SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

A Study of Portfolio Optimization in Discrete Time: From Markowitz to Reinforcement Learning

av

Zacharias Veiksaar

2025 - No K33

A Study of Portfolio Optimization in Discrete Time: From Markowitz to Reinforcement Learning

Zacharias Veiksaar

Självständigt arbete i matematik 15 högskolepoäng, grundnivå

Handledare: Yishao Zhou

2025

Abstract

This paper investigates portfolio optimization in discrete time, covering its development from the classical mean-variance framework, multi-period extensions, and modern reinforcement learning approaches. We begin with a rigorous treatment of the single-period case, deriving analytical solutions, highlighting their sensitivity to estimation errors, and proposing regularization as a solution to this sensitivity. We then extend the framework to a multi-period setting using dynamic programming, where we encounter time-inconsistency in the mean-variance formulation and propose a time-consistent reformulation that we solve analytically. As the reliance on estimating asset return distributions remains we propose reinforcement learning as a suitable model-free alternative, circumventing the need for explicit estimation of these parameters. We reformulate the time-consistent multi-period problem as a Markov decision process, prove that optimal solutions exist and argue these can be found within the reinforcement learning framework. The results highlight the mathematical structure of portfolio optimization, provide a broad treatment of limitations in classical approaches, and lay the groundwork for more robust machine learning methods as an alternative.

Sammanfattning

Denna artikel undersöker portföljoptimering i diskret tid, med fokus på områdets utveckling från den klassiska mean-varianceramverket genom utvidgningar till flera perioder och förstärkningsinlärning. Vi inleder med en rigorös behandling av enperiodsfallet där vi hittar analytiska lösningar, belyser deras känslighet för fel i parameterestimeringar, och föreslår regularisering som en lösning till denna känslighet. Vi utvidgar sedan ramverket till flera perioder genom dynamisk programmering, där vi stöter på tidsinkonsekvens i lösningen och föreslår en tidskonsekvent omformulering som vi löser analytiskt. Då beroendet av parameterestimering kvarstår föreslår vi förstärkningsinlärning som ett lämpligt modellfritt alternativ som kringgår parameterestimeringen. Vi omformulerar den tidskonsekventa flerperiodsformuleringen som en Markoviansk beslutsprocess, bevisar att optimala lösningar existerar, och motiverar att dessa kan hittas inom förstärkningsinlärningsramverket. Resultatet belyser den matematiska strukturen i portföljoptimering, tillhandahåller en genomförlig behandling av begränsningar i klassiska metoder, och lägger grunden för mer robusta maskininlärningsmetoder som alternativ.

Acknowledgements

I would like to thank my supervisor, Professor Yishao Zhou, for her extensive support and guidance throughout the writing process. I would also like to thank my close friends and classmates Emre Kaplaner and Alvar Mikkola for the many discussions that helped shape the ideas presented in this paper.

Contents

1 Introduction

Portfolio optimization lies at the intersection of mathematics and finance and regards the question of how to best allocate capital across several financial assets. It is nontrivial to translate the real-world problem of allocating capital into a proper mathematical framework, but attempting to do so yields valuable insight and opportunity for creativity. In the 1950s Harry Markowitz made a significant first step on this journey, modeling asset returns as random variables and balancing the expected return against the variance of return. In the paper he formulated the problem as a convex quadratic programming problem, which provides an analytical solution under some convexity assumptions.(Markowitz, 1952) This mean-variance trade-off captures the intuitive objective of desiring a flat and steep curve representing one's wealth, and has become a cornerstone in portfolio optimization, ultimately earning Markowitz the Nobel Prize in Economics.(Wikipedia, 2025b)

While the original mean-variance framework is a significant contribution, it relies on the strong assumption that we know the expected return and covariance matrix of the assets we wish to invest in. Estimating these parameters is a difficult task and it turns out that the solutions provided by Markowitz's framework are highly sensitive to changes in these variables. This fact has motivated both efforts to better model the expected return and covariance of assets as well as finding techniques to mitigate the impact of estimation errors in order to make solutions more robust.

