AR&P

SocioEconomic Challenges (SEC)
Volume 10, Issue 2, 2026 ISSN: 2520-6214
https://armgpublishing.com/journals/sec/

Addressing Socioeconomic Challenges through Robust Portfolio Optimization: Integrating the P versus NP Problem, the Markowitz–Capital Asset Pricing Model Framework, Cardinality Constraints, and Black–Scholes Derivative Pricing

Davit GONDAURI (corresponding author, Dgondauri@gmail.com), Doctor of Business Administration, Professor, Researcher at the Research Center, Business School, Business & Technology University, Tbilisi, Georgia

Type of manuscript: research paper

Received 13 of March, 2026
Accepted 18 of May, 2026
Published 30 of June, 2026

Keywords: socioeconomic challenges, portfolio optimization, P versus NP, cardinality constraints, Markowitz-CAPM, mixed-integer quadratic programming, Black–Scholes model, systemic risk.

JEL Classification: G11, G12, C61, C63, C15, G13.

Founder and Publisher: Academic Research and Publishing UG (i.G.), Germany

Abstract: *The growing complexity of financial decision-making under real-world constraints makes addressing socioeconomic challenges related to capital allocation efficiency, financial stability, systemic risk containment, and computational feasibility in portfolio optimization increasingly important. A critical gap in the literature on socioeconomic challenges lies in the insufficient integration of computational complexity, specifically the P versus NP problem, into empirically grounded portfolio optimization frameworks under realistic cardinality constraints and derivative instruments. The aim of this study is to develop an integrated economic-computational framework that operationalizes the P versus NP problem within a Markowitz–capital asset pricing model (CAPM) setting augmented by cardinality constraints and Black–Scholes derivative pricing. The empirical analysis is based on cross-sectional data for approximately 94 U.S. industry portfolios, including levered betas, annualized volatilities, firm-count information, and CAPM-implied expected returns, with a fully reconstructible covariance matrix derived through a single-index market model. The study employs mixed-integer quadratic programming (MIQP) logic alongside heuristic and metaheuristic approaches, including greedy selection, Monte Carlo (MC) sampling, and genetic algorithm (GA) search, implemented through a reproducibility protocol with random-seed control, distributional performance evaluation, convergence diagnostics, runtime profiling and interquartile-range (IQR) reporting. The results demonstrate that under a cardinality constraint of K = 10, the maximum Sharpe ratio reaches 0.0880, with expected return μ_p = 0.0868 and risk σ_p = 0.5356, confirming that combinatorial restrictions materially reshape the efficient frontier. Dependence diagnostics reveal strong systemic correlation, with median pairwise correlation equal to 0.7692 and the leading eigenvalue explaining 76.10% of total variance, indicating structurally limited diversification potential under the single-index covariance geometry. Sensitivity analysis shows that increasing the equity risk premium (ERP) by 100 basis points raises the median Sharpe ratio from 0.0866 to 0.1070, while variations in the risk-free rate (R_f) do not affect Sharpe under the CAPM structure, highlighting parameter sensitivity in optimization outcomes. The findings suggest that improved computational approaches to constrained portfolio optimization can help address socioeconomic challenges by enhancing capital allocation efficiency, strengthening portfolio-level risk diagnostics, and reducing systemic vulnerability in financially constrained and computationally complex decision-making environments.*

Cite as: Gondauri, D. (2026). Addressing socioeconomic challenges through robust portfolio optimization: Integrating the P versus NP problem, the Markowitz–capital asset pricing model framework, cardinality constraints, and Black–Scholes derivative pricing. *SocioEconomic Challenges, 10*(2), 37-70. https://doi.org/10.61093/sec.10(2).37-70.2026.

Copyright: © 2026 by the author(s). This is an Open Access article, distributed under the terms of the Creative Commons Attribution 4.0 International license (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution, and reproduction in any medium, provided the original work is properly cited.

INTRODUCTION

Modern portfolio construction is not only a technical problem in finance but also a socioeconomic challenge of allocating capital under institutional, governance, and market stability constraints. Capital-allocation decisions affect savings, pension, and institutional portfolios, financial stability, and the capacity of markets to absorb shocks. The Markowitz mean-variance paradigm and the capital asset pricing model (CAPM) are used here as a transparent risk-return scaffold, while the central emphasis of the Introduction is placed on why sparse and implementable portfolio mandates matter for capital-allocation efficiency, financial stability, and systemic-risk containment.

The practical difficulty arises because real portfolios are rarely implemented as unlimited allocations to all available assets. Position limits, monitoring capacity, liquidity constraints, and concentration rules force decision-makers to select a limited subset of instruments, which creates a direct trade-off between diversification and implementability. The detailed mixed-integer quadratic programming (MIQP) formulation and computational-complexity proof are therefore reserved for the methodology section, so that the introduction remains focused on the socioeconomic motivation and institutional relevance of the research problem.

Derivative instruments add a further socioeconomic dimension because they can transfer, reshape, or amplify risk within portfolios. The Black–Scholes component is therefore introduced as a controlled and auditable overlay rather than as a full derivatives-pricing engine: it tests whether nonlinear exposure can be incorporated into a constrained mean-variance allocation without mechanically inflating performance through leverage. This framing links financial realism, computational feasibility, and socioeconomic relevance within one reproducible research design.

The research question is operational: given an empirically grounded universe of industry portfolios, how does the computational hardness induced by a hard cardinality constraint shape the attainable efficient frontier, the choice and convergence behavior of algorithms, and the interpretability of portfolio composition when the universe is also augmented by a derivative instrument? By coupling the Markowitz-CAPM framework with explicit complexity framing, the study integrates portfolio theory, algorithmic search, and socioeconomic relevance into a single reproducible research design.

The empirical setting uses industry-level U.S. portfolio inputs to represent institutional allocation choices across economically relevant sectors. The introduction, therefore, focuses on why sparse portfolio mandates matter for capital-allocation efficiency, financial stability, governance capacity, and systemic-risk diagnostics. The detailed CAPM calibration, covariance construction, and optimization mechanics are reserved for the Methodology section, where they can be audited without overloading the introductory argument.

The study makes five contributions that strengthen the theoretical, empirical, and computational transparency of constrained portfolio optimization:

- First, the study anchors the cardinality-constrained mean–variance program in computational-complexity theory by formalizing an explicit decision version and positioning it within the NP-complete/NP-hard taxonomy, thereby clarifying why exact global optimization is not expected to scale in n and K.
- Second, all portfolio metrics, including μ_p , σ_p and the Sharpe ratio, are computed from a documented CAPM calibration and a fully specified covariance matrix Σ, which enables independent reconstruction and audit.
- Third, the stochastic search evidence is reported through distributional statistics, including the median, IQR and quantiles across multiple random seeds, alongside convergence-effort curves and runtime profiling.
- Fourth, correlation and covariance summaries and heatmaps are treated as explanatory diagnostics that connect algorithmic outputs, especially selected subsets, to the geometry implied by the covariance model.
- Fifth, a Black–Scholes European call worked example is embedded into the same Markowitz–CAPM instance through delta-based linearization, with moneyness-maturity robustness checks and a bump test used to quantify approximation error.

LITERATURE REVIEW

Foundations of portfolio theory: mean–variance efficiency

Modern quantitative portfolio selection begins with the mean–variance paradigm, in which the investor chooses portfolio weights to trade expected return against variance as a proxy for risk. The seminal contribution formalizes diversification, the efficient frontier, and tangency-portfolio logic under quadratic risk and linear return aggregation (Markowitz, 1952, 1959).

Later syntheses clarify that mean–variance efficiency is exact under elliptically distributed returns (or quadratic utility) and becomes an approximation otherwise; nevertheless, it remains the canonical benchmark for constrained allocation studies because the objective and constraints are explicit, auditable, and computationally tractable in its unconstrained convex form (Bodie et al., 2021; Elton et al., 2014; Boyd & Vandenberghe, 2004).

The unconstrained Markowitz program is solvable via convex quadratic programming, with a continuous feasible set that yields a smooth frontier parameterized by a risk-aversion scalar or by a target-return grid. Convexity is central: it is what makes efficient-frontier computation polynomial-time in the classical setting (Boyd & Vandenberghe, 2004; Nesterov & Nemirovskii, 1994).

Expected return calibration: CAPM and modern asset-pricing perspectives

To operationalize mean–variance optimization, expected returns must be specified. The capital asset pricing model (CAPM) provides a parsimonious mapping from systematic risk exposure to expected excess return, enabling transparent calibration of μ through β-loadings and an equity risk premium (Lintner, 1965; Mossin, 1966; Sharpe, 1964).

Empirical critiques and extensions highlight that single-factor CAPM is often incomplete in explaining the cross-section; multi-factor models (notably size and value) and broader asset-pricing frameworks provide alternative risk-premium decompositions and may materially change implied μ vectors (Cochrane, 2005; Fama & French, 1993, 2004).

In the present study, CAPM is used as an auditable, reconstructable prior for μ, not as a claim of realized performance. This aligns with applied optimization practice: a disciplined parametric return model is used to isolate computational and constraint-induced geometry, and the resulting conclusions are then subjected to sensitivity tests on premium and rate inputs (DeMiguel et al., 2009; Fama & French, 2004; Elton et al., 2014).

Risk modeling and dependence structure: covariance estimation and factor models

Covariance estimation is the second pillar of mean–variance analysis. Classical implementations rely on sample covariances, but finite-sample error can destabilize optimized weights; this motivates shrinkage estimators, Bayesian adjustments, and factor-structure approaches that trade bias for variance reduction (Jorion, 1986; Ledoit & Wolf, 2004).

The single-index (market) model provides an analytically convenient factorization, decomposing each asset return into a market-driven component and an idiosyncratic residual. This yields a covariance structure with rank-one commonality plus diagonal idiosyncratic terms—attractive when full return histories are unavailable but β and σ inputs are (Sharpe, 1963; Elton et al., 2014).

While factor models enhance interpretability, they can compress the effective diversification space when the common factor dominates. Dynamic correlation models and multi-factor risk models provide richer dependence representations and are frequently used as robustness alternatives when single-index structure may overimpose comovement (Campbell et al., 1997; Engle, 2002).

Real-world constraints and the emergence of mixed-integer portfolio optimization

Institutional portfolios face constraints absent from the idealized Markowitz program: limited position counts (cardinality), buy-in thresholds, minimum/maximum holdings, sector caps, turnover bounds, and discrete-lot trading constraints. These restrictions are empirically motivated (monitoring capacity, liquidity, transaction costs, governance) but they fundamentally alter the optimization landscape (Jobst et al., 2001; Mansini & Speranza, 1999; Schaerf, 2002).

A hard cardinality constraint, $|\text{supp}(w)| \leq K$, introduces binary selection variables coupled to continuous weights, producing a mixed-integer

quadratic program (MIQP). This discrete–continuous coupling breaks convexity and makes the efficient frontier discontinuous under many practical restrictions. Consequently, the literature develops exact methods (branch-and-bound/cut) and heuristic approximations (Bertsimas & Shioda, 2009; Chang et al., 2000; Gao & Li, 2013).

Computational complexity: P vs NP framing, NP-hardness, and sparse quadratic programs

The P vs NP framework distinguishes problems whose candidate solutions can be verified efficiently from problems whose optimal solutions are not known to be computable in polynomial time. The classical foundations of NP-completeness and polynomial-time reducibility are established by Cook (1971) and Karp (1972). General treatments of computational complexity provide the taxonomy used to position applied optimization tasks within NP-hard and NP-complete classes (Garey & Johnson, 1979; Papadimitriou, 1994; Arora & Barak, 2009).

Cardinality-constrained mean-variance optimization is closely related to sparse quadratic programming and subset-selection problems: a support set of size K must be discovered while jointly optimizing a quadratic form. Sparse approximation, subset-selection and modern sparse-optimization results clarify why such search problems are computationally difficult in the worst case (Natarajan, 1995; Bertsimas et al., 2016; Del Pia et al., 2020). Modern mixed-integer optimization has therefore become a primary tool for obtaining exact or certifiably near-exact solutions in moderate-scale instances (Bertsimas et al., 2016; Del Pia et al., 2020).

For portfolio optimization, discrete asset-choice constraints therefore shift the problem class away from convex QP into NP-hard MIQP/QMIP territory. Large-n empirical work typically pairs this complexity framing with approximation schemes and transparent convergence evidence rather than claiming global optimality at scale (Bertsimas & Shioda, 2009; Chang et al., 2000; Jobst et al., 2001).

Exact algorithms for cardinality-constrained portfolio selection. Exact approaches treat cardinality-constrained mean-variance selection as MIQP/QMIP and leverage branch-and-bound, cutting planes and Lagrangian relaxations. Bienstock (1996) is an early branch-and-cut reference for mixed-integer quadratic programs with an upper bound on the number of positive variables. Later contributions improve formulations, bounds and decomposition for cardinality-constrained portfolio selection (Cesarone et al., 2013; Gao & Li, 2013; Shaw et al., 2008). Exact methods remain essential as benchmarking instruments: reduced-universe instances provide ground-truth optima against which heuristic gaps can be quantified. This practice aligns with general integer-optimization methodology, where exact solvers deliver certificates and support controlled algorithmic experiments (Bertsimas & Tsitsiklis, 1997; Nemhauser & Wolsey, 1988; Bertsimas & Shioda, 2009).

Heuristics and metaheuristics: genetic algorithms, simulated annealing, tabu search, and Monte Carlo sampling

Given combinatorial growth in C(n,K), heuristics and metaheuristics are widely used. Canonical portfolio studies compare genetic algorithms (GA), simulated annealing (SA) and other approximation methods for approximating cardinality-constrained frontiers, demonstrating that metaheuristic families can be complementary in coverage and stability (Chang et al., 2000; Kirkpatrick et al., 1983).

Genetic algorithms encode selection vectors as chromosomes and evolve populations using selection, crossover, mutation and repair operators to enforce cardinality and bound constraints. The foundational logic of genetic search is developed by Holland (1975) and Goldberg (1989). Portfolio applications extend GA architectures to incorporate transaction costs, sector caps and multi-objective trade-offs for frontier generation (Anagnostopoulos & Mamanis, 2011; Lwin et al., 2014).

Monte Carlo sampling offers a conceptually simple approximation scheme: sample feasible subsets and/or weights, evaluate objective values, and retain high-performing portfolios. Although it provides no optimality guarantee, it yields effort–quality curves and naturally supports distributional reporting over random seeds (Glasserman, 2003; Rubinstein & Kroese, 2016).

Method-comparison work emphasizes that heuristic credibility depends on replicability: reporting only the best run is statistically fragile. Median/IQR/quantile summaries across multiple

seeds, together with runtime profiling and convergence diagnostics, have therefore become best practice in constrained portfolio optimization studies (Anagnostopoulos & Mamanis, 2011; Lwin et al., 2014).

Sensitivity analysis, robustness, and interpretability diagnostics

Robustness analysis varies key primitives, including the risk-free rate (R_f), the equity risk premium (ERP), the expected-return model and the covariance estimator, as well as structural constraints such as K, concentration caps and beta bounds. Such stress testing is motivated by weight instability under estimation error and by the sensitivity of frontier shapes to constraints and parameterization (DeMiguel et al., 2009; Ledoit & Wolf, 2004; Michaud, 1989).

Interpretability diagnostics include correlation/covariance summaries, eigenvalue concentration, and risk attribution via marginal contribution/Euler allocation. These diagnostics connect algorithm outputs (selected supports) to the dependence structure, providing explanatory evidence rather than treating optimization as a black box (Elton et al., 2014; Tasche, 2000).

