Columbia University
Syllabus
COMS W4995-Topics in Computer Science:
Machine Learning with Applications in Finance

Semester: Summer 2023	
Instructor: Germán G. Creamer ggc14@columbia.edu OH: after every session, by appointment or Wed 12 PM -1.30 PM Zoom: 91996779628 Passcode: 314614	Schedule: MW 5:30pm-8:40pm Location: 833 Seeley W. Mudd Building Website: coursework
TAs/CAs sessions: Abhi Navinkumar Lad al4363@columbia.edu	 Saravanan Thanu st3523@columbia.edu

Overview:
The significant amount of information available in any field requires a systematic and analytical approach to select the most important information and anticipate major events. Machine learning algorithms facilitate this process of understanding, modeling, and forecasting the behavior of major social or economic systems and their variables.

This research course explores applying fundamental machine learning models for financial prediction, algorithmic trading, model calibration, and portfolio optimization.

Prerequisite: A Machine learning course or similar training.

Course Objectives
Students will:

- Learn how to adapt or develop machine learning algorithms for the finance domain.
- Explore existing and new applications of statistical learning methods such as forecasting or classification of financial problems.
- Learn how to preprocess financial data, generate and select features, and forecast financial time series for trading, portfolio optimization, and investment decisions.

List of Course Outcomes:

By the end of this course, the students will be able to:
- Apply statistical models and analytical methods to the finance domain.
- Recognize the value and limits of statistical learning algorithms to solve financial problems.
- Develop analytical models for financial forecasting, portfolio optimization, and trading.

Pedagogy
The class will combine class presentations, discussions, exercises, and case analysis to motivate students and train them in the appropriate use of statistical and machine learning techniques for investment and portfolio optimization.

Course Topics:

- Introduction to main topics of machine learning
- Introduction to credit risk
- Capital Allocation to Risky Assets
- Optimal Risky Portfolios
- Index Models
- The Capital Asset Pricing Model
- Arbitrage Pricing Theory and Multifactor Models of Risk and Return
- Innovating into Active ETFs
- Momentum Funds
- Empirical asset pricing and machine learning
- Natural language processing: Extracting information and investment signals
- Company valuations
- Portfolio Performance Evaluation
- Theory of Active Portfolio Management
- Black Litterman model
- Investment Policy
- Clustering analysis
- Portfolio optimization
- Cryptocurrencies and blockchain
- Microstructure and high-frequency finance
- Technical analysis and algorithmic trading
- Financial data structures and cross-validation
- Feature importance, model calibration & backtesting

Required Texts

- Zvie Bodie, Alex Kane and Alan Marcus, Investments, 13th Edition, McGraw-Hill.
- Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, Springer, 2nd edition, 2021 (link) (ISLR) (selected sections)
- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze, Introduction to Information Retrieval, Cambridge University Press. 2008 http://nlp.stanford.edu/IR-book (selected sections)

Required Harvard cases: You can buy the coursepack with the following cases at https://hbsp.harvard.edu/import/1058938

- AQR Momentum Funds A
- AQR's Delta strategy
- Valuing Walmart 2010
- Innovating into Active ETFs

Selected papers:

- S. Gu, B. Kelly, D. Xiu, Empirical Asset Pricing via MachineLearning, The Review of Financial Studies 33 (2020): 2223–2273. https://dachxiu.chicagobooth.edu/download/ML.pdf https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2668919
- Lauren Cohen, Christopher Malloy & Quoc Nguyen, Lazy Prices, NBER paper # 25084, 2019. https://www.nber.org/papers/w25084
- Eisdorfer, Assaf and Froot, Kenneth and Ozik, Gideon and Sadka, Ronnie, Competition Links and Stock Returns 2019. https://ssrn.com/abstract=3469642
- G. Creamer (2015). "Can a Corporate Network and News Sentiment Improve Portfolio Optimization Using the Black Litterman Model?" Quantitative Finance 15 (8): 1405-1416. https://ssrn.com/abstract=2668919
- Marco Lopez de Prado, Building Diversified Portfolios that Outperform Out-of-Sample
  Journal of Portfolio Management, 2016
  https://jpm.pm-research.com/content/42/4/59
- G. Creamer and Y. Freund (2007). "A Boosting Approach for Automated Trading." Journal of Trading 2 (3): 84-96. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=938042
- Johnson, chapter 1, Algorithmic Trading and DMA: An introduction to direct access trading strategies http://www.mediafire.com/file/kxa9gve6fxccbg6/algo-dma_preview.pdf
- G. Creamer (2012). "Model Calibration and Automated Trading Agent for Euro Futures." Quantitative Finance 12 (4): 531-545. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2028677

Optional references:

- Trevor Hastie, Robert Tibshirani and Jerome Friedman, The Elements of Statistical Learning. Springer-Verlag, 2nd. Ed., New York, 2009 (link).
- Marco López de Prado, Advances in Financial Machine Learning, Wiley, 2018.
- Andrew Ang, Asset Management: A Systematic Approach to Factor Investing, Oxford University Press, Oxford, 2014.