Another limiting assumption of the classical Markowitz framework is that we restrict the setting to a static, single-period investment horizon, where the investor makes a single decision at the beginning and waits for an outcome at the end of the period. Investing however is not a static undertaking and allows for intermittent decisions to be made, down to millisecond intervals in extreme cases.(Wikipedia, 2025a) To loosen this assumption we extend the problem to a multi-period setting allowing for several allocation decisions to be made throughout the period using dynamic programming. Adopting the same objective function however we find that the problem is time-inconsistent and not compatible with dynamic programming principles, requiring another simplifying assumption in the objective function. The multi-period extension also inherits the problem of being reliant on difficult to obtain estimates of the expected value and covariance matrix of asset returns, along with the sensitivity to estimation errors from the single-period framework.

Given the limitations of the traditional frameworks we turn to a fundamentally different approach, introducing reinforcement learning as a framework not reliant on explicit estimation of asset return parameters. We reformulate the problem as a Markov decision process suitable for reinforcement learning algorithms and show that optimal allocations exist and can be found in this setting. We finish up by reviewing recent results in the literature showing encouraging results for reinforcement learning as a solution to portfolio optimization problems.

This paper aims to introduce the mathematical structure of portfolio optimization in discrete time and map out the developments in the field from classical theory to modern machine learning approaches. It also aims to offer a thorough treatment of each framework while acting as a foundation for further work and exploration.

2 Preliminaries

While we assume the reader has an understanding of mathematics that one might expect following the completion of a Bachelor's Degree in Mathematics, this section provides a review of the key concepts and results that appear throughout the paper. It is intended primarily as a reference to be consulted when unfamiliar facts arise and to prepare the reader for the topics to expect in the main text, rather than as a linear exposition.

2.1 Probability Theory

We begin by establishing some key results from probability theory that will prove useful when discussing the characteristics of asset returns. Asset prices are treated as random variables in this paper and discussing the expected value and variance of returns will translate into an understanding of the properties of an investor's portfolio, as when we are constructing a portfolio we are essentially creating a linear combination of these random variables.

Proposition 2.1 (Linearity of expectation)**.** *For random variables $X_1, \dots, X_n$ with expected values*

$$\mathbb{E}[X_1], \dots, \mathbb{E}[X_n]$$

the following property holds

$$\mathbb{E}[X_1 + \cdots + X_n] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_n].$$

Proof. We refer to Ross (2019, Page 317). □

Proposition 2.2 (Variance of a linear combination)**.** *The variance of a linear combination $\sum_{i=1}^{n} \alpha_i X_i$ of random variables $X_1, \dots, X_n$ is given by*

$$\operatorname{Var}\left(\sum_{i=1}^{n} \alpha_i X_i\right) = \sum_{i=1}^{n} \alpha_i^2 \operatorname{Var}(X_i) + 2\sum\sum_{i<j} \alpha_i \alpha_j \operatorname{Cov}(X_i, X_j).$$

Proof. We refer to Ross (2019, Page 341-342). □

We will also encounter a time-inconsistent dynamic programming problem where the conditional expectation and conditional variance formulas will prove useful.

Proposition 2.3 (Conditional expectation formula)**.** *Let X be a random variable with expected value $\mathbb{E}[X]$ and let Y be any random variable in the same probability space. Then*

$$\mathbb{E}[X] = \mathbb{E}\big[\mathbb{E}[X \mid Y]\big],$$

which is known as the conditional expectation formula.

Proof. We refer to Ross (2019, Page 351). □

Proposition 2.4 (Variance expressed using expected values)**.** *Let X be a random variable, then*

$$\operatorname{Var}(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2.$$

Proof. We refer to Ross (2019, Page 145). □

Proposition 2.5 (Conditional variance formula)**.** *Let X and Y be random variables in the same probability space. Then*

$$\operatorname{Var}(X) = \mathbb{E}\big[\operatorname{Var}(X \mid Y)\big] + \operatorname{Var}\big(\mathbb{E}[X \mid Y]\big).$$

Proof. We refer to Ross (2019, Page 366). □

Remark 2.1. *Throughout the paper when we come across time-indexed variables, we write $\mathbb{E}_t[\cdot]$ and $\operatorname{Var}_t(\cdot)$ to denote the expectation and variance condition on the information available at time t.*

2.2 Linear Algebra and Matrix Calculus

Portfolio optimization problems naturally leads to expressions that can be represented by vectors and matrices which can make calculations convenient. We can draw conclusions on the convexity of a quadratic form through the properties of its associated matrix and when finding optimal solutions we will work directly with matrices and vectors using matrix calculus.