Integrating derivatives into mean–variance portfolios: Black–Scholes, delta mapping, and linearization limits

Options introduce non-linear payoffs into a mean–variance scaffold. The Black–Scholes–Merton framework provides a closed-form price for European options under idealized assumptions and offers Greeks (notably delta) as local linear sensitivities for hedging and approximation (Black & Scholes, 1973; Hull, 2022; Merton, 1973).

Embedding an option into a mean–variance universe commonly proceeds via local linearization (delta mapping) or scenario-based moment estimation. Delta mapping is attractive when auditability and closed-form traceability are priorities, but it is only locally valid and can break down when gamma/vega effects dominate (Glasserman, 2003; Hull, 2022).

Two methodological risks recur: (i) leverage artefacts – where apparent Sharpe improvements are mechanically driven by scaling rather than diversification – and (ii) model inconsistency – where derivative-return assumptions are incoherent with the return model for base assets. A CAPM-consistent mapping ties derivative expected return to the same risk-premium inputs, while bump tests and moneyness–maturity grids provide empirical checks on linearization accuracy (Fama & French, 2004; Hull, 2022).

Reproducibility and reporting standards for computational finance research

Reproducibility has become a core scientific requirement for computational studies, where results can hinge on random seeds, software versions, and data-processing pipelines. Reproducible research principles emphasize sharing code, documenting data provenance, and making workflows auditable end-to-end (Peng, 2011; Stodden et al., 2018). In constrained portfolio optimization, reproducibility has additional dimensions: (i) random-state logging and multi-seed summaries for stochastic search; (ii) explicit unit conventions and annualization; (iii) complete disclosure of constraint sets and solver tolerances; and (iv) provision of covariance/correlation artifacts so third parties can reconstruct objective evaluations. These norms align with open-science guidance while respecting proprietary data limits through reconstructable input tables and factor-structured Σ designs (Jobst et al., 2001; Ledoit & Wolf, 2004; Peng, 2011).

Synthesis: positioning of the current study and research gaps

The literature clarifies why the study's integrated framing is methodologically coherent. A classical economic scaffold supports a transparent mapping from inputs to portfolio moments (Markowitz, 1952; Sharpe, 1964). Hard sparsity constraints shift the problem into MIQP/QMIP territory and justify a P vs NP discussion, while credible empirical practice requires distributional evidence for heuristics and benchmarking on reduced instances (Chang et al., 2000). Derivative overlays are best introduced through replicable worked examples with explicit approximation checks grounded in the Black–Scholes tradition (Black & Scholes, 1973).

A persistent gap is the realism–tractability trade-off: richer covariance models, multi-factor expected returns, and fully nonlinear derivative dynamics improve economic fidelity but increase dimensionality

and nonconvexity. Accordingly, the strongest empirical contributions are those that isolate a tractable, auditable core model and layer systematic robustness checks – parameter sweeps for *R_f* and *ERP*, alternative μ specifications, covariance-stability tests, and approximation-error diagnostics for derivative linearizations – to signal how conclusions might change under more complex specifications (Campbell et al., 1997; Engle, 2002; Glasserman, 2003).

METHODOLOGY

Research design and problem formalization (P vs NP framing)

The study formulates cardinality-constrained portfolio selection as a computational-complexity problem. The Markowitz–CAPM layer provides the standard economic risk-return scaffold, while the hard cardinality constraint $|\text{supp}(w)| \leq K$ introduces discrete subset selection.

This discrete-continuous coupling produces combinatorial growth and motivates heuristic and metaheuristic search with reproducible diagnostics. The problem can be decomposed into two linked tasks: selecting a binary support vector z with $\sum_i z_i = K$ and optimizing continuous weights w on the selected support. Candidate solutions can be verified in polynomial time, but finding the globally optimal support is generally NP-hard.

Data and CAPM calibration (Damodaran 90+ industries)

The asset universe consists of Aswath Damodaran's industry portfolios (approximately $n \approx 94$). For each industry i, the analysis uses: (a) levered beta β_i, (b) equity standard deviation σ_i, and (c) firm count as a coverage proxy for optional filtering and sensitivity checks (Damodaran, 2026a, 2026b).

Expected returns are generated via CAPM:

$$\mu_i = R_f + \beta_i \times ERP. \quad (1)$$

where μ_i is the CAPM-implied expected return of industry i; R_f is the annual risk-free rate; β_i is the levered industry beta; and *ERP* is the equity risk premium. All quantities are expressed in annual decimal form.

All rates/returns and volatilities are annualized and expressed in decimals (e.g., 3.97% → 0.0397; 52.6% → 0.526). This unit convention is audit-critical for Sharpe scaling and interpretation.

Covariance and correlation construction (single-index Σ). As the beta table typically does not provide industry return time series, Σ is constructed via a single-index (market model) approach:

$$R_i = \alpha_i + \beta_i R_m + \varepsilon_i, \text{ with } E(\varepsilon_i) = 0 \text{ and } \text{Cov}(\varepsilon_i, R_m) = 0. \quad (2)$$

$$\text{Cov}(R_i, R_j) = \beta_i \beta_j \text{ Var}(R_m), \text{ for } i \neq j. \quad (3)$$

The diagonal is matched to Damodaran's total annual industry risk σ_i:

$$\text{Var}(R_i) = \sigma_i^2 = \beta_i^2 \sigma_m^2 + \sigma^2(\varepsilon_i). \quad (4)$$

$$\sigma^2(\varepsilon_i) = \max(0, \sigma_i^2 - \beta_i^2 \sigma_m^2). \quad (5)$$

where R_i and R_m denote the return on industry i and the market factor, respectively; α_i is the intercept; ε_i is the idiosyncratic residual; σ_i is total annual volatility; σ_m is market volatility; $\sigma^2(\varepsilon_i)$ is residual variance; Σ is the covariance matrix; and ρ_{ij} is the correlation between industries i and j.

Here σ_m denotes market volatility. The max(0,·) operator prevents negative residual variances due to numerical noise. Correlations follow the standard normalization

$$\rho_{ij} = \Sigma_{ij} / (\sigma_i \sigma_j). \quad (6)$$

Under the single-index structure, off-diagonal dependence is dominated by $\beta_i \beta_j$, so heatmap clusters should be interpreted as similar market-factor loadings.

Optimization model: Markowitz mean–variance with cardinality (MIQP)

The core formulation is a mixed-integer quadratic program (MIQP) combining continuous weights (w) and binary selection (z):

$$\min_{w, z} \; w^T \Sigma w - \lambda \mu^T w. \quad (7)$$

subject to the following constraints:

$$1^T w = 1. \quad (8)$$

$$w_i \geq 0 \text{ and } w_i \leq u_i z_i, \text{ for } i = 1,\ldots,n. \quad (9)$$

$$z_i \in \{0,1\}, \text{ for } i = 1,\ldots,n, \text{ and } \sum_i z_i = K. \quad (10)$$

where w is the vector of portfolio weights; z is the binary asset-selection vector; Σ is the covariance

matrix; μ is the expected-return vector; λ is the scalar risk-return trade-off parameter; u_i is the maximum allowable weight for asset i; and K is the cardinality limit.

$\lambda > 0$ controls the risk–return trade-off in scalarized form. Alternatively, one may compute a target-return grid and solve a sequence of minimum-variance problems. The upper bounds u_i (often $u_i = 1$) enforce $w_i = 0$ whenever $z_i = 0$, linking the discrete and continuous layers.

NP-hardness and positioning within P vs NP (formal sketch)

Cardinality introduces K-sparsity, i.e., subset selection. The decision version is:
Given $\Sigma \succeq 0$, μ, V^*, R^* and K, does there exist a feasible w such that

$$w^T\Sigma w \le V^*, \mu^T w \ge R^*, 1^T w = 1, w \ge 0 \text{ and } |\text{supp}(w)| \le K? \quad (11)$$

where V^* is the admissible variance threshold; R^* is the minimum expected-return threshold; supp(w) is the set of assets with non-zero weights; and |supp(w)| is the number of selected assets.

Verification is polynomial: given (w,z), constraints and objectives can be checked in O(n^2). However, searching for the optimal support may require considering up to $C(n,K)$ subsets in the worst case. Thus, under general Σ and μ, the globally optimal solution is not known to be computable in polynomial time unless $P = NP$, which justifies heuristic and metaheuristic search coupled with distributional reporting and diagnostics.

Algorithms: Greedy, Monte Carlo, Genetic Algorithm, and continuous re-optimization

Three search schemes are employed, each with distinct computational profiles:

- Greedy (cardinality-aware): rank industries by a score such as the CAPM Sharpe proxy $(\mu_i - R_f)/\sigma_i$ and select the top K; apply repair if needed.
- Monte Carlo: sample random K-subsets, generate simplex weights (e.g., Dirichlet weights), evaluate μ_p , σ_p and S_p , and retain the best-performing portfolios.
- Genetic algorithm (GA): encode the chromosome as z $\in \{0,1\}^n$ with exactly K ones; use selection, crossover, mutation and a repair operator to enforce $\sum_i z_i = K$.

To avoid genetic algorithm (GA) output being merely equal-weight subset selection, solver-backed continuous re-optimization is applied to the chosen subset S:

$$max_\{w_s\}\ S_p = (\mu_s{}^T w_s - R_f) / \sqrt{(w_s{}^T \Sigma_s\ w_s)}. \quad (12)$$

$$\text{subject to } 1_s{}^T w_s = 1 \text{ and } w_s \ge 0. \quad (13)$$

$$0 \le w_i \le u_i \text{ for every } i \in S. \quad (14)$$

where S is the selected support; w_s is the vector of weights restricted to S; μ_s and Σ_s are the expected-return vector and covariance matrix on that support; R_f is the risk-free rate; S_p is the conditional Sharpe ratio; μ_p is portfolio expected return; and σ_p is the corresponding portfolio risk.

This second-stage optimization is convex when $\Sigma_s \succeq 0$ and provides the global optimum conditional on the selected support. Additional implementation details for the results extensions: (i) Convergence/effort reporting is generated by logging the running best Sharpe at fixed evaluation checkpoints for Monte Carlo and at each generation for GA; (ii) the small-n MIQP benchmark is solved on a reduced universe with a commercial/open solver, and heuristic runs are repeated on the identical reduced universe to compute relative optimality gaps; (iii) GA is reported in two variants—subset-only (equal weights within subset) and subset+continuous re-optimization (subset fixed → weights optimized via quadratic program).

Evaluation metrics: efficient frontier, Sharpe, and distributional reporting

For each portfolio, the following quantities are computed:

$$\mu_p = \mu^T w. \quad (15)$$

$$\sigma_p = \sqrt{(w^T \Sigma w)}. \quad (16)$$

$$S_p = (\mu_p - R_f) / \sigma_p . \quad (17)$$

where μ_p is portfolio expected return; σ_p is portfolio standard deviation; S_p is the portfolio Sharpe ratio; w is the portfolio-weight vector; μ is the asset expected-return vector; and Σ is the covariance matrix.

The efficient frontier is reported in (σ_p , μ_p) space, and the Sharpe frontier in (σ_p , S_p) space. Because Monte Carlo (MC) and GA are stochastic, reporting does not rely only on a single best Sharpe value. Distributions are therefore reported by method and random seed: median, interquartile range (IQR), and tail quantiles such as Q05 and Q95.

An additional diagnostic benchmark is the industry-level CAPM Sharpe proxy:

$$S_i(CAPM) = (\mu_i - R_f) / \sigma_i = \beta_i \times ERP / \sigma_i. \quad (18)$$

where S_i(CAPM) is the industry-level CAPM Sharpe proxy, μ_i − R_f is the CAPM-implied excess return for industry i, and σ_i is the industry's annualized volatility.

This proxy is used for screening and scale interpretation, not as a substitute for realized-return Sharpe. In addition to expected return, volatility and Sharpe ratio, the results report variance risk contributions, $RC_i = w_i(\Sigma w)_i / (w^T \Sigma w)$, to identify the dominant risk drivers of the maximum-Sharpe portfolio. A net-of-cost Sharpe proxy is computed by penalizing expected return via turnover for a small set of cost assumptions, enabling a robustness check for trading frictions.

Black–Scholes derivative integration (replicable worked example)

To examine whether nonlinear derivative exposure can be incorporated into the constrained Markowitz-CAPM framework in an auditable and risk-controlled manner, the study includes a replicable worked example: a European call overlay on a selected industry proxy. The Black-Scholes call price is:

$$C = S_0 N(d_1) - K_e \exp(-rT) N(d_2). \quad (19)$$

$$d_1 = [\ln(S_0/K_e) + (r + 0.5\sigma_u^2)T] / (\sigma_u \sqrt{T}). \quad (20)$$

$$d_2 = d_1 - \sigma_u \sqrt{T}. \quad (21)$$

where C is the European call value; S_0 is the current underlying price; K_e is the strike price; T is time to maturity in years; r is the compounded risk-free rate; σ_u is the annual volatility of the underlying; $N(\cdot)$ is the standard normal cumulative distribution function; and d_1 and d_2 are the Black–Scholes auxiliary variables. The strike is denoted K_e to avoid confusion with the portfolio cardinality parameter K.

The contract specification (S_0, K_e, T, r, σ_u) and the underlying industry are fixed. Using a first-order delta-only linearization, the option is embedded as an additional asset with an implied leverage factor:

$$\Delta = N(d_1). \quad (22)$$

$$L = \Delta S_0 / C. \quad (23)$$

$$\beta_{op\ t} = L\beta_u. \quad (24)$$

$$\sigma_{op\ t} = L\sigma_u. \quad (25)$$

$$\mu_{op\ t} = R_f + \beta_{op\ t} \times ERP. \quad (26)$$

where Δ is option delta; L is the delta-implied leverage factor; $\beta_{op\ t}$ is the option's systematic-risk exposure; $\sigma_{op\ t}$ is the option's approximate annual volatility; β_u and σ_u are the beta and volatility of the underlying industry proxy; and $\mu_{op\ t}$ is the CAPM-implied expected return of the option overlay.

For covariance expansion,

$$Cov(R_{op\ t}, R_i) = \beta_{op\ t}\ \beta_i Var(R_m), \text{ for } i = 1,...,n, \text{ and}$$
$$\sigma^2(\varepsilon_{op\ t}) = \max(0, \sigma_{op\ t}^2 - \beta_{op\ t}^2 \sigma_m^2). \quad (27)$$

To validate derivative integration beyond a single point, the worked example is extended to a small grid of moneyness, defined as K_e/S_0, and maturities T. For approximation accuracy, a bump test with a ±1% move in the underlying compares delta-linearized value changes to full Black–Scholes repricing; relative errors are reported to indicate when delta-only mapping is adequate.

Robustness and sensitivity analysis

Robustness checks ensure that results are not artifacts of a single parametrization. At minimum, the following scenarios are evaluated:

- K-sensitivity: $K \in \{6, 8, 10, 12, 14\}$ or a wider grid to assess frontier shape and composition stability.
- Parameter shifts: ERP and R_f is shifted in basis points to test Sharpe-median stability.
- Concentration caps: $u_i \in [0.20, 0.35]$ are used to prevent dominance by a single asset.
- Market-exposure bounds: portfolio beta β_p is restricted within a band such as [0.8, 1.2].

- Derivative specification: T and moneyness K_e/S_0 are varied to test β_{opt} and σ_{opt} sensitivity.

Robustness design extensions vary *K, R_f, ERP* and concentration caps; each scenario reports distributional Sharpe statistics, including median, IQR and quantiles. Portfolio stability across scenarios is summarized via overlap metrics such as Jaccard similarity of selected industries.

