# DATA SCIENCE II:
## Machine Learning
## MTH 9899
## Baruch College

Spring 2017

| | | | |
|---|---|---|---|
| **Instructor:** | Adrian Sisser | **Email:** | Adrian.Sisser@baruch.cuny.edu |
| **Time:** | W 18:05 – 21:00 | **Room:** | 9-140, (24 St & Lexington Ave) |
| **TA:** | David Zhang | **Email:** | davidzhang.wyx0709@gmail.com |

**Office Hours:** By appointment before class.

**Main Reference:** The main text for the course is below. In addition, there will be occasional links provided to relevant supplemental material.

- Trevor J. Hatie and Robert John Tibshirani and Jerome H. Friedman, *The Elements Of Statistical Learning : Data Mining, Inference, and Prediction* - Available here .

**Objectives:** This course should teach you machine learning with applications to finance. There will be an emphasis on implementation of several algorithms that are widely uses in Finance.

**Prerequisites:**

- An undergraduate-level understanding of Probability, Statistics, Graph Theory, Algorithms, and Linear Algebra.
- Some knowledge of Python is highly preferable but can be picked up during the course (we are not going to teach it).

**Grading Policy:** Homework and quizzes (40%), Final Project (50%), instructor discretion (10%). At the end of the course, the lowest grade from any quiz or homework will be automatically dropped.

**Class Policy:**

- Regular attendance is essential and expected.
- We will do our best to provide a 10 to 15 minute break in the middle of class.

**Academic Honesty:** Lack of knowledge of the academic honesty policy is not a reasonable explanation for a violation. We take it seriously.

**Overview:** Machine learning is a very broad area of study that can't be completely covered in a single course. We will give very brief coverage to a lot of different algorithms, while providing more extensive focus on two that have found widespread practice inside of finance, Regression Trees and Neural Networks/Deep Learning. Lectures will occasionally be supplemented with online lecture videos from various sources that will be provided throughout the class. We will then use lecture to reinforce these lectures and focus more deeply on the details of certain algorithms and applications to finance. The supplemental lectures are mandatory, and the material learned in them will be used in periodic quizzes. In addition, they will be covered in homeworks and might be used in projects. While implementing Neural Networks and Deep

Learning algorithms, we'll use the Keras library as a technology platform to allow us to build and test interesting models.

**Homework:** Most homework assignments will consist of using and implementing various ML algos on datasets that will be provided. The goal is to implement algorithms as well as use existing tools to understand what techniques work in different problems. There will approximately 4 homework assignments over the class. Homework assignments will be due at 6PM on the day of class. For each student, we will allow 1 late (up to 24 hours) homework assignment per semester, and NO credit will be received for other late homeworks. Homeworks can be worked on collaboratively, but not copied. The names of collaborators must be included. **All code for homework will follow several rules:**

- All code must be Python 3.x compatible. Python 2.7 code will NOT be accepted.
- Code should NOT use packages other than: numpy, scipy, sklearn, keras, and other basic Python packages. If you need to use another package, please let us know ahead of time. To facilitate this, we highly recommend that you work in virtual environment, with a basic set of packages only. We are open to expanding this list to other common packages.
- IPython notebooks are preferred.
- All work should be submitted as a .tgz file and the data, etc. referenced from the notebook should be located in the correct paths relative to .tgz file
- **If you don't follow these rules, it's incredibly hard and time-consuming for us to get your code to run. If we can't get it to run, you can't get a grade**

**Quizzes:** Quizzes will be given at the start of class and will include topics covered in assigned readings and supplemental materials up through that day. All quizzes will be announced at least 48 hours in advance, if you have any issues making it to class on time for the quiz, please let us know at least 24 hours in advance and we will try to make arrangements.

**Projects:** The class will work on a large project in assigned groups of 3 or 4. The subject for the projects will be real financial data sets, and the goal will be to apply the ML algos to build models essential to successful quantitative trading. As part of the projects, students will model the datasets using existing analysis tools in Python or write new and innovative algorithms. At the end of the course, groups will present their findings to the class.

**Optimistic Course Outline:** Below is a list of topics we hope to cover. Please note that after the first 2 lectures, this is only a general outline of topics we hope to cover.

- Lecture 1 - General topics
  - Supervised vs Unsupervised Learning
  - Classification vs Regression
  - Cross-Validation
  - Fit-quality Measurements
  - Bias-Variance Trade-off
  - Ridge Regression
  - Introduction to Neural Networks?
- Lecture 2 - Linear Models and More
  - Logistic Regression

  - Least Squares, Partial Least Squares, Lasso and Elastic Net Regression
  - Least Angle Regression
  - Basis Expansion
  - Principal Component Analysis, Factor Models and Dimensionality Reduction
  - Nearest Neighbors
  - EM Algorithm
  - Backpropagation
  - Optimization Techniques - Stochastic Gradient Descent and Batch Learning
- Other Topics in Trees + Boosting/Ensemble Methods
  - ID3/C4.5/C5.0 Algorithms
  - Model-Based Trees
  - Ensemble Learning Methods
  - Boosting
  - Bootstrapping and Subsampling
- Other Topics in Deep Learning
  - Keras - A Python Framework for Deep Learning
  - Restricted Boltzman Machines
  - Neural Networks
  - Multilayer Perceptrons
  - Convolutional Neural Networks
  - Reccurent Neural Network (LSTM)
  - Bidirectional Reccurent Neural Network