Definition 2.1 (Positive definite matrices)**.** *A symmetric matrix $S = S^T$ is positive (semi-)definite if it has all positive (non-negative) eigenvalues. Equivalently, S is positive (semi-)definite if the energy $\boldsymbol{x}^T S\boldsymbol{x}$ is positive (non-negative) for all vectors $\boldsymbol{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$.(Strang, 2019, Page 45-46)*

Remark 2.2. *When making a reference to a positive definite or positive semi-definite matrix in this paper we are referring exclusively to symmetric matrices.*

Proposition 2.6 (Positive semi-definiteness of single-rank matrices)**.** *Any single-rank matrix $\boldsymbol{v}\boldsymbol{v}^T$ is positive semi-definite when $\boldsymbol{v} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$.*

Proof. By the energy definition of positive semi-definiteness we have

$$\boldsymbol{x}^T(\boldsymbol{v}\boldsymbol{v}^T)\boldsymbol{x} = (\boldsymbol{x}^T\boldsymbol{v})(\boldsymbol{v}^T\boldsymbol{x}) = (\boldsymbol{v}^T\boldsymbol{x})^2 \geq 0.$$

for any vector $\boldsymbol{x} \in \mathbb{R}^n$, so $\boldsymbol{v}\boldsymbol{v}^T$ is positive semi-definite. □

Positive semi-definite matrices arise naturally when working with the covariance matrix as the property ensures the variance is non-negative and symmetric matrices in general have some useful properties of their own.

Theorem 2.1 (The Spectral Theorem)**.** *Every real symmetric matrix has the form $S = Q\Lambda Q^T$ where Q is an orthogonal matrix such that $Q^{-1} = Q^T$ and $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$ is the diagonal matrix containing the eigenvalues of S.(Strang, 2019, Page 44)*

Proposition 2.7 (Eigenvalues of an inverse)**.** *Let $S \in \mathbb{R}^{n\times n}$ be an invertible symmetric matrix with eigenvalues $\lambda_1, \ldots, \lambda_n$, then the eigenvalues of its inverse S^{-1} will be $\frac{1}{\lambda_1}, \ldots, \frac{1}{\lambda_n}$.*

Proof. As S is symmetric the Spectral Theorem yields the identity $S = Q\Lambda Q^T$. Since Q is orthogonal we have $Q^{-1} = Q^T$ and propose the inverse $S^{-1} = Q\Lambda^{-1}Q^T$ with $\Lambda^{-1} = \text{diag}(\frac{1}{\lambda_1}, \ldots, \frac{1}{\lambda_n})$. Clearly

$$SS^{-1} = Q\Lambda Q^T Q\Lambda^{-1}Q^T = Q\Lambda\Lambda^{-1}Q^T = QQ^T = I,$$

and our proposed inverse was valid. Since Λ^{-1} contains the eigenvalues of S^{-1} by Theorem 2.1 the proof is done. □

The analytical solutions to the portfolio optimization problems we will encounter all include the inverse of the covariance matrix. If we imagine a near-singular matrix S with one or many eigenvalues close to zero, we can see that the eigenvalues of its inverse can become large. This can pose a problem as solutions may become very large or sensitive to small perturbations in the other inputs. To better understand and treat this we may want to shift the eigenvalues of S.

Proposition 2.8 (Shifting eigenvalues)**.** *Let $\alpha \in \mathbb{R}$ and I be the $n \times n$ identity matrix. Adding the diagonal matrix αI to a square matrix $A \in \mathbb{R}^{n\times n}$ shifts the eigenvalues λ_i of A by α such that the eigenvalue λ_i of A becomes the eigenvalue $\lambda_i + \alpha$ for $A + \alpha I$.*

Proof. We refer to Strang (2019, Page 38). □

Introducing the shifting of eigenvalues as a solution to size and sensitivity in matrices naturally raises the question what these two properties actually entail. In linear algebra we can formalize the notion of size and sensitivity of vectors and matrices using norms and the condition number of matrices. We refer to Strang (2019, Page 88) for the original definitions and present a selection of these here.