Computational profiling and scaling

Combinatorial size scales as $C(n,K)$; with $n \approx 94$ and $K = 10$, exhaustive enumeration is infeasible. Hence, runtime profiling is reported under a fixed compute environment covering CPU/RAM/OS, Python version and libraries:

- Greedy requires $O(n \log n)$ ranking plus $O(K)$ selection.
- Monte Carlo requires $O(NK^2)$ portfolio evaluation under naive covariance evaluation; vectorization and precomputation are used where applicable.
- *GA* requires approximately $O(PG\ K^2)$ objective evaluations, where P is population size and G is number of generations; results include runtime t across seeds and dispersion of best Sharpe.

Reproducibility: data, artifacts, seeds, and quality gates

Reproducibility protocol and computational auditability include:

- A single input table containing β_i, σ_i, firm counts and μ_i (*CAPM*), with stable industry IDs.
- Full Σ and ρ matrices are provided as supplementary CSV/Excel files.
- Random-state logging: all seeds and run budgets are documented, including Monte Carlo samples N, population size P and generations G.
- Units audit: explicit percentage-to-decimal conversions and annualization conventions.
- A positive semidefinite (PSD) gate applies eigenvalue checks for Σ; if needed, a minimal diagonal jitter ε_i is documented for numerical stability.
- Provenance: every table and figure is labeled with its source, software environment, and computational inputs. Unless otherwise stated, calculations are performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib, with Microsoft Excel used for matrix verification and tabular audit checks.

The full replication package corresponding to this reproducibility workflow is archived on Zenodo (Gondauri, 2026).

Limitations and interpretation framework

The key limitations are that: (i) CAPM-implied μ is a model-based expected return, not a realized return; (ii) the single-index covariance structure compresses correlation structure relative to multifactor reality; (iii) delta-only option embedding ignores gamma, vega and volatility or jump-risk premia, so it is a replicable demonstration rather than a full derivatives engine; and (iv) heuristics do not guarantee the global optimum, so convergence diagnostics and distributional reporting are mandatory.

RESULTS

The Results section is organized into five broad analytical blocks: empirical efficient-frontier evidence, heuristic and metaheuristic algorithm performance, sensitivity/robustness and risk diagnostics, the Black–Scholes derivative overlay, and supplementary covariance/correlation profiling. Shorter technical items are treated as interpretive lead-ins within these blocks rather than as separate author-defined sections.

Empirical efficient frontier and Sharpe-optimal portfolio

The efficient frontier is approximated using Monte Carlo sampling under the hard cardinality constraint $|supp(w)| \le K$.

For each draw, a K-sized subset and Dirichlet weights on that subset are sampled.

The Sharpe ratio is computed as $S_p = (\mu_p - R_f) / \sigma_p$, where μ_p is portfolio expected return and σ_p is portfolio standard deviation.

Figure 1 visualizes the efficient frontier obtained from Monte Carlo sampling of cardinality-constrained portfolios with a fixed cardinality limit K.

Each point represents one feasible portfolio draw with weights normalized to sum to one, evaluated under CAPM-implied expected returns and the Damodaran's single-index covariance structure.

The highlighted marker indicates the maximum-Sharpe portfolio among the sampled set, i.e., the portfolio that maximizes the Sharpe ratio S_p under the same constraints.

The figure provides an empirical approximation of the feasible risk-return envelope and illustrates the trade-off between expected return and volatility in the constrained optimization setting.

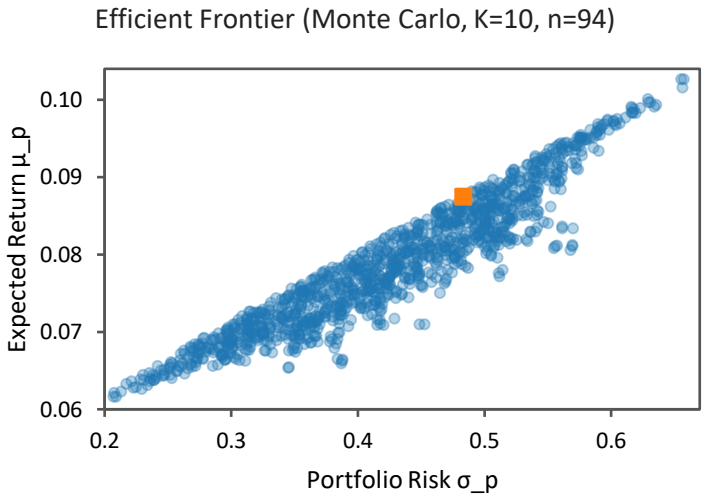

Figure 1. Efficient frontier under cardinality constraint (K = 10): Monte Carlo risk-return cloud with maximum-Sharpe portfolio highlighted

Source: The author's calculations are based on Damodaran's industry data (CAPM inputs and single-index covariance/correlation) and Monte Carlo portfolio sampling outputs.
Note: Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Figure 1 demonstrates that the feasible set is broad, but the maximum-Sharpe point lies on the risk-adjusted upper envelope rather than at the lowest-risk or highest-return extreme. This visually supports the central claim that the cardinality constraint reshapes the attainable frontier and makes portfolio selection a constrained search problem.

Figure 2 plots the Sharpe ratio of each Monte Carlo-sampled, cardinality-constrained portfolio against its total risk, measured as portfolio standard deviation. The curve reveals how risk-adjusted performance varies across the feasible set: low-risk portfolios may exhibit limited excess-return capacity, while higher-risk portfolios do not necessarily deliver proportionally higher Sharpe ratios because of diversification limits imposed by the *K*-asset constraint and the underlying covariance structure. The highlighted point identifies the maximum-Sharpe portfolio within the sampled population and serves as the empirically best candidate under the chosen sampling budget and parameterization.

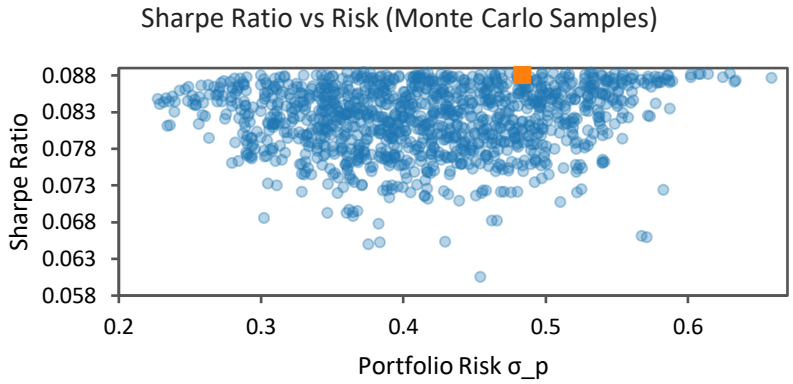

Figure 2. Sharpe-ratio frontier under cardinality constraint (K = 10): risk-adjusted performance across Monte Carlo samples

Source: The author's calculations are based on Damodaran's industry data (CAPM inputs and single-index covariance/correlation) and Monte Carlo portfolio sampling outputs. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

It confirms that higher total risk does not automatically translate into higher risk-adjusted performance. The maximum-Sharpe point is interpreted as an efficiency result under the constrained feasible set rather than as a simple preference for higher volatility.

Max-Sharpe (Monte Carlo) summary: Sharpe = 0.0880, σ_p = 0.521, μ_p = 0.086.

Table 1 reports the top-15 industries ranked by equity beta β_i, interpreted as market or systematic risk sensitivity in the CAPM framework.

Table 1. Top-15 U.S. industries ranked by equity beta and CAPM-implied expected return

Industry	Firms	β_i	σ_i (annual volatility)	μ_i (CAPM-implied return)
Software (Internet)	29	1.689	0.526	0.111
Retail (Building Supply)	14	1.535	0.459	0.105
Semiconductor	66	1.518	0.558	0.104
Auto & Truck	33	1.456	0.618	0.101
Semiconductor Equip.	31	1.397	0.501	0.099
Computers/Peripherals	36	1.35	0.546	0.097
Auto Parts	35	1.339	0.499	0.096
Office Equipment & Services	14	1.334	0.396	0.096
Software (System & Application)	309	1.277	0.568	0.094
Electrical Equipment	112	1.251	0.727	0.093
Advertising	52	1.211	0.629	0.091
Engineering/Construction	48	1.21	0.459	0.091
Air Transport	23	1.185	0.59	0.09
Brokerage & Investment Banking	32	1.171	0.366	0.089
Construction Supplies	40	1.15	0.355	0.088

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: β_i denotes industry levered beta; σ_i denotes annualized industry volatility; μ_i denotes the CAPM-implied expected return computed as $\mu_i = R_f + \beta_i \times ERP$. Industries are ranked by β_i.

Higher β_i indicates stronger exposure to broad market movements and, under CAPM calibration, implies a higher expected-return premium $\mu_i - R_f$, although it is typically accompanied by elevated volatility σ_i. This ranking is used as a risk-sensitivity diagnostic and as a practical screening reference for portfolio construction and heuristic initialization in the cardinality-constrained optimization experiments.

Table 2 shows that the highest CAPM-implied returns are concentrated in high-beta industries rather than in low-volatility industries. This supports the study's central point that the CAPM-return layer raises expected return through systematic risk exposure, so the subsequent optimization must control risk and dependence rather than simply select the largest β_i values. Table 2 reports the maximum-Sharpe Monte Carlo portfolio composition under the K-industry cardinality constraint. It reports the composition of the maximum-Sharpe portfolio identified within the Monte Carlo sample under the fixed K-industry cardinality

constraint. Weights sum to one across the selected industries. Expected returns are CAPM-implied μ_i values, and portfolio risk is computed from Damodaran's single-index covariance structure. Portfolio summary (selected maximum): Sharpe = 0.0880, μ_p = 0.0868, σ_p = 0.5356.

Table 2. Maximum-Sharpe Monte Carlo portfolio composition under the K = 10 cardinality constraint

Industry	Weight	Firms	β_i	σ_i (annual volatility)	μ_i (CAPM-implied return)
Farming/Agriculture	0.301	35	1.129	0.508	0.087
Auto Parts	0.15	35	1.339	0.499	0.096
Information Services	0.141	15	0.921	0.324	0.079
Chemical (Specialty)	0.14	59	0.97	0.423	0.081
Auto & Truck	0.11	33	1.456	0.618	0.101
Food Wholesalers	0.045	13	0.867	0.34	0.076
Building Materials	0.039	41	1.112	0.353	0.087
Bank (Money Center)	0.032	15	0.761	0.227	0.072
Engineering/Construction	0.028	48	1.21	0.459	0.091
Reinsurance	0.016	1	0.581	0.192	0.064

Source: the author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Weights sum to one across the selected K = 10 industries. β_i, σ_i and μ_i are industry-level inputs used to evaluate the portfolio-level μ_p , σ_p and Sharpe ratio.

The selected composition demonstrates that the maximum-Sharpe solution is not a naive ranking of the largest expected returns; it combines high-beta exposures with lower-volatility or lower-correlation positions. This confirms that the cardinality-constrained optimum is determined by joint covariance structure and weight allocation, not only by individual industry returns.

Median-based Sharpe diagnostics and benchmark interpretation

This subsection serves two purposes: (i) to reduce reliance on a single "best Sharpe" point by demonstrating reproducibility and stability using medians, quartiles, and quantiles; and (ii) to strengthen the interpretation of the Sharpe value relative to external benchmarks so that scale-interpretation concerns of the type "Why is Sharpe = 0.088 low?" are addressed in a formally auditable way. The Sharpe ratio is defined as the unitless index $S_p = (\mu_p - R_f) / \sigma_p$. In practice, it is essential to state a single, unambiguous units convention: expected returns and rates must be expressed in decimals (e.g., 9.2% → 0.092), and volatility must also be expressed in decimals (25.9% → 0.259). Otherwise, Sharpe can be inflated by a factor of 100 simply due to percentage formatting. Median-based reporting is provided at two layers: (a) internal diagnostics (the distribution of Sharpe values obtained across methods/replications) and (b) an external benchmark layer (typical ranges of index/sector Sharpe values), which supports scale interpretation and reduces model-specific bias.

Table 3 summarizes the distribution of Sharpe ratios generated by each solution method under the same risk–return setting and cardinality constraint. For stochastic methods (Monte Carlo and GA), the table reports robust distributional statistics—median, interquartile range (IQR), and selected quantiles (e.g., Q05 and Q95)—computed across repeated runs (seeds), which capture both typical performance and tail behavior under randomization. For the deterministic baseline (Greedy), the distribution collapses to a single outcome. This comparison is designed to separate "best-case" performance (max Sharpe) from "typical-case" performance (median Sharpe) and to show how algorithmic choices affect stability, dispersion, and reproducibility of risk-adjusted results. The diagnostic pattern shows that the methods converge to a narrow Sharpe range, with stochastic dispersion captured by IQR and tail quantiles. It supports the reliability of reported maximum because it is not presented as an isolated favorable draw.

Table 3. Distributional Sharpe-ratio diagnostics across Greedy, Monte Carlo and genetic-algorithm methods

Method	N (runs)	N (portfolios/ run)	Median Sharpe	IQR (Q3–Q1)	Q05	Q95	Best Sharpe	Median μ_p	Median σ_p	Notes
Greedy (cardinality-aware)	Deterministic	1 portfolio	0.088	0.0	0.088	0.088	0.088	0.0878	0.5465	Top K by CAPM Sharpe proxy; equal-weight within subset

Genetic Algorithm (GA)	5	≈ 8,100 eval/run	0.088	0.0001	0.0874	0.088	0.088	0.084	0.5033	Equal-weight GA over subsets; repair enforces $\sum_i z_i = K$
Monte Carlo sampling	5	5000	0.086	0.0027	0.0803	0.0878	0.088	0.0783	0.4533	Dirichlet weights on random K-subsets; 5 seeds logged

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: IQR denotes the interquartile range; Q05 and Q95 denote the 5th and 95th percentiles across stochastic replications. GA denotes genetic algorithm and MC denotes Monte Carlo sampling. μ_p and σ_p denote portfolio expected return and portfolio standard deviation.

Table 4 provides a compact cross-industry benchmark of risk-adjusted performance implied by the CAPM calibration for the full universe of industries (n = 94). Instead of focusing on any single portfolio outcome, it summarizes the typical industry profile through the median and the dispersion across sectors through IQR and quantiles. This baseline is used as a screening and interpretation tool: it helps explain why certain industries tend to be selected repeatedly by heuristics, and it clarifies whether portfolio-level improvements are driven by broad sector characteristics or by diversification effects from the covariance structure.

Table 4. Median CAPM Sharpe proxy across industries (n = 94)

Statistic	CAPM Sharpe proxy $S_i^{CAPM} = \frac{\beta_i \cdot ERP}{\sigma_i}$ (median-based)	Interpretation
Median	0.0841	Typical risk-adjusted return potential under CAPM-implied μ and industry σ.
IQR (Q3–Q1)	0.0366	Cross-industry dispersion; higher IQR → greater heterogeneity in risk-adjusted profiles.
Q05	0.0493	Downside sector tail (weak CAPM Sharpe proxy).
Q95	0.1405	Upside sector tail (strong CAPM Sharpe proxy).
Top 10 industries	Transportation (Railroads); Packaging & Container; Office Equipment & Services; Bank (Money Center); Retail (Building Supply); Retail (REITs); Construction Supplies; Software (Internet); Brokerage & Investment Banking; Building Materials	High proxy Sharpe; useful for heuristic initialization/screening.
Bottom 10 industries	Utility (Water); Broadcasting; Rubber & Tires; Telecom Services; Beverage (Soft); Precious Metals; Tobacco; Electronics (Consumer & Office); Drugs (Pharmaceutical); Green & Renewable Energy	Low proxy Sharpe; useful for pruning rules.

