GSoC 2025 - PorQua Mentor Review

Arsh Tulshyan

March 2025

Project

This is a rough framework of all I'll be doing within this project.

Project Overview

My project aims to advance **PorQua's portfolio optimization, machine learning, and deliverables**, enhancing its theoretical and practical foundations to create a robust framework for real-world investment strategies.

Portfolio Optimization Framework

I will develop **LinearProgram** and **ConvexProgram** classes as solver wrappers, mirroring `QuadraticProgram`'s structure—solving:

$$\min c^T w \quad \text{and} \quad \min f(w) \quad \text{subject to} \quad Aw = b, \quad Gw \leq h$$

via **qpsolvers** (e.g., **CVXPY, Gurobi**) to support linear and convex problems alongside quadratic ones, ensuring solver flexibility and a unified API for seamless problem definition and solution.

Portfolio Optimization Objectives

I'll define **extensible objective classes**:

- **MaximumUtility** will optimize a logarithmic utility function:
$$U(w) = \ln(w^T r)$$
to reflect investor risk preferences.
- **MaximumRatio** will target the **Sharpe ratio**:
$$\frac{w^T r - r_f}{\sqrt{w^T \Sigma w}}$$
and explore **Sortino or Omega ratios**.
- **MinimumRisk** will minimize metrics like **variance** $w^T \Sigma w$, **CVaR**:
$$\alpha^{-1} \int_{-\infty}^{\text{VaR}} x \, dF(x)$$
or **tracking error** $\|w^T r - b\|_2^2$.
- **EqualRiskContribution** will balance **marginal risk contributions**:
$$w_i \frac{\partial \sigma}{\partial w_i}$$
across assets, all formulated to integrate with the optimization classes.

Multi-Objective Optimization

I'll enable **objective combinations**, such as **maximizing utility while minimizing risk**, using weighted objectives:

$$\lambda U(w) - (1 - \lambda) w^T \Sigma w$$

to explore **trade-offs within a convex or quadratic framework**, ensuring theoretical coherence across solvers.

Constraints Implementation

I'll enhance constraints with **risk contribution bounds**, enforcing:

$$w_i \frac{\partial \sigma}{\partial w_i} \leq \kappa$$

to prevent risk concentration. This complements **QuadraticProgram's turnover linearization**:

$$|w - w_{\text{init}}| \leq \tau$$

and introduces a **liquidity constraint**:

$$w_i \leq \frac{V_i}{V_{\text{total}}} \cdot 0.05$$

which will be analyzed for impact on feasible sets and solver performance.

Machine Learning - Time Series Prediction (LSTM)

I'll implement **LSTM networks** to forecast **stock returns** r_{t+1} from historical sequences, designing preprocessing pipelines to normalize inputs (e.g., **log-returns**) and training models across horizons (e.g., **1-day, 5-day**), with theoretical validation against statistical benchmarks.

Machine Learning - Learning to Rank (LTR)

I'll develop an **LTR model** (e.g., **RankNet or LambdaRank**) to **order stocks by expected returns**, optimizing a **pairwise ranking loss function**:

$$-\sum_{i>j} \ln(1 + e^{-(s_i - s_j)})$$

ensuring alignment with portfolio objectives through **ranked weight assignments**.

Machine Learning - Feature Generation

I'll create a suite of **technical indicators**—e.g., **moving averages, RSI, volatility**:

$$\sigma_t = \sqrt{\frac{1}{n} \sum (r_t - \bar{r})^2}$$

to enrich ML inputs, grounded in financial theory to **capture market dynamics**.

Machine Learning - Feature Selection

I'll apply **Recursive Feature Elimination (RFE), Lasso**:

$$\min \|y - Xw\|_2^2 + \lambda \|w\|_1$$

and **Random Forest importance scores** to select **the most predictive features**, enhancing model efficiency and interpretability.

Deliverables

I will deliver:

- ✓ **A full-featured optimization engine** spanning **linear, convex, and quadratic** problems.
- ✓ **ML models (LSTM, LTR)** for return prediction and ranking.
- ✓ **A technical feature suite** for model training and portfolio construction.
- ✓ **Detailed documentation with tutorials**, ensuring accessibility for quants and newcomers alike.

Theoretical Approach

1. **Analyze QuadraticProgram's QP structure** and refine turnover/liquidity constraints.
2. **Extend to LinearProgram and ConvexProgram**, integrating ML predictions into optimization objectives.
3. **Develop a backtesting system** to assess **real-world performance**, ensuring solver-agnostic compatibility and mathematical rigor throughout.

Personal Data

Name: Arsh Tulshyan
Affiliation: B.Tech in Computer Engineering, Vidyalankar Institute of Technology
Location: Mumbai, India

Studies and Programming Courses I've Taken

- Analysis of Algorithms & Advanced Data Structures
- Data Structures & OOP (Java)
- Structured Programming (C) & Python
- DBMS & Computer Networks
- Operating Systems & Theory of Computation
- Discrete Mathematics & Probability/Statistics
- Linear Algebra & Complex Numbers
- Microprocessors & Computer Graphics
- Digital Logic Circuits & Hardware Fundamentals
- Quantum Physics & Professional Skills

Some Projects I've Worked On

- Advent of Code 2024
- AI Crop Disease Prediction
- International Quant Championship (IQC)

My Best Skills in Programming & Scientific Computing

- **Programming:** C++, Python, Java
- Algorithms & Data Structures
- Scientific Computing
- Machine Learning & AI
- Quantitative Finance
- Software Development
- Performance Optimization

My Programming Style

I mostly **code in C++** because of how powerful and fast it is. Python is great too, but honestly, it sometimes feels like a cheat code. My approach is to **think through the best solution first, then start coding**.

Availability During the Program

No conflicts—I'm fully committed to this project throughout the summer and will be working on it every day.

My Plans After GSoC

I definitely see myself **continuing to contribute to PorQua** even after GSoC ends.

Am I More of a Theorist or a Hacker?

I'd say I have a **hacker mindset**, but with GeomScale, I'm drawn to the **deep mathematical and financial statistics side** because it fits perfectly with my goal of working in quant.

Hoping to hear back soon 72