Definition 2.2 (Vector norms)**.** *The L^2-norm, or the Euclidean norm, of a vector $\boldsymbol{x} \in \mathbb{R}^n$ is defined as*

$$\|\boldsymbol{x}\|_2 := \sqrt{|x_1|^2 + \cdots + |x_n|^2} = \sqrt{\boldsymbol{x}^T \boldsymbol{x}},$$

where x_i is the scalar value of coordinate i of $\boldsymbol{x}$. The L^1-norm is defined as

$$\|\boldsymbol{x}\|_1 := |x_1| + \cdots + |x_n|,$$

and the L^∞-norm, or the max-norm, is defined as

$$\|\boldsymbol{x}\|_\infty = \max_{i=1,2,\ldots,n} |x_i|.$$

Through these vector norms we can now define norms for matrices, which we can interpret as the largest growth factor of some matrix $A \in \mathbb{R}^{m\times n}$. We do this by comparing $\|A\boldsymbol{v}\|$ to $\|\boldsymbol{v}\|$ for some norm $\|\cdot\|$ and some vector $\boldsymbol{v} \in \mathbb{R}^n$.

Definition 2.3 (Induced operator norms)**.** *For a matrix $A \in \mathbb{R}^{m\times n}$ we define its induced norm by*

$$\|A\| := \max_{\boldsymbol{v}\in\mathbb{R}^n\setminus\{\boldsymbol{0}\}} \frac{\|A\boldsymbol{v}\|}{\|\boldsymbol{v}\|},$$

for any vector norm $\|\cdot\|$. If the matrix norm is induced by the L^2-norm for vectors we denote the associated operator norm of A by $\|A\|_2$ as expected, and analogously for other choices of vector norm.

With a notion of the size of matrices, we can now begin looking at how possible errors in measurements can propagate through a matrix calculation. A common way to analyze such a propagation is through condition numbers. Consider the equation $A\boldsymbol{x} = \boldsymbol{b}$ where we wish to solve for $\boldsymbol{x}$. Suppose $\boldsymbol{b}$ represents the true state of our right hand side but that we, due to an imperfect measurement or other disturbance, only have access to an imperfect $\boldsymbol{b}_e$, which includes some error $\boldsymbol{e} := \boldsymbol{b}_e - \boldsymbol{b}$, and denote the corresponding solution by $\boldsymbol{x}_e$. Now by measuring the relative error in our imperfect solution relative to the original relative error in the measurement we can find out how much our matrix A magnified the error. We find that

$$\frac{\|\boldsymbol{x}_e - \boldsymbol{x}\|/\|\boldsymbol{x}\|}{\|\boldsymbol{b}_e - \boldsymbol{b}\|/\|\boldsymbol{b}\|} = \frac{\|A^{-1}\boldsymbol{e}\|/\|A^{-1}\boldsymbol{b}\|}{\|\boldsymbol{e}\|/\|\boldsymbol{b}\|} = \frac{\|A^{-1}\boldsymbol{e}\|}{\|\boldsymbol{e}\|}\frac{\|\boldsymbol{b}\|}{\|A^{-1}\boldsymbol{b}\|} = \frac{\|A^{-1}\boldsymbol{e}\|}{\|\boldsymbol{e}\|}\frac{\|A\boldsymbol{x}\|}{\|\boldsymbol{x}\|}.$$

Using this identity we can find the maximum possible relative error by

$$\max_{\boldsymbol{e},\boldsymbol{x}\neq 0} \frac{\|A^{-1}\boldsymbol{e}\|}{\|\boldsymbol{e}\|}\frac{\|A\boldsymbol{x}\|}{\|\boldsymbol{x}\|} = \max_{\boldsymbol{e}\neq 0} \frac{\|A^{-1}\boldsymbol{e}\|}{\|\boldsymbol{e}\|} \max_{\boldsymbol{x}\neq 0} \frac{\|A\boldsymbol{x}\|}{\|\boldsymbol{x}\|} = \|A^{-1}\|\|A\|,$$

through Definition 2.3.

Definition 2.4 (Condition numbers)**.** *We define the condition number $\kappa(A)$ of a matrix A as*

$$\kappa(A) = \|A^{-1}\|\|A\|,$$

where $\|\cdot\|$ is any matrix norm.

Remark 2.3. *For least squares problems one usually uses the L^2-norm.*

Vector and matrix norms as well as condition numbers will allow us to discuss the size and sensitivity of matrices and will help in discussing the problem of parameter estimation in our single and multi-period problem formulations.

Working directly with the matrix and vector representation of our objective functions matrix calculus will be a useful tool and below we collect a few standard results as presented in Handa (2011) which are used throughout this paper.