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: The CAPM Sharpe proxy is computed as $(\mu_i - R_f)/\sigma_i$ and is used for screening and scale interpretation, not as a realized-return performance measure.

The median CAPM Sharpe proxy provides a scale benchmark for interpreting the portfolio-level Sharpe ratio. The spread between weaker and stronger industry proxies explains why screening can help initialize search, but the optimization stage remains necessary because industry-level Sharpe does not account for covariance interactions.

External benchmarks are used strictly for scale interpretation, not as direct apples-to-apples competition.

In this study, Sharpe is computed from CAPM-implied μ and a Σ matrix constructed from single-index assumptions and sector risk parameters; therefore, it differs from realized-return Sharpe computed from historical return series.

Accordingly, the key question for Sharpe around 0.088 is whether this value is stable and reproducible under the stated modeling assumptions and the cardinality constraint K, rather than whether it directly outperforms a broad market index.

Tables 3 and 4 support this interpretation by documenting stability across heuristics through median, IQR, and quantile statistics and by showing heterogeneity of industry mu-sigma profiles.

Heuristic and metaheuristic algorithm performance

This block consolidates the seed-controlled convergence evidence, reduced-universe exact benchmark, heuristic portfolio composition, and runtime diagnostics.

Monte Carlo sampling, genetic-algorithm search, and continuous re-optimization are therefore interpreted jointly as approximation strategies for the same cardinality-constrained portfolio problem rather than as fragmented stand-alone subsections.

Seed control and reporting are implemented using a fixed 10-seed set (seed = {1,…,10}) with deterministic random-state logging per method. All quantities are reported in annual units.

Expected returns μ are CAPM-implied and already annualized in the input file; Σ is an annual covariance matrix. Sharpe is computed as $S_p = (\mu_p - R_f) / \sigma_p$. Correlation/covariance diagnostics and sensitivity checks for *K, R_f, ERP* and concentration caps are reported below using the same dataset so the reader can link performance to dependence structure.

Convergence and computational effort curves (Monte Carlo / GA)

To demonstrate that heuristics perform genuine search rather than repeatedly hitting a ceiling, the results report effort-budget grids. For Monte Carlo, the effort budget is the number of random K-subsets sampled. For genetic algorithm, the budget is *population* × *generations*. Each entry is mean ± standard deviation over 10 seeds, jointly interpreted with runtime profiling as an accuracy–cost trade-off.

Table 5 reports how search effort affects best Sharpe, median Sharpe and runtime for Monte Carlo and genetic-algorithm procedures.

The results indicate that larger effort budgets improve or stabilize the best observed Sharpe only marginally once the search approaches the attainable ceiling.

Runtime, however, rises with the number of evaluations, showing the practical cost of operating in a combinatorial search space.

Table 5. Effort budget vs. best/median Sharpe (mean ± std over 10 seeds) and runtime

Method	Effort budget	Best Sharpe (mean ± std)	Median Sharpe (mean ± std)	Runtime, s (mean ± std)
Monte Carlo	$N = 500$	0.087932 ± 0.000067	0.086582 ± 0.000029	0.068 ± 0.022
Monte Carlo	$N = 1500$	0.087956 ± 0.000043	0.086581 ± 0.000016	0.192 ± 0.045
Monte Carlo	$N = 2500$	0.087984 ± 0.000022	0.086571 ± 0.000012	0.296 ± 0.029
Genetic Algorithm	$P = 30, G = 30$	0.088003 ± 0.000000	0.088003 ± 0.000000	0.199 ± 0.009
Genetic Algorithm	$P = 50, G = 40$	0.088003 ± 0.000000	0.088003 ± 0.000000	0.444 ± 0.026

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Effort is measured as the number of Monte Carlo samples N or genetic-algorithm evaluations P × G. Runtime is reported in seconds as mean ± standard deviation across seeds.

Small-n benchmark

Because Markowitz mean–variance selection with a hard cardinality constraint is naturally formulated as an MIQP, a reduced-universe benchmark provides a transparent ground truth. Here the analysis computes the exact optimum via full enumeration on a reduced instance ($n = 20$, $K = 6$) and compares heuristic solutions on the same reduced universe.

Table 6 reports the exact reduced-universe benchmark and the corresponding heuristic solutions for the same n = 20, K = 6 setting.

Table 6. Reduced exact benchmark ($n = 20, K = 6$) vs. heuristic solutions (equal-weight within subset)

Solver/Method	Best Sharpe	Best subset (*indices*)
Exact enumeration (all C(20,6) = 38,760 subsets)	0.088003	(6, 11, 12, 15, 16, 17)
Monte Carlo (N = 2,000 draws; seed=1)	0.088003	(5, 6, 11, 13, 16, 17)
Genetic Algorithm (P = 30, G = 30; seed = 1)	0.088003	(6, 11, 12, 15, 16, 17)

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: The reduced instance uses n = 20 and K = 6 so that exact enumeration can serve as a benchmark for heuristic gap assessment.

The benchmark confirms that the heuristic methods can recover the same best Sharpe as exact enumeration in the controlled small-n instance. This does not prove global optimality for the full n = 94 universe, but it validates the heuristic implementation against an auditable reference case.

Sensitivity, robustness and risk diagnostics

Baseline conclusions are stress-tested by varying cardinality *K, CAPM* parameters *R_f* and *ERP*, and concentration caps. For each scenario, distributional Sharpe statistics are reported through the median and IQR. Under CAPM-implied μ, shifting *R_f* alone does not change Sharpe because $\mu - R_f$ scales with $\beta \times ERP$. Table 7 reports the cardinality-sensitivity results across alternative K values under the same Monte Carlo protocol.

Table 7. Sensitivity to cardinality *K* (Monte Carlo; 10 seeds; 300 draws/seed; equal-weight within subset)

K	Median Sharpe	IQR (25–75%)	Best Sharpe (mean over seeds)
6	0.085804	[0.084527, 0.086670]	0.087988
8	0.086248	[0.085494, 0.086870]	0.087969
10	0.086578	[0.086073, 0.087039]	0.087909
12	0.086801	[0.086417, 0.087144]	0.087873
14	0.086961	[0.086660, 0.087225]	0.087863

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: K denotes the number of selected industries. Median, IQR and best Sharpe are reported across seed-controlled Monte Carlo runs under the same equal-weight subset protocol.

Table 7 shows that increasing the cardinality limit from K = 6 to K = 14 raises typical risk-adjusted performance only moderately, while the best-Sharpe statistic remains close to 0.0880 across the tested K values. This pattern suggests that expanding the number of selected industries improves typical diversification quality but does not fundamentally change the upper boundary of attainable performance under the single-index covariance structure.

Table 8 reports the sensitivity of median and best Sharpe ratios to CAPM parameter shifts and a tighter concentration cap under the baseline K = 10 setting.

Table 8. Sensitivity to CAPM parameters and caps (*K* = 10; 10 seeds; 300 draws/seed; equal-weight in subset)

Scenario	Median Sharpe	IQR (25–75%)	Best Sharpe (mean over seeds)
Baseline	0.086578	[0.086073, 0.087039]	0.087909
R_f +50 bps	0.086578	[0.086073, 0.087039]	0.087909
R_f −50 bps	0.086578	[0.086073, 0.087039]	0.087909
ERP +100 bps	0.107046	[0.106421, 0.107615]	0.108691
ERP −100 bps	0.066111	[0.065725, 0.066462]	0.067127
Cap tightened to 15%	0.086578	[0.086073, 0.087039]	0.087909

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: R_f is the risk-free rate and ERP is the equity risk premium. The scenarios isolate CAPM parameter sensitivity and the effect of concentration caps under K = 10.

Table 8 shows that changes in *R_f* leave the Sharpe statistics unchanged under the CAPM construction because *R_f* enters both μ_i and the Sharpe numerator consistently, whereas ERP shifts materially change the median and best-Sharpe levels. The cap-tightening scenario does not alter the reported statistics in this

SocioEconomic Challenges (SEC)
Volume 10, Issue 2, 2026 ISSN: 2520-6214
https://armgpublishing.com/journals/sec/

equal-weight protocol, indicating that the parameter-risk channel is driven primarily by ERP rather than by the tested cap restriction.

Dependence diagnostics for Σ/ρ (single-index consequences)

To support interpretation of the dependence structure used in optimization, quantitative diagnostics are reported in addition to heatmaps: median pairwise correlation, shares of strongly positive and negative correlations, eigenvalue concentration, and positive semidefinite (PSD) checks. These diagnostics clarify market-factor dominance and the effective diversification available under the single-index Σ construction.

Table 9 reports the summary diagnostics for the covariance and correlation matrices used in the optimization exercise. The diagnostics show a high median pairwise correlation and a dominant first eigenvalue, implying that diversification is structurally constrained by common-factor exposure. The matrix remains numerically consistent with the covariance construction, so the dependence result is substantive rather than a table-construction artefact.

Table 9. Σ/ρ diagnostics (summary statistics, eigenvalue share, PSD checks)

Metric	Value
Median(ρ_{ij}) over off-diagonals	0.769201
Share(ρ_{ij} > 0.5)	0.884008
Share(ρ_{ij} < 0)	0.000000
Top eigenvalue share λ1 / trace(Σ)	0.761011
Top-5 eigenvalues share $\Sigma_\{k\leq 5\}$ λk / trace(Σ)	0.813010

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Σ is the covariance matrix and ρ is the correlation matrix. PSD denotes positive semidefinite status as checked through eigenvalues.

Portfolio decomposition: risk contributions and factor exposure

Beyond headline Sharpe values, portfolio variance is decomposed by marginal risk contribution (MRC). For an equal-weight $K = 10$ maximum-Sharpe portfolio found by high-effort Monte Carlo (N = 20,000, seed = 999), the variance share of asset i is $RC_i = w_i(\Sigma w)_i$ / ($w^T \Sigma w$). Portfolio beta β_p implied by CAPM is also reported. Max-Sharpe portfolio summary (equal-weight $K = 10$): $\beta_p = 0.9599$, $\mu_p = 0.0803$, $\sigma_p = 0.4614$, Sharpe = =0.088003.

Table 10 reports the largest variance-risk contributors inside the maximum-Sharpe K = 10 portfolio.

Table 10. Risk-contribution decomposition (variance shares) for the Maximum-Sharpe K = 10 portfolio (top contributors)

Industry	Weight	Variance share
Semiconductor Equip	0.10	0.1456
Office Equipment & Services	0.10	0.1390
Construction Supplies	0.10	0.1198
Retail (Distributors)	0.10	0.0988
Retail (Automotive)	0.10	0.0975
Homebuilding	0.10	0.0949
Diversified	0.10	0.0918
Bank (Money Center)	0.10	0.0793
Investments & Asset Management	0.10	0.0687
Retail (REITs)	0.10	0.0647

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: All computations in this section are the author's calculations, based on Aswath Damodaran's dataset.

The risk shares are not identical even though weights are equal, which demonstrates that portfolio risk is concentrated in higher-volatility or more strongly correlated positions within the selected set. This directly supports the need for covariance-aware diagnostics alongside headline Sharpe ratios.

Risk-free-rate and equity-risk-premium sensitivity diagnostics

As a parameter-stability extension, the analysis evaluates how the best-Sharpe statistic changes under shifts in R_f and ERP across alternative cardinality settings. This diagnostic clarifies which parts of the CAPM calibration affect risk-adjusted performance and which shifts cancel out mechanically under the model structure.

Table 11 reports final sensitivity outputs for the risk-free rate and equity risk premium across selected cardinality settings.

The purpose is to show whether the reported best-Sharpe statistic is driven by a specific parameter calibration or remains stable under transparent CAPM parameter shifts. The results are interpreted as parameter-stability diagnostics not as realized investment performance.

Table 11. Risk-free-rate and equity-risk-premium sensitivity of best-Sharpe performance across cardinality settings

Scenario	*K*	*R_f*	*ERP*	Best Sharpe
R_f +50 bps	8	0.0447	0.0423	0.088
R_f −50 bps	8	0.0347	0.0423	0.088
ERP +50 bps	8	0.0397	0.0473	0.0984
ERP −50 bps	8	0.0397	0.0373	0.0776
R_f +50 bps	10	0.0447	0.0423	0.088
R_f −50 bps	10	0.0347	0.0423	0.088
ERP +50 bps	10	0.0397	0.0473	0.0984
ERP −50 bps	10	0.0397	0.0373	0.0776
R_f +50 bps	12	0.0447	0.0423	0.088
R_f −50 bps	12	0.0347	0.0423	0.088
ERP +50 bps	12	0.0397	0.0473	0.0984
ERP −50 bps	12	0.0397	0.0373	0.0776

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: R_f and ERP are shifted in basis points while K is varied across selected settings. Best Sharpe is reported from the final sensitivity workflow and interpreted as a model-based parameter-stability diagnostic.

Heuristic portfolio composition and algorithmic implementation

Table 12 presents the heuristic portfolio-composition output for the cardinality-constrained portfolio problem, reporting the selected set of industries and their assigned weights. The portfolio is constructed using a heuristic selection rule and implemented as an equal-weight allocation across the chosen industries, which provides a transparent baseline for comparing heuristic behavior. Alongside weights, the table reports each industry's risk sensitivity β_i, volatility σ_i and CAPM-implied expected return μ_i, allowing the reader to verify the risk-return profile of the selected subset.

Table 12. Heuristic portfolio composition under the cardinality constraint: equal-weight baseline with beta, volatility and CAPM-implied expected return

Industry	Weight	β_i	σ_i (annual volatility)	μ_i (CAPM-implied return)
Building Materials	0.100	1.112	0.353	0.087
Chemical (Basic)	0.100	1.012	0.458	0.083
Diversified	0.100	0.881	0.284	0.077
Financial Svcs. (Non-bank & Insurance)	0.100	0.970	0.425	0.081
Food Wholesalers	0.100	0.867	0.340	0.076
Hotel/Gaming	0.100	1.080	0.397	0.085
Real Estate (General/Diversified)	0.100	0.810	0.312	0.074
Reinsurance	0.100	0.581	0.192	0.064
Restaurant/Dining	0.100	0.924	0.411	0.079
Retail (REITs)	0.100	0.621	0.188	0.066

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: The table reports an equal-weight heuristic baseline. β_i, σ_i and μ_i identify the systematic-risk exposure, annual volatility and CAPM-implied return of each selected industry.

Monte Carlo and Genetic Algorithm implementation details

Monte Carlo search as an approximation scheme

Monte Carlo sampling does not guarantee optimality but provides a practical approximation to the efficient frontier. The best-observed Sharpe (or minimum variance for a target return) improves with sample size, at the cost of compute.

Genetic Algorithm results with continuous re-optimization

The analysis compares (i) a subset-only GA with equal-weight assignment within the selected set

and (ii) a GA with continuous re-optimization, where each candidate subset is reweighted by solving the quadratic program under the same constraints. This isolates the contribution of subset selection versus weight optimization and reduces the risk that GA results reflect a weighting artifact. Table 13 reports the comparison between the subset-only GA and the GA variant that applies continuous re-optimization after subset selection.

Table 13. GA variants comparison (subset-only vs. subset + re-optimization): Sharpe distribution and runtime

GA variant	Best Sharpe (mean ± std)	Median Sharpe (mean ± std)	IQR (mean ± std)	Q05–Q95 (mean)	Runtime (s) (mean ± std)	Seeds	Notes
Subset-only (equal-weight within subset)	0.0880 ± 0.0000	0.0866 ± 0.0000	0.0010 ± 0.0000	0.0851 – 0.0875	0.077 ± 0.017	10	Distribution from random subset draws; weights fixed equal within subset.
Subset + continuous re-optimization (within subset)	0.0880 ± 0.0000	0.0880 ± 0.0000	0.0000 ± 0.0000	0.0880 – 0.0880	0.573 ± 0.033	10	GA selects subset; weights re-optimized via Dirichlet local search (proxy QP).