Software

The programming language of this course is Python with the Anaconda distribution: https://www.anaconda.com. We will have an introductory session about Python; however, if you are not proficient in Python, you should participate in a campus or online training session or follow one of the tutorials suggested in our training session.

Assignments and evaluation:

This is an applied research-oriented course structured around several assignments distributed during the semester that cover the main topics of the course. The emphasis of these assignments is on the ability of students to implement analytical solutions to relevant industry problems. The assignments must be submitted electronically through the course website. Each student must submit his/her report. E-mail submissions will not be accepted.

For all the programming homeworks and unless indicated otherwise, you should submit two UNCOMPRESSED files: a report as a Jupyter notebook saved as an HTML or pdf file organized by questions, and the Python code file (File --> Download as). Create your tables with the output of your program and EXPLAIN the results. Failure to include BOTH the report (Jupyter notebook) and the code file will result in a deduction of 50% for each component missing. So, for example, if you do an assignment perfectly, but don't provide the report file, you will receive at most 50% of the grade on the assignment. If you want to improve your homework, you can resubmit it until the deadline.

Do not send sections of your code or ask a complex homework question by email. I cannot debug your program or write a long explanation by email. However, you are welcome to ask any questions about the homework or any other issue related to this class during class, after class, or during office hours.

Grades

Assignment	**Grade %**
Assignments & project	69%
Quizzes & exam	30%
Completion final evaluation	1%

The final grade will be determined according to the following scale:

A	>= 94.0%
A-	< 94.0% to 90.0%
B+	< 90.0% to 87.0%
B	< 87.0% to 84.0%
B-	< 84.0% to 80.0%
C+	< 80.0% to 77.0%
C	< 77.0% to 74.0%
C-	< 74.0% to 70.0%
D+	< 70.0% to 67.0%
D	< 67.0% to 64.0%
D-	< 64.0% to 61.0%
F	< 61.0% to 0.0%

Participation/Attendance

Class attendance is mandatory for on-campus students. Regulatory attendance is considered 10/12 of class sessions. A point (out of 100) will be deducted for every unjustified missing session after two absences or a half-point for late or partial attendance. Students that miss more than 50% or more of their classes (unexcused) will receive an F. In the case of Covid-19 diagnosis, we follow the current school's policies.

Absences are only excused in documented cases of medical or personal/family emergencies or religious holidays. Students must provide notification in advance of absence from class when possible.

You should upload your picture in Canvas and bring a name card to every session.

Class policy
Late Policy: 10% of the grade lost for a late submission. No assignments are accepted after 1 day.

Re-grades: If you dispute the grade received for an assignment, you must submit, by an email directed to the course instructor, your detailed and clearly stated argument for what you believe is incorrect and why (**DO NOT SUBMIT YOUR REQUEST as a COMMENT on the course's website**). This must be submitted by the beginning of the next class after the assignment was returned. Requests for re-grade after the beginning of class will not be accepted. A written response will be provided by the next class indicating your final score. Be aware that requests for a re-grade of a specific problem can result in a regrade of the entire assignment. This re-grade and a written response are final, no additional re-grades or debate for that assignment.

Extra Credit: Possibly on an assignment, there will be the occasional extra credit problem. These are the only source of extra credit for the course. There are no "extra assignments" that students can do to raise their average outside of the ones assigned. There are no exceptions.

Ethics and Cooperation: You are allowed to discuss lecture and textbook materials, and how to approach assignments.

You cannot share ideas in any written form: code, pseudocode, or solutions. You cannot submit someone else's work found through the internet or any other source, or a modification of that work, with or without that person's knowledge, regardless of the circumstances under which it was obtained, copied or modified. Of course, no cooperation is allowed during exams. This policy will be strictly enforced. See the Computer Science department's Policies and Procedures Regarding Academic Honesty for details.