Proposition 2.9 (Gradient of a linear form). *Let $\boldsymbol{a} \in \mathbb{R}^n$. Then the gradient of the linear form $f(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{x}$ is*

$$\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = \boldsymbol{a}.$$

Proposition 2.10 (Gradient of a quadratic form). *Let $A \in \mathbb{R}^{n\times n}$ be a matrix, then the gradient of the quadratic form $f(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x}$ is*

$$\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = A\boldsymbol{x} + A^T\boldsymbol{x}.$$

If A is symmetric, this simplifies to

$$\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = 2A\boldsymbol{x}.$$

Proposition 2.11 (Hessian of a quadratic form). *The Hessian of $f(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x}$ is*

$$\nabla^2_{\boldsymbol{x}} f(\boldsymbol{x}) = A + A^T.$$

In particular, if A is symmetric,

$$\nabla^2_{\boldsymbol{x}} f(\boldsymbol{x}) = 2A.$$

The results in this section will be thoroughly used in deriving analytical solutions in the single and multi-period setting and allows us to better understand how estimation errors can propagate to disturb our final solutions.

2.3 Convex Analysis and Optimization

As we will be solving several convex optimization problems throughout this paper we introduce some results from convex analysis and optimization which lay a foundation for later discussions on convexity, optimality, and finding our solutions.

Definition 2.5 (Convex sets). *A set S in $\mathbb{R}^n$ is said to be convex if the line segment joining any two points of the set also belongs to the set. In other words, if $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are in S, then $\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2$, must also belong to S for each $\lambda \in [0,1]$.(Bazaraa et al., 2006, Page 40))*

Definition 2.6 (Convex functions). *Let $f : S \to R$ where S is a nonempty convex set in R^n. The function f is said to be convex on S if*

$$f(\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2) \leq \lambda f(\boldsymbol{x}_1) + (1-\lambda)f(\boldsymbol{x}_2)$$

for each $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$ and for each $\lambda \in (0,1)$. The function f is called strictly convex on S if the above inequality is true as a strict inequality for each distinct $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in S and for each $\lambda \in (0,1)$.(Bazaraa et al., 2006, Page 98)

Theorem 2.2 (Sufficient and necessary condition for optimality). *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function, then $\bar{\boldsymbol{x}}$ is a global minimum if and only if $\nabla f(\bar{\boldsymbol{x}}) = \boldsymbol{0}$.*

Proof. We refer to Bazaraa et al. (2006, Page 169) for a proof. □

What follows here are results adapted from lectures and notes provided by Y.Zhou of Stockholm University for the course *Optimization (MM7028)* in the fall semester of 2024.(Zhou, 2024)

Lemma 2.1. *Consider a function $f : \mathbb{R}^n \to \mathbb{R}$, and for any point $\bar{\boldsymbol{x}} \in \mathbb{R}^n$ and a non-zero direction $\boldsymbol{d} \in \mathbb{R}^n$, define $\varphi(\lambda) = f(\bar{\boldsymbol{x}} + \lambda\boldsymbol{d})$ as a function of $\lambda \in \mathbb{R}$. Then, f is convex if and only if φ is convex for all $\bar{\boldsymbol{x}}$ and $\boldsymbol{d}$ in $\mathbb{R}^n \setminus \{\boldsymbol{0}\}$.*

Proof. ($\Longrightarrow$): Given any $\bar{\boldsymbol{x}}$ and $\boldsymbol{d}$ in $\mathbb{R}^n \setminus \{\boldsymbol{0}\}$. If f is convex, then for any $\lambda_1, \lambda_2 \in \mathbb{R}$ and for any $0 \leq \alpha \leq 1$, we have

$$\begin{aligned}
\varphi(\alpha\lambda_1 + (1-\alpha)\lambda_2) &= f(\alpha[\bar{\boldsymbol{x}} + \lambda_1\boldsymbol{d}] + (1-\alpha)[\bar{\boldsymbol{x}} + \lambda_2\boldsymbol{d}]) \\
&\leq \alpha f(\bar{\boldsymbol{x}} + \lambda_1\boldsymbol{d}) + (1-\alpha)f(\bar{\boldsymbol{x}} + \lambda_2\boldsymbol{d}) \\
&= \alpha\varphi(\lambda_1) + (1-\alpha)\varphi(\lambda_2).
\end{aligned}$$

Hence, φ is convex.