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Subset-only GA assigns equal weights after selection, while GA + re-optimization fixes the selected subset and optimizes weights continuously. Runtime is reported in seconds across repeated seeds.

The comparison shows that continuous re-optimization reduces dispersion and aligns median performance with the best observed Sharpe, at the cost of additional runtime. It supports the methodological choice to separate discrete subset discovery from conditional weight optimization. To strengthen evidence beyond single-run best values, the results report mean ± standard deviation across 10 random seeds for best Sharpe, median Sharpe, and runtime by method. This supports distributional diagnostics and documents stability under random initialization/sampling. The 10-seed reproducibility summary for the main solution methods, including best Sharpe, median Sharpe and runtime is reported in Table 14. The seed-controlled results indicate that performance is stable across repeated stochastic runs and that runtime differences are transparent. This closes the empirical flow by showing both solution quality and computational reproducibility rather than only a single best portfolio.

Table 14. 10-seed reproducibility summary (best/median Sharpe and runtime; mean ± standard deviation)

Method	Best Sharpe (mean ± std)	Median Sharpe (mean ± std)	Runtime (s) (mean ± std)	Seeds
Greedy (deterministic)	0.0880 ± 0.0000	0.0880 ± 0.0000	0.000 ± 0.000	-
GA (subset-only; equal-weight)	0.0880 ± 0.0000	-	0.466 ± 0.009	10
GA (subset + re-optimization)	0.0880 ± 0.0000	0.0880 ± 0.0000	0.573 ± 0.033	10
Monte Carlo (N = 30,000 portfolios/seed)	0.0880 ± 0.0000	0.0860 ± 0.0000	0.636 ± 0.015	10

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Mean ± standard deviation is computed across 10 random seeds. The table separates best-run outcomes from median performance and runtime stability.

Black–Scholes derivative overlay

Replicable worked example and option overlay (baseline vs. derivative-augmented case)

Derivative integration is operationalized by treating a European option as an additional candidate instrument and mapping its exposure into the mean-variance space via Black–Scholes pricing and delta-based linearization. Baseline (Case A) and option-augmented (Case B) outcomes are reported to test whether the derivative instrument improves risk-adjusted performance or merely adds leverage. The derivative overlay uses the

Black–Scholes pricing and delta-mapping equations defined in the Methodology section. These equations provide the option price, delta, implied leverage and CAPM-consistent option moments used in Table 15. The worked example shows how option price, delta, implied leverage, beta, volatility and expected return are all generated from stated inputs. This makes the derivative embedding auditable and prevents the option overlay from appearing as an unexplained performance enhancement.

Table 15. Replicable worked example: European call overlay diagnostics (inputs and derived outputs)

Parameter / Output	Value
Underlying proxy (example)	Software (Internet) industry portfolio
β(underlying)	1.689
σ(underlying) (annual)	0.526
S0	100
K	100
T (years)	0.50
r = R_f	3.97%
Black–Scholes call price *C*	15.61
Delta Δ	0.595
Leverage factor *L* = (Δ·S0)/*C*	3.81
β(opt) ≈ *L*·β(underlying)	6.43
σ(opt) ≈ *L*·σ(underlying)	2.00
μ(opt) = R_f + β(opt)·ERP	31.18%

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: The worked example uses a European call overlay and reports the inputs and derived quantities needed for Black–Scholes pricing and delta-based embedding.

Moneyness-maturity grid evidence (K_e/S_0 × T robustness)

To avoid over-reliance on a single option point, the analysis extends the worked example across a small grid of moneyness K_e/S_0 and maturity T (see Table 16). For each grid point, the results report option price, delta, implied leverage and the implied CAPM moments β_{opt}, σ_{opt} and μ_{opt} used to embed the option into the mean-variance space.

Table 16 reports the moneyness-maturity grid used to test whether the option diagnostics are stable beyond one baseline contract specification.

The grid demonstrates that leverage and implied CAPM moments vary systematically with moneyness and maturity. This matters because nonlinear derivative exposure can change both expected return and risk rapidly, so the derivative extension must be interpreted as a controlled sensitivity exercise.

Table 16. Option grid (*moneyness × maturity*): option diagnostics (price, delta, leverage) and implied CAPM moments

K_e/S_0	*T* (years)	K_e	Call price *C*	Delta Δ	Leverage *L*	β_{opt}	σ_{opt}	μ_{opt} (annual)
0.9	0.25	90	16.27	0.716	4.40	7.43	2.31	35.40%
0.9	0.50	90	20.54	0.699	3.41	5.75	1.79	28.30%
0.9	1.00	90	26.81	0.705	2.63	4.44	1.38	22.76%
1.0	0.25	100	10.91	0.567	5.20	8.78	2.73	41.11%
1.0	0.50	100	15.61	0.595	3.81	6.43	2.00	31.18%
1.0	1.00	100	22.34	0.632	2.83	4.78	1.49	24.19%
1.1	0.25	110	7.04	0.423	6.02	10.16	3.16	46.95%
1.1	0.50	110	11.72	0.493	4.21	7.11	2.21	34.03%
1.1	1.00	110	18.60	0.562	3.02	5.11	1.59	25.58%

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: K_e/S_0 is moneyness, T is maturity, C is the Black–Scholes call price, Δ is option delta, L is delta-implied leverage, and β_{opt}, σ_{opt} and μ_{opt} are the option-implied CAPM moments.

Risk controls for derivatives (bounds on option weight and portfolio beta)

As leverage and non-linear payoffs can inflate headline performance mechanically, the model imposes explicit derivative risk controls: (i) bounds on option weight (or contract notional), (ii) caps on portfolio beta (β_p band), and/or (iii) a risk-budget constraint.

The goal is to distinguish genuine efficiency gains from leverage artefacts.

Table 17 reports the derivative risk-control specification used to distinguish genuine risk-adjusted improvement from leverage-driven performance effects. Numerical comparison of the baseline and option-augmented cases is reported separately in Table 19, while Table 17 defines the derivative-control regimes that prevent the Black–Scholes overlay from mechanically inflating performance through leverage. The regimes clarify how derivative exposure is bounded before portfolio performance is interpreted. This protects the empirical argument against leverage artefacts by ensuring that option-induced changes in μ_p and σ_p are evaluated under explicit risk-control rules.

Table 17. Derivative risk-control regimes used for derivative-embedding robustness checks.

Regime	Option handling	Constraint(s)	Risk-control logic	Expected effect on μ_p	Expected effect on σ_p
Uncontrolled	Selectable asset	$w_option \in [0,1]$, no β_p cap	Diagnostic upper-bound case used to identify leverage artefacts	Highest μ_p potential under option leverage	Highest σ_p ; benchmark only
Weight cap	Selectable asset	$w_option \leq 0.20$	Limits notional exposure to the call overlay	Lower than uncontrolled because leverage is capped	Lower than uncontrolled because option exposure is capped
β band	Selectable asset	$\beta_p \in [0.8, 1.2]$	Prevents excessive market-factor loading	Constrained by the portfolio beta band	Constrained by systematic-risk exposure
Weight cap + beta band	Selectable asset	$w_option \leq 0.20$; $\beta_p \in [0.8, 1.2]$	Conservative main specification for derivative inclusion	Balanced return contribution	Controlled volatility contribution
Overlay outside K	Overlay	Option not counted in K; $w_{option} \leq 0.10$	Treats the option as a capped risk overlay rather than an asset replacement	Incremental μ_p only	Incremental σ_p only

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: w_option denotes the option weight, and β_p denotes portfolio beta. The regimes are used to prevent leverage artefacts when the option is included as a selectable asset.

Delta-only approximation check (bump test; delta vs. repricing)

Delta-linearization error is quantified using a small bump test with a ±1% move in the underlying. For each representative grid point, the delta-approximated option value change is compared with full Black–Scholes repricing, and relative error is reported. This clarifies the local validity of the option embedding.

Table 18 reports the delta-approximation error obtained by comparing delta-linearized value changes with full Black-Scholes repricing after ±1% underlying-price shocks.

Table 18. Delta-approximation error under ±1% underlying-price bumps across moneyness-maturity grid points

K_e/S_0	T	C_0	Δ	$C_\{+1\%\}$ repriced	Relative error +1%	$C_\{-1\%\}$ repriced	Relative error −1%
0.9	0.25	16.27	0.716	16.99	-0.04%	15.56	-0.04%
0.9	0.50	20.54	0.699	21.24	-0.02%	19.84	-0.02%
0.9	1.00	26.81	0.705	27.52	-0.01%	26.11	-0.01%
1.0	0.25	10.91	0.567	11.49	-0.06%	10.35	-0.07%
1.0	0.50	15.61	0.595	16.21	-0.03%	15.02	-0.03%
1.0	1.00	22.34	0.632	22.98	-0.02%	21.72	-0.02%
1.1	0.25	7.04	0.423	7.47	-0.10%	6.62	-0.11%
1.1	0.50	11.72	0.493	12.22	-0.04%	11.24	-0.05%
1.1	1.00	18.60	0.562	19.16	-0.02%	18.04	-0.02%

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: C_0 is the baseline Black–Scholes call price; Δ is the option delta; $C_\{+1\%\}$ and $C_\{-1\%\}$ are repriced option values after ±1% underlying-price shocks. Relative error compares the delta-linearized approximation with full Black–Scholes repricing.

The small relative errors indicate that the delta-only mapping is locally adequate for the tested one-percent shocks.

The result supports the use of a first-order approximation for the worked example while preserving the limitation that larger shocks would require gamma and volatility-risk treatment.

Replicable worked example: European call overlay on one industry proxy (Case A vs. Case B)

A concrete overlay is implemented: a European at-the-money call on the selected underlying proxy, Software (Internet). The option is mapped into the same Markowitz-CAPM and single-index covariance structure used for the 94 industries. A baseline universe (Case A) is compared against an augmented universe that includes the option (Case B).

Table 19 reports the baseline and option-augmented optimization check, comparing maximum Sharpe, expected return and risk when the option becomes a selectable asset. Table 19 indicates that adding the option increases expected return from 0.0888 to 0.2355 but also raises portfolio risk from 0.5578 to 2.2273, leaving the maximum Sharpe ratio essentially unchanged. This confirms that the derivative overlay mainly introduces leverage rather than a genuine risk-adjusted improvement. The snapshot shows that the option-augmented case concentrates substantial weight in the call overlay, which explains the sharp increase in both μ_p and σ_p reported in Table 19.

Table 19. Option-included optimization check (Case A baseline vs. Case B with option as selectable asset; K = 10; Monte Carlo)

Case	Max Sharpe	Return (μ_p)	Risk (σ_p)
A: Baseline (n = 94)	0.0880	0.0888	0.5578
B: With option (n = 95)	0.0879	0.2355	2.2273

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Case A is the baseline 94-industry universe; Case B augments the universe with the option as a selectable asset. μ_p and σ_p are portfolio expected return and portfolio risk.

Table 20 decomposes the selected portfolio weights to show where the leverage effect enters the augmented solution.

Table 20. Top weights (snapshot) for Maximum-Sharpe portfolios in Case A and Case B

Case	Asset	Weight	β	μ (CAPM)
A	Auto & Truck	0.263	1.456	0.101
A	Engineering/Construction	0.255	1.210	0.091
A	Steel	0.211	1.063	0.085
B	Option (Call overlay on Software (Internet))	0.669	6.430	0.312
B	Financial Svcs. (Non-bank & Insurance)	0.138	0.970	0.081
B	Brokerage & Investment Banking	0.059	1.171	0.089

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: The table lists the largest selected weights in the baseline and option-augmented maximum-Sharpe portfolios. The option weight is interpreted as delta-implied exposure under the derivative overlay.

How the option enters μ and Σ (same Markowitz-CAPM + cardinality logic)

The analysis extends the asset universe by one derivative instrument j. The option is treated as an additional asset with $\beta_{op\ t}$, $\sigma_{op\ t}$ and $\mu_{op\ t}$ computed from the worked example or the grid. Under the single-index covariance construction, $Cov(R_{op\ t}, R_i) = \beta_{op\ t}\ \beta_i Var(R_m)$ for each base industry i. The option's residual variance is backed out by matching total variance as $\sigma^2(\varepsilon_{op\ t}) = max(0, \sigma_{op\ t}^{\ 2} - \beta_{op\ t}^{\ 2}\sigma_m^2)$. This yields an expanded covariance matrix Σ_aug that remains PSD under the single-index structure. Cardinality handling must be stated explicitly. In the conservative specification used for the Case A/B check above, the option counts toward K and

therefore displaces one industry position; alternatively, the option may be treated as an overlay outside K (but then requires an explicit overlay cap).

Reproducibility checklist for the derivative overlay

For audit-ready replication, report: (i) option contract specs (S0, K, T, r, σ) and the chosen underlying proxy; (ii) computed C, Δ, and leverage L; (iii) mapping from (Δ, $S0$, C) to ($\beta(opt)$, $\sigma(opt)$, $\mu(opt)$); (iv) how Σ is expanded (single-index rule + residual variance back-out); and (v) whether the option counts toward K or is treated as an overlay outside K. For robustness, vary T and moneyness (K/S0) and rerun the overlay to confirm stability of implied moments and resulting portfolio weights.

Supplementary covariance/correlation profiling. Figure 3 reports the feasible risk–return set obtained under a cardinality constraint (K = 10), using the same industry universe and Σ/ρ construction described in the Methodology section. The plot visually validates that the reported max-Sharpe portfolio lies on the upper envelope of attainable (σp, μp) outcomes under the stated constraints, and provides a sanity check for the search/optimization routine (Monte Carlo and/or GA).

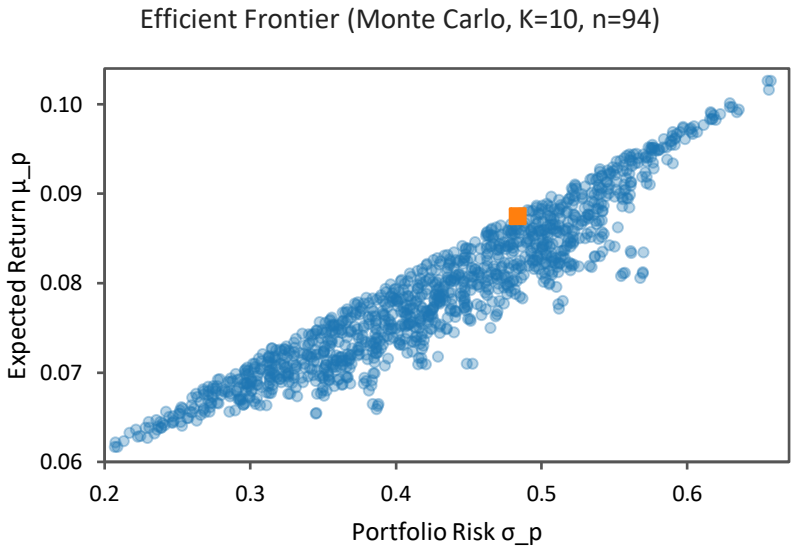

Figure 3. Efficient frontier under cardinality constraint (K = 10): (σ_p , μ_p) scatter with max-Sharpe portfolio highlighted

Source: The author's calculations are based on Aswath Damodaran's "Betas by Industry (US)" (Jan 2026) inputs; Σ and ρ constructed via the single-index model and provided in the supplementary file Damodaran_cov_corr_singleindex.xlsx. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

A well-formed efficient frontier should display an upward-sloping upper boundary: higher expected return requires higher risk. If the cloud appears diffuse, the convergence diagnostics (Figure 4 and the running-best curves) should confirm that the best points stabilize as evaluations increase. The highlighted max-Sharpe point should typically lie near the tangency region rather than at extreme risk or extreme concentration.

