees

- Ensemble learning: Boosted Trees
- Classification use case: prediction of whether income exceeds $50K/yr or not with trees

Week 5: Support Vector Machines

- Statistical learning theory: learning with generalization guarantees
- Maximum margin separation
- Kernel trick
- Support Vector Machines (SVM) for classification
- SVM for regression: Support Vector Regression (SVR)
- Are SVMs good for a large-scale ML?
- Regression use case: stock return prediction with SVR

Week 6: Feed-forward neural networks

- Perceptron
- Multi-layer (feed-forward) neural networks: universal function approximation
- Error backpropagation
- Optimization algorithms
- Deep neural networks (deep learning)
- Neural network use case: Applying Deep Learning to option pricing
-

Week 7: Feed-forward networks and calibration

- Deep Learning calibration

Week 8: Unsupervised learning and clustering methods

- Nearest Neighbor Methods (KNN, KD-trees)
- K-means clustering
- Hierarchical clustering methods
- Spectral clustering
- Self-Organized Maps (SOM)
- Manifold learning
- Clustering use case: Hierarchical clustering of stocks

Week 9: Unsupervised feature learning and deep earning

- The grand promise of deep learning: hand-engineered features no more!
- Restricted Boltzmann Machine (RBM)
- Deep Boltzmann Machines
- Dimensionality reduction with neural networks: the auto-encoder
- Neural networks use case: analysis of stocks with auto-encoders
- Neural networks use case: regime change detection with deep learning

Week 10: Sequence modeling with Hidden Markov Models (HMM) and Linear Dynamic Systems (LDS)

- Sequences and Autoregressions
- Moving averages and financial time-series
- State-Space Models and Linear Dynamic Systems
- Inference and learning in LDS
- Filtering, Kalman filter

Week 11: Sequence modeling with Hidden Markov Models (HMM) and Linear Dynamic Systems (LDS)

- Markov Models, Hidden Markov Models (HMM)
- Inference and Learning in HMM
- Recurrent Neural Networks (RNN)
- Long-Short Term Memory (LSTM) Networks
- Training RNN and LSTM
- Generative Adversarial Training
- Use case: Prediction bank closure with RNN and LSTM

Week 12: Latent variable models and dimensionality reduction

- Factor analysis
- Principal Component Analysis (PCA)
- Independent Component Analysis (ICA)
- Gaussian mixture models
- Expectation Maximization (EM) algorithm
- Variational autoencoders
- Dimensionality reduction use case: analysis of stock returns with PCA and ICA

Week 13: Neural networks for sequence modeling

- Q-learning
- RL approach to option pricing in Finance

COURSE PROJECTS AND PROGRAMMING ASSIGNMENTS

Students are expected to do all programming assignments and one of the course projects.

Programming assignments:

1. Ridge regression for implied volatility
2. Bias-variance tradeoff" or "bias-variance dilemma"
3. Prediction of the option prices and Greeks by using boost regressor
4. SVR with various kernels
5. Build an feed-forward ANN for the Black-Scholes model
6. Self-Organizing Maps
7. HMM models with Categorical HMM and Multinomial HMM
8. LDS
9. Autoencoders
10. Kalman filtering
11. QLBS and fitted q-iterations

Course project: Classification

- Based on stock market data, Self-Organizing Maps (SOMs) can be used to classify days by their **market behavior or characteristics**. Applications to momentum strategies in stock trading.

Course project: Deep-learning calibration

- Implement feed-forward ANN for the Heston model and fit it to the market data.

SUGGESTED TEXTBOOKS

- Machine Learning in Finance: From Theory to Practice – Matthew F. Dixon, Igor Halperin, Paul Bilokon, 2020.
- Jake VanderPlas (2017) Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly
- Media, Inc. (Publicly available online: https://jakevdp.github.io/PythonDataScienceHandbook/)
- Aurélien Géron (2019) Hands-on Machine Learning with Scikit-Learn, Kearas, and TensorFlow, 2nd Edition. O'Reilly Media, Inc.
- Machine Learning for Asset Managers (Elements in Quantitative Finance) Part of: Elements in Quantitative Finance (3 Books) by Marcos M. López de Prado, 2020.
- C. Bishop, "Pattern Recognition and Machine Learning" (2006)
- I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning" (2016).

