Homework: Smooth and Nonsmooth Markowitz Portfolio Optimization

LINMA2471: Optimization Models and Methods II

Objective and Context

This project introduces one of the most influential applications of continuous optimization in finance: the **Markowitz portfolio selection problem**. The goal is to decide how to allocate wealth among different assets in order to balance *expected return* and *risk*. Through this homework, you will connect the theoretical and algorithmic concepts studied in class to a concrete optimization problem that has shaped modern financial theory.

Your task is to analyze, implement, and compare optimization algorithms for both the classical smooth Markowitz model and a nonsmooth variant including transaction costs. You will produce a self-contained short article that presents the problem, explains the methods, and discusses the obtained results in a clear and critical way.

Problem Formulations

Let $w \in \mathbb{R}^n$ denote the portfolio weights (fractions of total wealth invested in each asset). Let $\mu \in \mathbb{R}^n$ be the vector of expected returns and $\Sigma \in \mathbb{R}^{n\times n}$ the covariance matrix of asset returns. Let $\lambda > 0$ and $c > 0$ be user-defined parameters controlling risk aversion and transaction sensitivity, respectively. Let $w_{\text{prev}} \in \mathbb{R}^n$ denote the portfolio held at the previous rebalancing date, which serves as a reference point in the model with transaction costs. Finally, define the feasible set (the simplex)

$$\Delta = \{w \in \mathbb{R}^n \mid w_i \geq 0,\ \mathbf{1}^\top w = 1\}.$$

Two models will be studied.

Smooth mean–variance model

$$\min_{w\in\Delta} f(w) = \tfrac{1}{2} w^\top \Sigma w - \lambda w^\top \mu.$$

Nonsmooth model with transaction costs

$$\min_{w\in\Delta} \tilde{f}(w) = \tfrac{1}{2} w^\top \Sigma w - \lambda w^\top \mu + c\|w - w_{\text{prev}}\|_1.$$

The first formulation corresponds to the classical mean–variance model proposed by Markowitz (1952). The second one introduces transaction costs or rebalancing frictions, making the problem more realistic but also nonsmooth. Both problems are convex and can be addressed with both the first-order and second-order methods studied in class.

General Directions

Your report will contain a clear theoretical and numerical study of both models. You are expected to:

- Explain and interpret the mathematical formulation of each model, including the role of Σ, μ, λ, c, and w_{prev}.
- Derive and discuss relevant analytical elements (gradients, Lipschitz constant, projection operator onto the simplex[1], proximal operator, self-concordant barrier functions, etc.).
- Implement the following optimization algorithms:
  - For the smooth case: (1) Projected Gradient Descent, (2) Projected Gradient Descent with Momentum (Heavy Ball), and (3) Projected Randomized Coordinate Descent
  - For the nonsmooth case: (1) Projected Subgradient Method, (2) Proximal Gradient Method and (3) Long-Step Path-Following Interior-Point method
- Discuss for each method hyperparameter choices (e.g. step sizes), convergence behavior, and how well the theoretical guarantees match the observed empirical performance.
- Compare the performance of theses algorithms to achieve a given solution quality, for example using plots. Pick a reasonable common starting point. The fact that iterations may have very different computational costs for different methods should not be neglected in the comparison.
- Pick the most efficient algorithm for each model and
  - Comment on the structure of the obtained portfolios and on the influence of the parameters λ and c.
  - Analyze the differences between the smooth and nonsmooth settings and interpret them in financial and algorithmic terms.

You are encouraged to go beyond the minimal requirements by testing additional algorithms or parameter variations (including heuristic algorithms not covered by theory if you wish), provided that the core elements above are covered.

All numerical experiments should be reproducible.

Data and Implementation

Use the provided financial data from the S&P500 stocks (CSV file from `https://www.kaggle.com/datasets/camnugent/sandp500`, also provided on Moodle). You should try to use as much data as possible (the more assets and time periods you consider, the better). From these data, compute:

$$\mu = \text{mean of returns}, \qquad \Sigma = \text{covariance matrix of returns}.$$

[1] *Hint*: The solution of the projection $P_\Delta(v) = \arg\min_{w\in\Delta} \|w - v\|_2^2$ can be written as $w_i = \max(v_i - \theta, 0)$, for some threshold θ that nees to be identified.

Briefly explain how you constructed these quantities, your choice of time window and frequency, and any preprocessing or regularization used to ensure that Σ is positive semidefinite.

For the nonsmooth model, you will also need a reference portfolio w_{prev}. You may take

$$w_{\text{prev}} = \frac{1}{n}\mathbf{1},$$

corresponding to an equally weighted portfolio held before rebalancing, but you can choose to also explore a different scenario.

Report and Deliverables

You must submit:

- A report written in the style of a short research article. It should be coherent and self-contained, rather than a sequence of answers. One should be able to read it by itself, without referring to this file with instructions and guidelines.

  It must contain clear explanations, mathematical justifications when relevant, and well-presented numerical results (tables, plots, convergence curves, etc.). A good report will not only present results, but also interpret them critically, both in light of optimization theory and your intuition in the application area.

  The report must be (**8-10 pages, double-column format**) and use the provided template, with *the same* section layout. You are free to decide how to organize the content within each section.

- A **runnable Jupyter Notebook** (or equivalent code) that allows us to reproduce all experiments and results, i.e. regenerate all figures and tables in your report.

Platform and authenticity. All reports must be written using the official **INGI Overleaf server** at `https://overleaf.info.ucl.ac.be/login`. To ensure the authenticity of the work and allow verification of the project history, each team must **add the teaching assistants as collaborators** to their Overleaf project when submitting the homework. This access is used solely to inspect the editing history and confirm that the report was genuinely written by the students, and not fully generated externally (for example by large language models). Projects written outside the official platform or without shared access for verification may not be accepted for grading.

Evaluation criteria: clarity and rigor of explanations, correctness of the analysis, quality of the implementation and experiments, and overall readability and coherence of the report.

Changelog. 2025-11-14: initial release.