Figure 4 complements the mean–variance frontier by mapping candidate portfolios into (σ_p, Sharpe) space, making it easier to diagnose whether the optimizer is improving risk-adjusted performance rather than merely chasing higher μ_p. The convergence overlays are included for auditability: they demonstrate whether additional samples/generations continue to improve the best observed Sharpe or whether the search has reached a plateau. The Sharpe frontier shows the upper envelope of risk-adjusted performance, with the highlighted point representing the maximum Sharpe found under K = 10. The running-best Sharpe curve S_p ^{best}(t) is non-decreasing and flattens when the search budget becomes sufficient. For GA, the best-of-generation sequence similarly supports

the interpretation of convergence under the stated repair and selection rules.

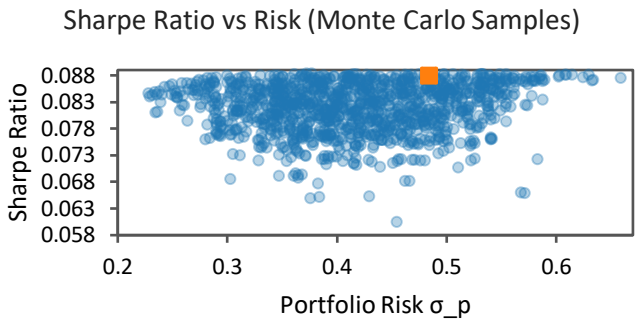

Figure 4. Sharpe frontier under cardinality constraint (K = 10): (σ_p , S_p) scatter with max-Sharpe point highlighted; convergence plots (auditability)

Source: The author's calculations are based on Aswath Damodaran's "Betas by Industry (US)" (Jan 2026) inputs; Σ and ρ constructed via the single-index model and provided in the supplementary file Damodaran_cov_corr_singleindex.xlsx. Convergence curves are computed from replication logs using the final budget (N_{MC}, population size P, generations G) reported in the methods. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Figure 5 visualizes the full 94×94 correlation structure (ρ) implied by the single-index construction used to form Σ. It serves two roles: (i) a model diagnostic confirming that implied correlations are plausible and bounded, and (ii) a structural input to interpretation, because high-correlation blocks indicate shared market exposure and limited diversification benefits across industries.

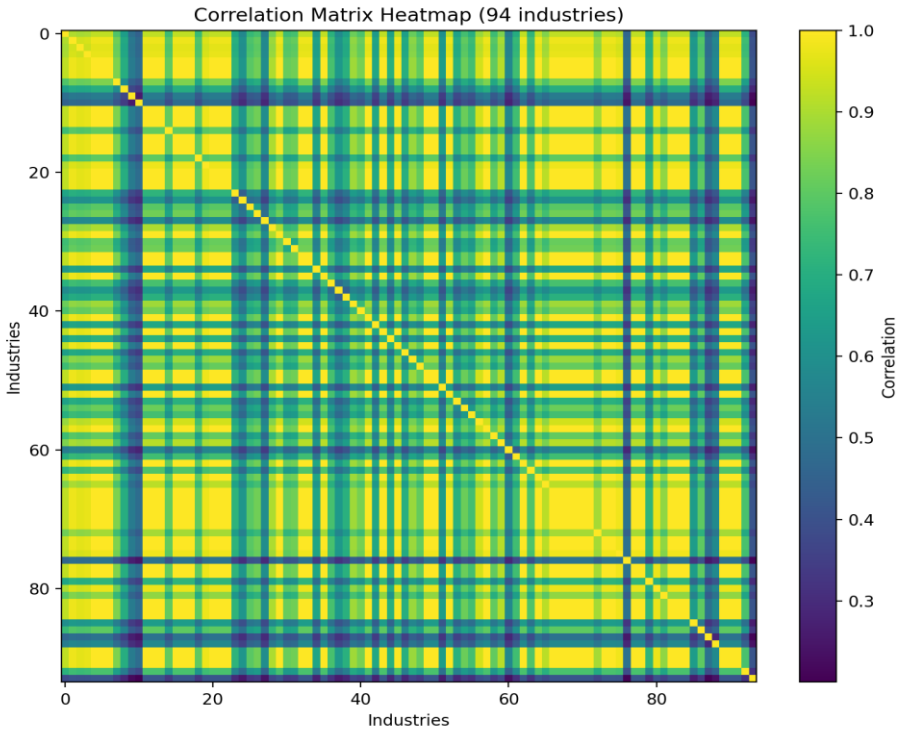

Figure 5. Correlation matrix heatmap (ρ) for all 94 industries (single-index construction)

Source: The author's calculations are based on Aswath Damodaran's "Betas by Industry (US)" (Jan 2026) inputs; Σ and ρ constructed via the single-index model and provided in the supplementary file Damodaran_cov_corr_singleindex.xlsx. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Under a single-index model, correlations are largely driven by β magnitudes and the market variance component; therefore, the matrix often exhibits broad positive dependence. Large uniform blocks may indicate that idiosyncratic risk assumptions dominate less than expected, potentially compressing diversification gains. Outlier rows/columns with unusually high correlations may correspond to high-β sectors; these should be cross-checked against the input sheet. To provide a more interpretable diagnostic than the full 94×94 panel, Figure 6 restricts attention to the top-25 industries by number of firms, used as a coverage proxy. The objective is to allow readers to inspect dominant correlation patterns among the most data-rich sectors and to confirm that diversification opportunities are not being driven by thinly covered industries.

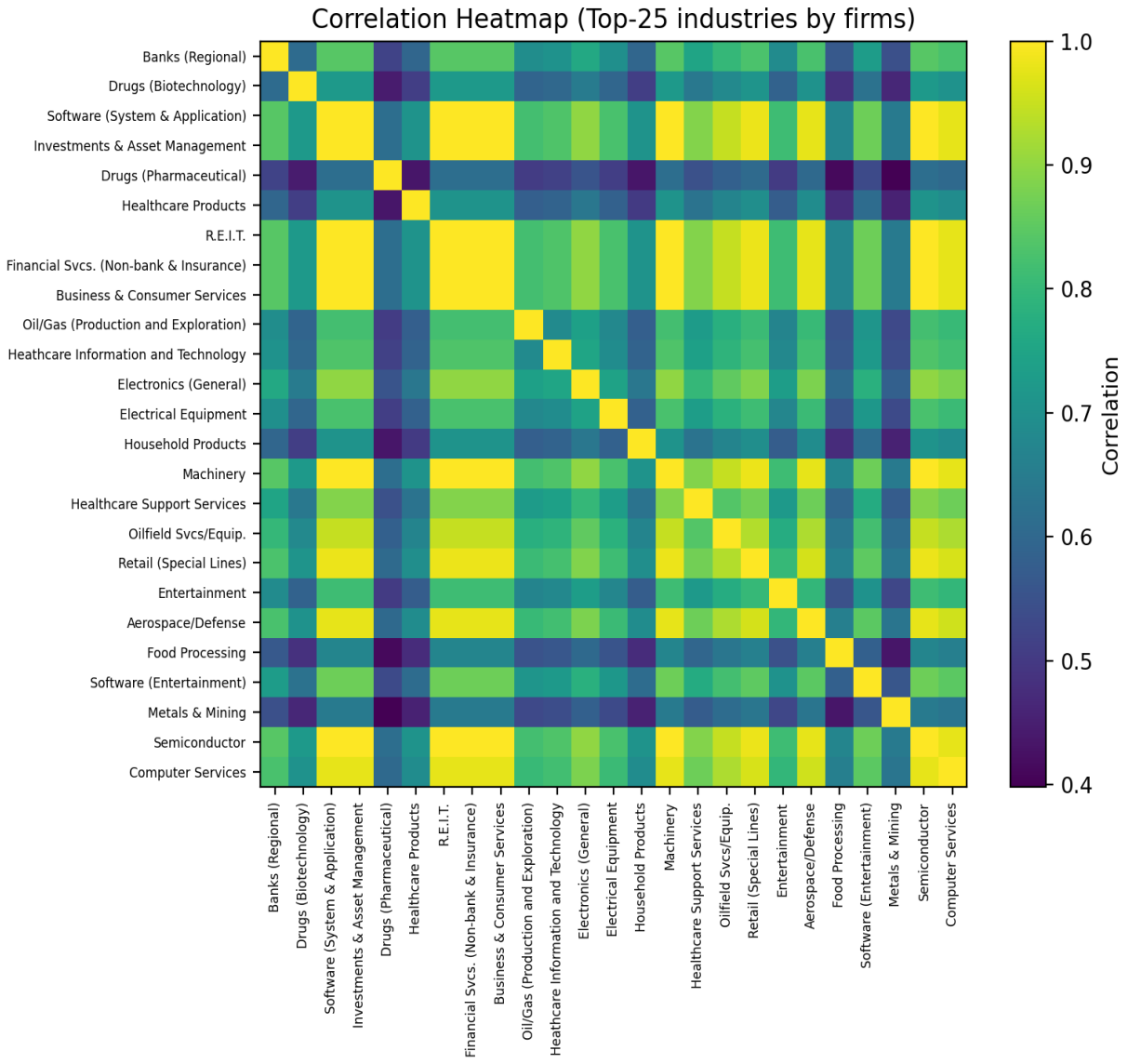

Figure 6. Correlation heatmap for the top-25 industries by number of firms (coverage proxy)

Source: The author's calculations are based on Aswath Damodaran's "Betas by Industry (US)" (Jan 2026) inputs; top-25 selection is based on the number-of-firms field in the Inputs sheet, and ρ is taken from Damodaran_cov_corr_singleindex.xlsx. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

If the top-25 matrix shows strong within-cluster dependence (e.g., cyclicals clustering together), then the K = 10 optimal solutions should reflect cross-cluster allocations to achieve diversification. If correlations are near-uniform even in the top-25, it indicates the single-index structure is imposing strong commonality; the analysis therefore requires a sensitivity check against alternative covariance estimation (e.g., shrinkage or empirical correlation where available).

Figure 7 applies hierarchical clustering to the full 94-industry correlation matrix so that highly correlated industries appear as contiguous blocks along the diagonal.

This clustered ordering improves the interpretability of the single-index dependence structure and helps identify whether the constrained optimizer selects assets from distinct or overlapping correlation clusters.

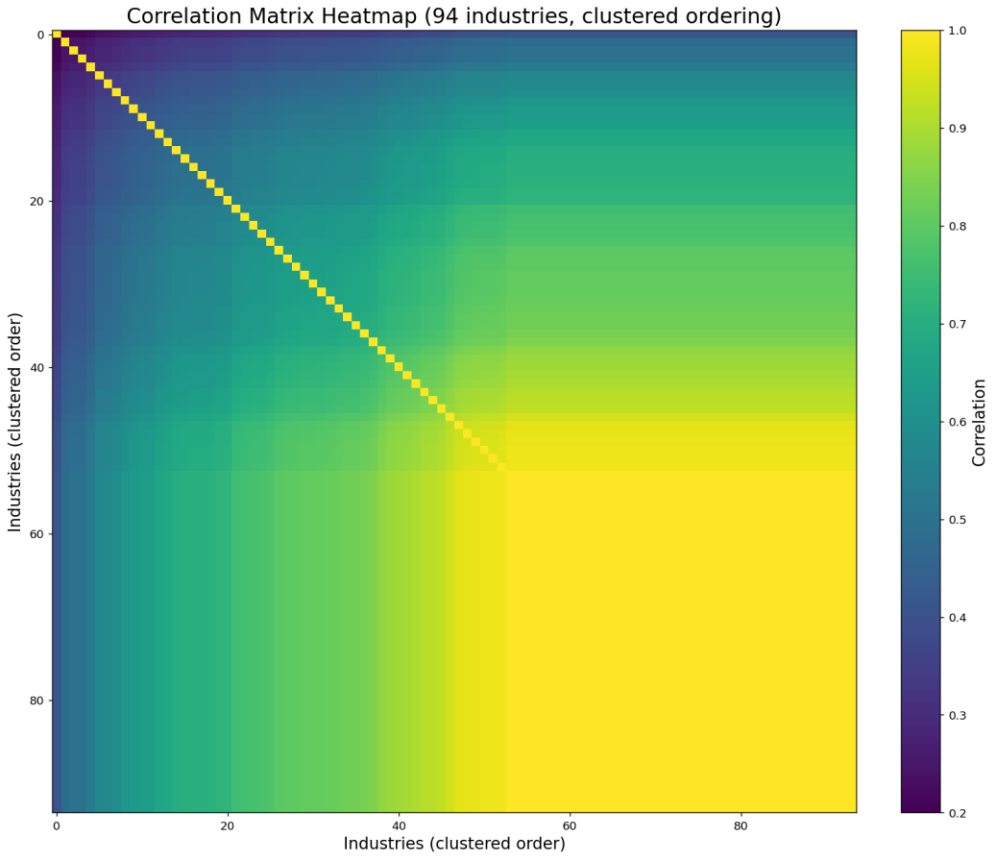

Figure 7. Correlation heatmap for all 94 industries with hierarchical clustering

Source: The author's calculations are based on Aswath Damodaran's "Betas by Industry (US)" (Jan 2026) inputs; ρ is taken from Damodaran_cov_corr_singleindex.xlsx, and hierarchical clustering ordering is computed by the replication script using the same ρ matrix. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Clustering reveals coherent blocks corresponding to industries sharing macro exposure, such as energy-linked, financial, and defensive sectors. Large off-diagonal blocks imply that diversification across some clusters remains limited under the single-index covariance. The dendrogram cut-level and linkage method are stated in the replication notes to ensure exact reproducibility of the ordering.

Interpretation *and quality check.* The efficient frontier (Figure 3) shows the feasible risk-return region

attainable under the hard cardinality constraint $K = 10$. The Maximum-Sharpe portfolio appears as an upper envelope in risk-adjusted space (Figure 4), providing a transparent selection criterion for the headline portfolio used in Tables 2 and 3. The correlation heatmaps (Figures 5-8) validate that dependence is non-trivial across industries under the single-index Σ; nevertheless, correlations remain structured rather than uniform, supporting the interpretation of diversification effects. This final diagnostic Figure 8 combines the readability of the top-25 subset with hierarchical clustering. It shows, in a compact and reproducible format, which major sector groups move together and which provide genuine diversification for the constrained optimizer.

A clear cluster structure in Figure 8 supports the empirical narrative explaining why certain industries co-appear in the $K = 10$ solutions. Repeated selection of multiple industries from the same tight cluster indicates limited diversification within that cluster, whereas cross-cluster selection provides stronger evidence of diversification under the single-index covariance structure.