For a refresher on linear algebra and probability theory in an amount needed for this course, see e.g. Chapters 2 and 3 in Goodfellow et. al., "Deep Learning" (2016)

Additional references:

- S. Marsland, " Machine Learning: An Algorithmic Perspective" (2009)
- K.P. Murphy, "Machine Learning: A Probabilistic Perspective" (2012)
- D. Barber, "Bayesian Reasoning and Machine Learning" (2012)
- N. Gershenfeld, "The Nature of Mathematical Modeling" (1999)

LEARNING OUTCOMES

GLOBAL:

The goal of this course is to expose the participant through lectures, readings, and hands-on homework to the following topics:

- Students understand the machine learning workflow.
- Students are familiar with different types of panel data encountered in finance: cross-sectional and sequential (time or location indexed). Brownian processes (random walks) and mean-reverting processes.
- Students understand the differences between: supervised vs. unsupervised, linear vs. non-linear, regression vs. classification, cross-sectional vs. sequential.
- Students can use multivariate regression, logistic regression, principal component analysis, support vector machines, decision trees, random forests, k-means, hierarchical clustering, Gaussian mixtures, multi-layer perceptron, recurrent neural networks, LSTMs, and auto-encoder neural networks.
- Students can understand the mathematical and algorithmic structure of the models, their assumptions and their purpose, their strengths and their weaknesses.
- Students can apply the machine learning models to credit modeling, time-series and financial time-series forecasting, investment portfolio design, spread trading, credit cycle regime identification.
- Students can utilize the financial metrics of model adequacy: profit or risk evaluation metrics associated with financial predictive models: information coefficient, Sharpe ratio, CAGR, annualized volatility, White Reality Check (a version of Superior Predictive Ability).

INSTRUCTIONAL:

After completing this course, participants will be able to use Python and out-of-the-box statistical learning libraries (e.g. Scikit-Learn, Keras/TensorFlow) to program a basic machine learning workflow applied to panel data and involving the following steps:

- Obtain free data from data providers such as Yahoo, Quandl etc. using Pandas-Datareader and Polars.
- Prepare the data into indexed dataframes using Pandas.Polars functions for date indexing, for hierarchical indexing, and for table management: pivot, join and merge.
- Engineer alpha-factors and risk-factors with specialized libraries including Ta-lib, FINTA, the Fama-Macbeth linear factor model, Pandas date processing functions.
- Engineer time-series decomposition features such as trend, seasonality, lookback window etc. using Pmdarima, HoltWinter. ExponentialSmoothing, simple and partial autocorrelation functions.
- Engineer categorical features with one-hot-encoding.
- Extract features using principal component analysis and autoencoders.
- Select the best features using Scikit-learn feature selection functions and Quantopian's Alphalens module.
- Scale the features using Scikit-learn functions.
- Construct machine learning workflows using Scikitlearn pipelines and Keras out-of-the-box functions including: splitting of data with train_test_split, model evaluation with cross-validation and model parameter tuning with grid-or-randomized-search-cross-validation.
- Apply these workflows to cross-sectional and time-series panel data using various types of models.
- Evaluate a model using simple statistical criteria (e.g. mean squared error, precision-recall), more sophisticated statistical criteria (e.g. bootstrap based), and financial criteria (Information coefficient, Sharpe ratio, CAGR, annualized volatility etc.)
- Display a model's feature importance and predictive adequacy using Scikit-Learn and Keras out-of- the-box functions and matplotlib.

STUDENT LEARNING ACTIVITIES

- Weekly lectures and discussions
- Weekly readings
- Programming homework: coding from scratch ML algorithms
- Group research project and presentation
- Peer-review of research projects

ASSESSMENTS

1. **Weekly Quizzes:** 10%
   Regular quizzes to assess understanding of weekly material.
2. **Programming Homework Assignments:** 50%
   Four mandatory coding assignments, each contributing 10% (optional fifth assignment available for additional credit).

3. **Course project:** 40%

This project includes several components:

- Complete Code Submission: 20%
- Final Report: 10%. Must include a detailed account of each member's contributions.
- 15-Minute Presentation: 10%

GRADING POLICY

The grades are reported based on what students earn by their work. Think of the requirements in this class as analogous to professional job expectations.

- A (A/A-) – Your work demonstrates exceptional quality, equivalent to performance that would put you on the fast track for a promotion in a professional setting.

- B (B+/B/B-) – Your work is solid and meets expectations, showing you as a reliable contributor but not necessarily a standout performer.

- C (C+/C) – Your work raises concerns about your long-term fit in a professional role, requiring significant improvement.

- F – Your work is unacceptable and would prompt serious repercussions in a workplace.

NYU Tandon’s graduate grading scale is: A, A-, B+, B, B-, C+, C, F.