Project Update

Zane Jakobs
Michael Wegner
Sarah DeNardis

April 16, 2019

Final Deliverables

We do not feel like a written paper at the level of detail necessary to fully describe the details of portfolio optimization is the best form of final deliverable. The truly interesting mathematics behind portfolio optimization–time series modeling and forecasting, Ito calculus, and advanced numerical optimization techniques, to name a subset–either cannot be taken advantage of without access to much higher-frequency data than is freely available in large enough datasets, or while interesting in its own right, is well beyond the scope of this course. The process of implementing models that were once state-of-the-art, like Markowitz portfolio optimization, is much more relevant to the skills and techniques taught in APPM 4120. As such, we propose that our final deliverables consist of a small Python module/library of functions and classes, and scripts implementing and testing each of several optimization techniques, and a short (order of 2-3 pages) summary of the model performance, along with a few graphics to illustrate said performance. If you disagree with the assessment that a paper is not the best deliverable, please notify us as soon as possible, and we will write one.

What we Have Already Accomplished

As of Tuesday, April 16, we have written code to perform all of the following tasks:

- Import forecast data from Zane's project for APPM 3400 Applied Regression.
- Import market data
- Compute the sample variance of equity returns (note: this will be replaced by the covariance matrix forecasts from Zane's Applied Regression project when they are available later this week or early next week)
- Optimize a portfolio in the Markowitz mean-variance framework, using Gurobi's quadratic programming engine.
- Compute n-day returns
- Compute observed returns of a given portfolio over a given time period
- Compute the daily sample variance of a given portfolio under the assumption of normally-distributed returns. (This assumption is manifestly untrue of our data in most cases; this again will be replaced by a more robust estimator based on Markov Chain Monte Carlo simulation with the perfect independent Metropolis Hastings algorithm.)
- Select the best equities to use for optimization, which we currently define as the n most positive and m most negative forecast returns, for natural numbers n and m (the default parameter is currently 200 each).

This code is available at `https://github.com/MichaelWegnerCU/ORProj`

What we Are Currently Working on

- Optimize a portfolio with expected returns only (which is a linear programming problem)
- Backtesting and model evaluation with the Zipline library for quantitative finance
- Library/module structure, including type checking and other safety mechanisms

What we Still Have to Do

- Graphics, including time series plots of portfolio performance per time
- The final scripts to run the functions we've written.
- Implement Value at Risk (VaR) as a portfolio risk metric, and an appropriate backtesting framework for VaR
- Test code for correctness and performance

What We Would Like To Do If We Finish All Of The Above Early

- Implement a more sophisticated optimizer that accounts for cumulants beyond the mean and variance (e.g. skewness and kurtosis)
- Implement a more sophisticated backtesting environment that accounts for factors like slippage, taxes, and brokerage fees.