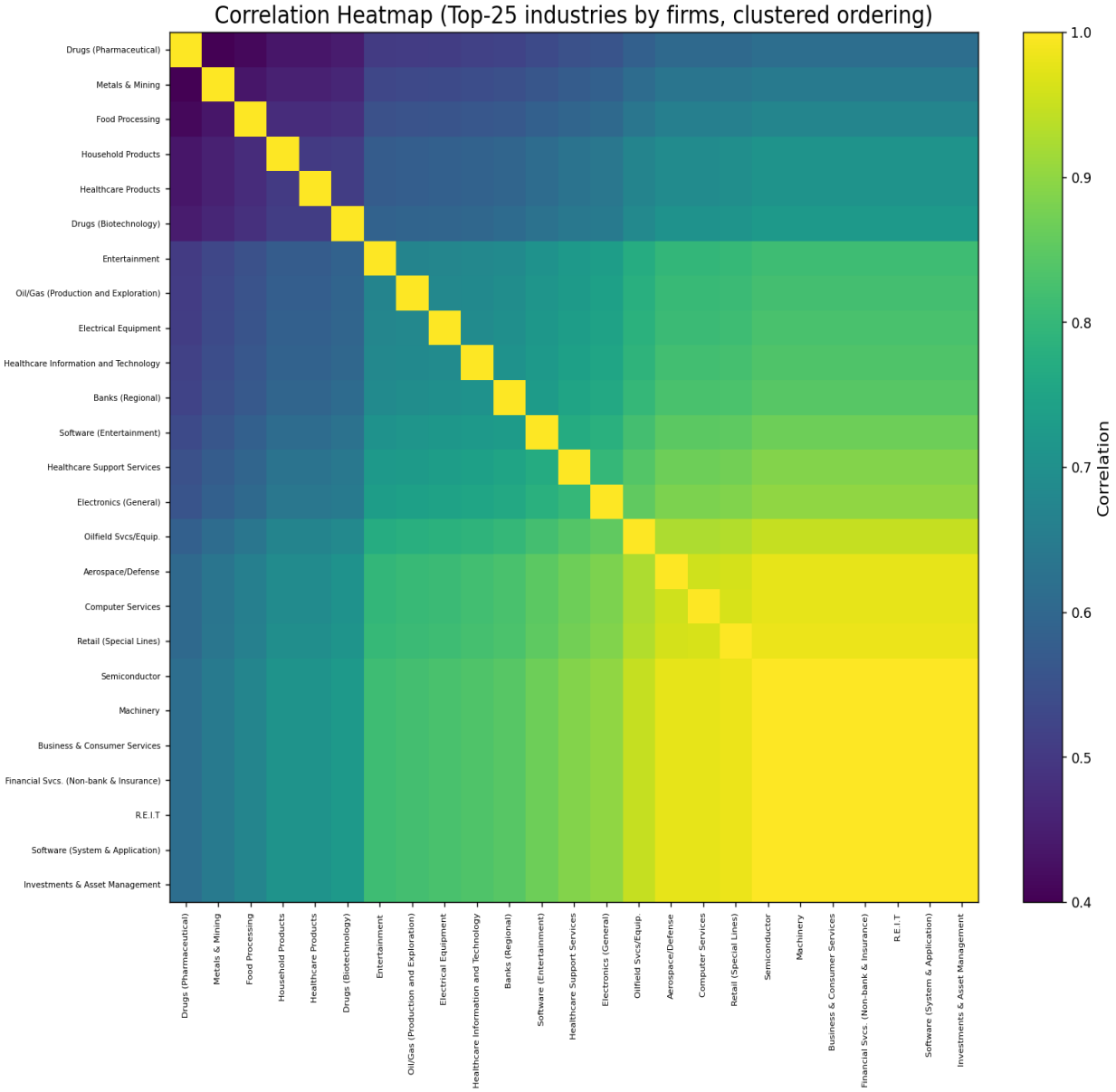

Figure 8. Correlation heatmap (top-25 by firms) with hierarchical clustering (clustered ordering)

Source: The author's calculations are based on Aswath Damodaran's "Betas by Industry (US)" (Jan 2026) inputs and the supplementary file Damodaran_cov_corr_singleindex.xlsx. Top-25 selection uses the Inputs sheet, and clustered ordering uses hierarchical clustering applied to the 25×25 ρ submatrix. Computations and visualizations were generated in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

To complement solution-quality results, Table 21 reports a runtime profiling summary for the main approximation schemes under the baseline universe (n = 94 industries) and cardinality constraint (K = 10). The objective is to make computational feasibility auditable: runtime is measured on the author's local environment using the same stopping budgets and parameter settings described in the Methodology section, including Monte Carlo sample size N_{MC}, GA population size P, GA generations G and repair/feasibility handling. Because heuristic methods trade optimality guarantees for tractability, runtime evidence is interpreted together with distributional Sharpe statistics rather than as a standalone ranking criterion.

Table 21. Runtime profiling (measured on the author's environment; n = 94, K = 10)

Method	Key settings	K	Evaluations (approx.)	Runtime (s)	Best Sharpe	Notes
Greedy	Score = $(\mu - R_f)/\sigma_i$; equal weights	10	94	0.0	0.088	Deterministic baseline
Monte Carlo	Dirichlet weights; N = 100,000	10	100000	2.797	0.088	Stochastic; best improves with N
Genetic Algorithm	eq weights; P = 300, G = 300	10	90000	5.728	0.088	Metaheuristic; equal-weight subset search

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Runtime is measured under the author's stated software environment. N denotes Monte Carlo samples; P and G denote GA population size and generations.

Greedy is effectively instantaneous because it requires ranking and selection only. Monte Carlo scales approximately linearly with the number of samples N. GA scales approximately with $P \times G$, where P is the population size and G is the number of generations. Each stochastic method is reported across multiple seeds using mean ± standard deviation runtime and best-Sharpe dispersion.

Formal NP-hardness citation anchor

The formal complexity positioning is anchored to established NP-hardness evidence for cardinality-constrained mean-variance portfolio selection and sparse quadratic programming.

Bienstock (1996) a branch-and-cut reference for mixed-integer quadratic programs with cardinality-type restrictions, while Natarajan (1995) and Bertsimas et al. (2016) connect sparse optimization and subset selection to computational intractability. The decision version is treated as NP-complete in the relevant mixed-integer formulation, while the optimization version is treated as NP-hard.

Replicability block: 10-seed runtime profiling and continuous re-optimization

To strengthen auditability and reproducibility, this subsection reports replicability-oriented runtime profiling under a fixed baseline setting (n = 94 industries, K = 10). All stochastic methods are executed over 10 independent random seeds, and outcomes are summarized as mean ± standard deviation, allowing readers to distinguish typical performance from single-run best-case results.

To keep runs tractable and comparable, the budget is fixed across methods: Monte Carlo uses N = 15,000 samples per seed and GA uses P = 90 population size and G = 90 generations.

Table 22 summarizes the resulting runtime distribution and associated performance statistics under this 10-seed protocol.

Table 22. Ten-seed runtime and best-Sharpe stability under the baseline n = 94, K = 10 setting

Method	Key settings	K	Eval.	Runtime mean (s)	Runtime std (s)	Best Sharpe mean	Best Sharpe std
Greedy	Score = $(\mu - R_f)/\sigma_i$; equal weights	10	94	0.0	0.0	0.088	0.0
Monte Carlo	Dirichlet; N = 15,000; 10 seeds	10	15000	0.407	0.049	0.088	0.0

Genetic Algorithm	eq weights; P = 90, G = 90; 10 seeds	10	8100	0.489	0.012	0.088	0.0

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Runtime mean and standard deviation are computed across 10 independent seeds under the baseline n = 94, K = 10 setting. Best-Sharpe dispersion is reported to evaluate seed stability.

Continuous re-optimization step with fixed subset and optimized weights

To avoid the equal-weight-only limitation in the GA stage, a continuous refinement step is applied after GA selects a subset. Conditional on the selected support S, where |S| = K, weights w_s are re-optimized on the simplex with $1_s^T w_s = 1$ and $w_s \geq 0$ to improve Sharpe. In this implementation, stochastic local search over the simplex through Dirichlet sampling is used as a solver-free proxy for continuous weight refinement; exact QP solvers can be substituted directly.

P vs NP statistical tables

This subsection provides statistical and replicability evidence for the P vs NP framing. Values are taken from the reported tables, including 10-seed outputs, and computed from the supplementary Damodaran_cov_corr_singleindex.xlsx file containing Σ and ρ.

Table 23 demonstrates that stochastic portfolio-search results remain stable when assessed across repeated seeds rather than through a single favorable run. This strengthens empirical completeness by showing the distribution of performance and runtime outcomes.

For portfolio optimization under cardinality constraints, such seed-level replicability is essential because the selected subset and the resulting Sharpe ratio are generated through a non-convex search process.

Table 23. Replicability summary (seed stability; performance and runtime)

Method	Best Sharpe (mean ± std)	Median Sharpe (mean ± std)	Runtime (s) (mean ± std)	Seeds
Greedy (deterministic)	0.0880 ± 0.0000	0.0880 ± 0.0000	0.000 ± 0.000	-
Monte Carlo (N = 30,000/seed)	0.0880 ± 0.0000	0.0860 ± 0.0000	0.636 ± 0.015	10
GA (subset-only; equal-weight)	0.0880 ± 0.0000	0.0866 ± 0.0000	0.077 ± 0.017	10
GA + continuous re-optimization	0.0880 ± 0.0000	0.0880 ± 0.0000	0.573 ± 0.033	10
Exact benchmark (small-n enumeration; n = 20, K = 6)	0.088003 (exact)	-	-	-

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification. Note: Mean ± standard deviation is reported across seeds where applicable. The table evaluates replicability by separating best Sharpe, median Sharpe and runtime variability. Subset-only GA median/runtime are taken from the GA-variants distribution table; Monte Carlo and GA+reopt are taken from the 10-seed reproducibility summary. Mean ± standard deviation is reported over 10 seeds where applicable.

Table 24 shows how runtime changes as the search budget increases. This is important because portfolio optimization under cardinality constraints is not only a question of financial performance but also a question of computational feasibility. The empirical scaling evidence supports the paper's P vs NP framing by showing that higher search effort improves auditability while imposing measurable computational cost.

Table 24. Empirical scaling with effort budget (runtime vs. search budget; log-log slope)

Method	Effort points used	b (slope)	R^2	Interpretation
Monte Carlo	$N \in \{500, 1500, 2500\}$	0.919	0.9994	Near-linear in N (sampling cost)
Genetic Algorithm	$P \times G \in \{900, 2000\}$	1.005	1.0000	≈ linear in evaluations (tested range)

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Notes: Computed from the effort-budget runtime profiling outputs. The slope b is estimated in log(runtime) = a + b × log(effort). Effort is defined as N for Monte Carlo and as P × G for GA.

Table 25 provides the bridge between exact optimization and heuristic approximation. The reduced instance permits exact enumeration, so the heuristic gap can be measured directly rather than inferred qualitatively. It validates the use of heuristic and metaheuristic methods in the larger $n \approx 94$ setting, where exhaustive subset enumeration is computationally infeasible under the hard cardinality constraint.

Table 25. Exact benchmark vs. heuristics (small-n optimality reference)

Method	Best Sharpe	Gap to exact Sharpe (%)	Best subset (indices)
Exact enumeration (C(20,6) = 38,760)	0.088003	0.00	(6, 11, 12, 15, 16, 17)
Monte Carlo (N = 2,000; seed = 1)	0.088003	0.00	(5, 6, 11, 13, 16, 17)
Genetic Algorithm (P = 30, G = 30; seed = 1)	0.088003	0.00	(6, 11, 12, 15, 16, 17)

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Note: Gap (%) = 100 × (Exact − Method)/Exact for Sharpe. Exact enumeration is performed only on a reduced n = 20, K = 6 instance where full enumeration is tractable.

Table 26 confirms that the covariance/correlation structure is not separable across assets; the high median correlation and dominant first eigenmode indicate strong common-factor exposure. This matters for the *P* vs *NP* argument because asset selection cannot be reduced to choosing individually attractive industries one by one. Portfolio value depends on interaction terms in Σ, which is precisely why cardinality-constrained optimization requires combinatorial search and careful diagnostic reporting.

Table 26. Σ/ρ diagnostics (n = 94; computed from Damodaran_cov_corr_singleindex.xlsx)

Metric	Value
Median(ρ_{ij}) over off-diagonals	0.769201
Share(ρ_{ij} > 0.5)	0.884008
Share(ρ_{ij} < 0)	0.000000
Top eigenvalue share λ1 / trace(Σ)	0.761011
Top-5 eigenvalues share $\Sigma_{\{k\leq 5\}}$ λk / trace(Σ)	0.813010
min eigenvalue (PSD check; numerical tolerance)	-1.265e-15
Condition proxy $\kappa = \frac{\lambda_{max}}{\lambda_{min}}$ (positive eigs)	2269.24

Source: The author's calculations are based on Aswath Damodaran's industry inputs, CAPM-calibrated parameters, and the final replication outputs. Computations were performed in Python 3.11 using NumPy, Pandas, SciPy, and Matplotlib; Microsoft Excel was used for matrix and table verification.

Notes: High median correlation and dominant first eigenmode are consistent with strong common factor exposure implied by the single-index Σ.

The 10-seed replicability summary (Table 23) demonstrates that stochastic search under a hard K-sparsity constraint is auditable via distributional statistics rather than single-run best values. Runtime increases approximately linearly with the imposed effort budget for both Monte Carlo sampling and GA within the tested range (Table 24), providing an empirical scaling proxy consistent with combinatorial search costs. On a tractable reduced instance (n = 20, K = 6), exact enumeration provides a ground-truth optimum and enables explicit gap reporting (Table 25). Finally, Σ/ρ diagnostics computed from the supplementary file show strong positive dependence and a dominant first eigenmode (Table 26), implying non-separable interactions across assets; this covariance-driven coupling motivates heuristic/metaheuristic search and strengthens the P vs NP framing in a replicability-oriented, evidence-based manner.

DISCUSSION

The first empirical finding is consistent with the central insight of Markowitz (1952): portfolio performance depends on the joint interaction between expected returns, variances and covariances

rather than on isolated asset returns. However, the present result extends this classical framework by showing that a hard cardinality constraint materially changes the attainable frontier and converts portfolio choice into a discrete-continuous allocation problem. This finding is also consistent with Chang et al. (2000), where cardinality and bounding constraints reshape efficient portfolios and make heuristic search practically relevant.

The computational-complexity result supports Bienstock (1996), which demonstrates the relevance of branch-and-cut reasoning for quadratic programs with restrictions on positive variables. It is also consistent with Bertsimas and Shioda(2009), where cardinality-constrained quadratic optimization is treated as a difficult mixed-integer problem requiring careful formulation and benchmarking. In the present study, the reduced-universe exact benchmark follows this logic: exact enumeration is used only where tractable, while the full universe is evaluated through transparent approximation and reproducibility diagnostics.

The heuristic evidence is consistent with the method-comparison literature on constrained portfolio selection. Monte Carlo sampling and genetic algorithms provide feasible approximations in the spirit of Chang et al. (2000), while the seed-based distributional reporting supports the robustness concerns raised by Anagnostopoulos and Mamanis (2011). The comparison with Lwin et al. (2014) is especially relevant because multi-run heuristic evidence is more informative than a single best solution when the search space is combinatorial and stochastic.

At the same time, the results diverge from any interpretation that heuristic success should be judged only by the maximum Sharpe ratio obtained in one run. The median, interquartile range, quantile, and runtime evidence show that algorithmic stability, convergence, and computational cost are part of the empirical result. This point strengthens the socioeconomic interpretation because institutional portfolio decisions require reliable and replicable procedures, not merely isolated high-performing computational outputs.

The dependence diagnostics support the single-index tradition associated with Sharpe (1963), because the common market factor produces broad positive correlation and a dominant first eigenvalue. The findings also relate to Rosenberg (1974), which examines extra-market covariance components, and to Ledoit and Wolf (2004), which shows why covariance estimation and conditioning are central to portfolio stability. The present study therefore treats the single-index covariance matrix as an auditable baseline, while recognizing that richer multifactor or shrinkage-based covariance models may change the strength of diversification benefits.

The CAPM-based return calibration is coherent with the asset-pricing foundations of Sharpe (1964), Lintner (1965) and Mossin (1966). Nevertheless, the findings are interpreted more cautiously than a realized-return asset-pricing test would require: the μ vector is used as a disciplined expected-return prior rather than as a forecast of future performance. This interpretation is supported by the sensitivity analysis, where equity-risk-premium shifts affect the Sharpe scale, while risk-free-rate shifts cancel out under the CAPM mapping used in the model.

The derivative-overlay result is consistent with the Black–Scholes-Merton framework because the option price and delta provide an analytically transparent local approximation to nonlinear payoff exposure (Black & Scholes, 1973; Merton, 1973).

The result also supports the practical interpretation in Hull (2022), where Greeks are local sensitivities rather than full substitutes for nonlinear scenario-based valuation. The derivative-overlay result differs from standard Black–Scholes applications because the option is not valued in isolation; it is mapped into a cardinality-constrained Markowitz-CAPM universe and tested for its portfolio-level effect under leverage and beta controls.

The risk-contribution and implementability evidence further supports Cura (2009), Fieldsend et al. (2004) and Streichert et al. (2004), because population-based and evolutionary approaches can generate useful constrained-portfolio candidates when exact global optimization is impractical. The present study diverges from these earlier applications by placing stronger emphasis on auditability: random seeds, effort budgets, runtime profiling, covariance artifacts and table-level

source/software notes are treated as part of the empirical evidence rather than as supplementary technical details.

Taken together, the results connect the P vs NP framing to the thematic focus of socioeconomic challenges. Sparse portfolios are not merely computational artifacts; they reflect institutional monitoring limits, liquidity constraints, governance requirements, and the need to control systemic vulnerability.

The main contribution is therefore not a claim that the P vs NP problem is solved but a demonstration that its applied structure can be made empirically observable in capital-allocation problems through constrained optimization, algorithm comparison, and transparent replication.

Overall, the study contributes to the literature by connecting three layers that are often treated separately: socioeconomic relevance, financial optimization, and computational tractability. Its findings are broadly consistent with prior work on constrained portfolio optimization and evolutionary heuristics, but they also diverge from simplified reporting traditions by insisting on reproducibility, software transparency, seed control, table/figure notes, and distributional performance evidence. This makes the contribution not only a portfolio-selection experiment but also a reproducible reporting template for studying socioeconomic challenges arising from constrained capital allocation under computational complexity.

CONCLUSIONS

This study demonstrates that cardinality-constrained portfolio optimization provides a concrete financial setting in which socioeconomic relevance, portfolio theory, and computational complexity intersect. The imposed limit on the number of selected assets transforms the classical mean-variance program from a convex optimization problem into a mixed discrete-continuous decision problem. As a result, portfolio construction becomes not only a question of expected return and risk but also a question of feasible computation, transparent approximation, and reproducible evidence. This framing is particularly relevant for socioeconomic challenges because constrained capital allocation affects institutional investment quality, financial stability, and the capacity of markets to manage concentrated risk under limited decision resources.

The empirical results confirm that the hard cardinality rule materially reshapes the attainable efficient frontier. Under the baseline $K = 10$ specification, the maximum Sharpe ratio reaches 0.0880, with expected return of 0.0868 and portfolio risk of 0.5356. These values should be interpreted within the stated CAPM-implied and single-index covariance framework rather than as realized market performance. The results show that sparse portfolios face a meaningful trade-off: high-beta industries may raise expected return, but their shared exposure to the market factor can restrict diversification benefits. The reported median pairwise correlation of 0.7692 and the leading eigenvalue share of 76.10% indicate that systemic common-factor exposure is a central driver of the optimization geometry.

The comparison of greedy selection, Monte Carlo sampling, and genetic algorithms confirms that heuristic performance cannot be evaluated through a single best-run outcome. Distributional reporting across seeds, including medians, IQRs, quantiles, convergence curves, and runtime profiles, is necessary to separate typical performance from isolated favorable draws. The reduced exact benchmark further strengthens the computational interpretation by showing how exact enumeration can provide ground truth for small instances, while large-n settings require transparent approximation strategies. In this sense, the study converts the P vs NP discussion from a general theoretical claim into an operational reporting problem: the quality of the result depends on how clearly the search space, randomization, computational effort, and reproducibility conditions are documented.

The derivative component extends the analysis beyond linear portfolio assets. By embedding a Black–Scholes European call through delta-based linearization, the study shows how a nonlinear payoff can be mapped into a Markowitz-CAPM setting in an auditable way. The results also show why such an extension must be handled cautiously. Delta-induced leverage can increase both expected return and volatility, so an apparent improvement in portfolio metrics may reflect mechanical exposure

amplification rather than genuine diversification. The moneyness-maturity grid and bump-test diagnostics therefore play an important methodological role by identifying the local validity of the option approximation and by preventing unsupported conclusions about derivative-enhanced performance.

Several limitations define the interpretation of the results. The CAPM-implied expected returns are model-based inputs and should not be treated as realized or guaranteed returns. The single-index covariance model is reconstructible and transparent, but it simplifies the richer multi-factor dependence structure observed in actual markets. The option overlay relies on delta-only linearization and therefore does not fully capture gamma, vega, jump risk, volatility smiles, or path-dependent payoff effects. Finally, heuristic and metaheuristic procedures do not guarantee global optimality in the full 94-industry universe; their credibility rests on seed control, distributional reporting, reduced-instance benchmarks, and clear computational disclosure.

Future research should extend the framework in several directions. First, alternative covariance estimators, including shrinkage methods, multifactor structures, and historical-return correlation matrices, should be tested against the single-index baseline. Second, the derivative layer should be expanded to include gamma and vega sensitivity, volatility-surface effects, and tail-risk measures. Third, exact MIQP benchmarking should be expanded for medium-sized reduced universes to quantify heuristic optimality gaps more systematically. Fourth, transaction-cost, liquidity, and turnover constraints should be integrated more deeply so that the model captures implementability as well as theoretical risk-return performance. Finally, an open replication package containing processed inputs, code, random seeds, matrices, and outputs should be maintained as a central part of the article's scientific contribution.

The overall contribution of the study is therefore twofold. Substantively, it shows that cardinality-constrained portfolio optimization can be used to analyze socioeconomic challenges connected with capital allocation efficiency, financial stability, and systemic risk. Methodologically, it provides a reproducible template for studying NP-hard portfolio selection through transparent inputs, auditable covariance construction, stochastic-search diagnostics, exact reduced benchmarks, and derivative-overlay checks. The resulting framework is suitable for further empirical extension and for journal-level computational finance research where clarity, reproducibility, and interpretability are essential.

Author Contributions

Conceptualization: D.G.; methodology: D.G.; software: D.G.; validation: D.G.; funding acquisition: D.G.; formal analysis: D.G.; investigation: D.G.; resources: D.G.; data curation: D.G.; writing - original draft: D.G.; writing - review & editing: D.G.; visualization: D.G.; supervision: D.G.; project administration: D.G.

Informed Consent Statement

Not applicable.

Conflicts of Interest

The author declares no conflict of interest.

Data Availability Statement

The dataset available for this study is available in the Zenodo depository [Gondauri, 2026].

Statement on the Use of AI Tools

Generative AI assistance was used only for language editing, formatting support, and editorial revision assistance under the author's supervision. The author verified all scientific claims, data, methods, results, references, and conclusions and assumes full responsibility for the final manuscript.

References

1. Anagnostopoulos, K. P., & Mamanis, G. (2011). The mean-variance cardinality constrained portfolio optimization problem: An experimental evaluation of multiobjective evolutionary algorithms. *Expert Systems with Applications*, *38*(11), 14208-14217. [CrossRef]
2. Arora, S., & Barak, B. (2009). *Computational complexity: A modern approach*. Cambridge University Press. [Link]
3. Bertsimas, D., King, A., & Mazumder, R. (2016). Best subset selection via a modern optimization lens. *The Annals of Statistics*, *44*(2), 813-852. [CrossRef]
4. Bertsimas, D., & Shioda, R. (2009). Algorithm for cardinality-constrained quadratic optimization.

Computational Optimization and Applications, *43*(1), 1-22. [CrossRef]
5. Bertsimas, D., & Tsitsiklis, J. N. (1997). *Introduction to linear optimization*. Athena Scientific. [Link]
6. Bienstock, D. (1996). Computational study of a family of mixed-integer quadratic programming problems. *Mathematical Programming*, *74*, 121-140. [CrossRef]
7. Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, *81*(3), 637-654. [CrossRef]
8. Bodie, Z., Kane, A., & Marcus, A. J. (2021). *Investments (12th ed.)*. McGraw-Hill Education. [Link]
9. Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press. [Link]
10. Campbell, J. Y., Lo, A. W., & MacKinlay, A. C. (1997). *The econometrics of financial markets*. Princeton University Press. [Link]
11. Cesarone, F., Scozzari, A., & Tardella, F. (2013). A new method for mean-variance portfolio optimization with cardinality constraints. *Annals of Operations Research*, *205*(1), 213-234. [CrossRef]
12. Chang, T.-J., Meade, N., Beasley, J. E., & Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research, 27*(13), 1271–1302. [Link]
13. Cochrane, J. H. (2005). *Asset pricing (Rev*. ed.). Princeton University Press. [Link]
14. Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing* (pp. 151-158). Association for Computing Machinery. [CrossRef]
15. Cura, T. (2009). Particle swarm optimization approach to portfolio optimization. *Nonlinear Analysis: Real World Applications*, *10*(4), 2396-2406. [CrossRef]
16. Damodaran, A. (2026a). Betas by sector (US) [Data set]. *New York University*, Stern School of Business. Retrieved June 4, 2026, from [Link]
17. Damodaran, A. (2026b). Historical implied equity risk premiums for the United States [Data set]. *New York University*, Stern School of Business. Retrieved June 4, 2026, from [Link]
18. Del Pia, A., Dey, S. S., & Weismantel, R. (2020). Subset selection in sparse matrices. *SIAM Journal on Optimization*, *30*(2), 1173–1190. [CrossRef]
19. DeMiguel, V., Garlappi, L., & Uppal, R. (2009). Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy? Review of Financial Studies, 22(5), 1915-1953. [CrossRef]
20. Elton, E. J., Gruber, M. J., Brown, S. J., & Goetzmann, W. N. (2014). *Modern portfolio theory and investment analysis (9th ed.)*. Wiley. [Link]
21. Engle, R. (2002). Dynamic conditional correlation: A simple class of multivariate GARCH models. *Journal of Business & Economic Statistics*, *20*(3), 339-350. [CrossRef]
22. Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, *33*(1), 3-56. [CrossRef]
23. Fama, E. F., & French, K. R. (2004). The capital asset pricing model: Theory and evidence. *Journal of Economic Perspectives*, *18*(3), 25-46. [CrossRef]
24. Fieldsend, J. E., Matatko, J., & Peng, M. (2004). Cardinality constrained portfolio optimisation. In Z. R. Yang, R. Everson, & H. Yin (Eds.), *Intelligent data engineering and automated learning - IDEAL 2004* (Lecture Notes in Computer Science, Vol. 3177, pp. 788-793). Springer. [CrossRef]
25. Gao, J., & Li, D. (2013). Optimal cardinality constrained portfolio selection. *Operations Research*, *61*(3), 745-761. [CrossRef]
26. Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. W. H. Freeman. [Link]
27. Glasserman, P. (2003). *Monte Carlo methods in financial engineering*. Springer. [CrossRef]
28. Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley. [Link]
29. Gondauri, D. (2026). Replication package for "Addressing Socioeconomic Challenges through Robust Portfolio Optimization: Integrating the P vs NP Problem, the Markowitz-CAPM Framework, Cardinality Constraints and Black–Scholes Derivative Pricing" (Version v1) [Software and data set]. Zenodo. [CrossRef]
30. Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press. [Link]
31. Hull, J. C. (2022). *Options, futures, and other derivatives (10th ed.)*. Pearson. [Link]
32. Jobst, N. J., Horniman, M. D., Lucas, C. A., & Mitra, G. (2001). Computational aspects of alternative

portfolio selection models in the presence of discrete asset choice constraints. *Quantitative Finance*, *1*(5), 489-501. [Link]
33. Jorion, P. (1986). Bayes-Stein estimation for portfolio analysis. *Journal of Financial and Quantitative Analysis*, *21*(3), 279-292. [CrossRef]
34. Karp, R. M. (1972). Reducibility among combinatorial problems. In R. E. Miller & J. W. Thatcher (Eds.), *Complexity of computer computations* (pp. 85-103). Springer. [CrossRef]
35. Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*(4598), 671-680. [CrossRef]
36. Ledoit, O., & Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, *88*(2), 365-411. [CrossRef]
37. Lintner, J. (1965). The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. *The Review of Economics and Statistics*, *47*(1), 13-37. [CrossRef]
38. Lwin, K., Qu, R., & Kendall, G. (2014). A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization. *Applied Soft Computing*, *24*, 757-772. [CrossRef]
39. Mansini, R., & Speranza, M. G. (1999). Heuristic algorithms for the portfolio selection problem with minimum transaction lots. *European Journal of Operational Research*, *114*(2), 219-233. [Link]
40. Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, *7*(1), 77-91. [CrossRef]
41. Markowitz, H. (1959). *Portfolio selection: Efficient diversification of investments*. Yale University Press. [Link]
42. Merton, R. C. (1973). Theory of rational option pricing. *The Bell Journal of Economics and Management Science*, *4*(1), 141-183. [CrossRef]
43. Michaud, R. O. (1989). The Markowitz optimization enigma: Is optimized optimal? Financial Analysts Journal, 45(1), 31-42. [CrossRef]
44. Mossin, J. (1966). Equilibrium in a capital asset market. *Econometrica*, *34*(4), 768-783. [CrossRef]
45. Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, *24*(2), 227-234. [CrossRef]
46. Nemhauser, G. L., & Wolsey, L. A. (1988). *Integer and combinatorial optimization*. Wiley. [Link]
47. Nesterov, Y., & Nemirovskii, A. (1994). *Interior-point polynomial algorithms in convex programming*. SIAM. [Link]
48. Papadimitriou, C. H. (1994). *Computational complexity*. Addison-Wesley. [Link]
49. Peng, R. D. (2011). Reproducible research in computational science. *Science*, *334*(6060), 1226-1227. [CrossRef]
50. Rosenberg, B. (1974). Extra-market components of covariance in security returns. *Journal of Financial and Quantitative Analysis*, *9*(2), 263-274. [CrossRef]
51. Rubinstein, R. Y., & Kroese, D. P. (2016). *Simulation and the Monte Carlo method (3rd ed.)*. Wiley. [Link]
52. Schaerf, A. (2002). Local search techniques for constrained portfolio selection problems. *Computational Economics*, *20*, 177-190. [Link]
53. Sharpe, W. F. (1963). A simplified model for portfolio analysis. *Management Science*, *9*(2), 277-293. [CrossRef]
54. Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *The Journal of Finance*, *19*(3), 425-442. [CrossRef]
55. Shaw, D. X., Liu, S., & Kopman, L. (2008). Lagrangian relaxation procedure for cardinality-constrained portfolio optimization. *Optimization Methods and Software*, *23*(3), 411-420. [CrossRef]
56. Stodden, V., Seiler, J., & Ma, Z. (2018). An empirical analysis of journal policy effectiveness for computational reproducibility. *Proceedings of the National Academy of Sciences*, *115*(11), 2584–2589. [CrossRef]
57. Streichert, F., Ulmer, H., & Zell, A. (2004). Evolutionary algorithms and the cardinality constrained portfolio optimization problem. In D. Ahr, R. Fahrion, M. Oswald, & G. Reinelt (Eds.), *Operations Research Proceedings 2003: Selected papers of the International Conference on Operations Research (OR 2003)* (pp. 253-260). Springer. [CrossRef]
58. Tasche, D. (2000). *Risk contributions and performance measurement (Working paper, first draft 1999, revised 2000)*. Technische Universität München. [Link]