($\Longleftarrow$): Suppose that $\varphi(\lambda)$, $\lambda \in \mathbb{R}$, is convex for all $\bar{\boldsymbol{x}}$ and $\boldsymbol{d}$ in $\mathbb{R}^n \setminus \{\mathbf{0}\}$. Then for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^n$, and $0 \leq \lambda \leq 1$, we have

$$\begin{aligned}
\lambda f(\boldsymbol{x}_1) + (1-\lambda) f(\boldsymbol{x}_2) &= \lambda f(\boldsymbol{x}_1 + 0(\boldsymbol{x}_2 - \boldsymbol{x}_1)) + (1-\lambda) f(\boldsymbol{x}_1 + 1(\boldsymbol{x}_2 - \boldsymbol{x}_1)) \\
&= \{\text{Here we choose } \boldsymbol{d} = \boldsymbol{x}_2 - \boldsymbol{x}_1 \text{ and } \boldsymbol{x} = \boldsymbol{x}_1\} \\
&= \lambda\varphi(0) + (1-\lambda)\varphi(1) \\
&\geq \varphi(1-\lambda) = f(\boldsymbol{x}_1 + (1-\lambda)(\boldsymbol{x}_2 - \boldsymbol{x}_1)) \\
&= f(\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2)
\end{aligned}$$

and so f is convex. □

Corollary 2.1. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex on $C \subseteq \mathbb{R}^n$ if and only if $\varphi(\lambda) = f((1-\lambda)\boldsymbol{x} + \lambda \boldsymbol{y})$ for all $\boldsymbol{x}, \boldsymbol{y} \in C$.*

Proof. Take $\boldsymbol{d}$ in Lemma 2.1 as $\boldsymbol{y} - \boldsymbol{x}$. □

Theorem 2.3. *A differentiable function f on a convex set $\in \mathbb{R}^n$ is convex if and only if*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x}) \forall \boldsymbol{x}, \boldsymbol{y} \in C$$

Proof. By corollary 2.1 f is convex if and only if $\varphi(\lambda) = f((1-\lambda)\boldsymbol{x} + \lambda \boldsymbol{y})$ is convex. By the two-point condition φ is convex if and only if

$$\varphi(\lambda) \geq \varphi(\mu) + \varphi'(\mu)(\lambda - \mu), \forall \lambda, \mu \tag{1}$$

Note that the directional derivative of f at $\boldsymbol{x}$ along the direction $\boldsymbol{y} - \boldsymbol{x}$ is $\nabla f(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})$, i.e.

$$\varphi'(\mu) = \nabla f(\boldsymbol{x} + \mu(\boldsymbol{y} - \boldsymbol{x}))(\boldsymbol{y} - \boldsymbol{x}). \tag{2}$$

Substituting (2) into (1), and choosing $\lambda = 1$, $\mu = 0$ yields the desired inequality. □

Theorem 2.4. *Assume that a function f is twice differentiable on $C \in \mathbb{R}^n$, where C is an open convex set in $\mathbb{R}^n$. Then f is strictly convex (convex) if and only if the Hessian $H(\boldsymbol{x}) = \{\frac{\partial^2 f}{\partial \boldsymbol{x}_i \partial \boldsymbol{x}_j}\}$ is positive (semi-)definite everywhere, denoted by $H \succ (\succeq) 0$ for all $\boldsymbol{x} \in \mathbb{R}^n$.*

Proof. By definition, $H(\boldsymbol{x})$ is positive semi-definite if and only $\langle H\boldsymbol{x}, \boldsymbol{x}\rangle \geq 0$ for all $\boldsymbol{x} \in \mathbb{R}^n$. Since f is convex, by corollary 2.1 $\varphi(\lambda) = f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x}))$ is convex for all $\boldsymbol{x}, \boldsymbol{y}, \in C$. By the 1-pt condition $\varphi'' \geq 0$. But

$$\varphi''(\lambda) = \langle H(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x}))(\boldsymbol{y} - \boldsymbol{x}), (\boldsymbol{y} - \boldsymbol{x})\rangle$$

Since C is an open set, the vector $\boldsymbol{y} - \boldsymbol{x}$ is any direction in C. Hence $\varphi''(\lambda) \geq 0$ for all $\lambda, \boldsymbol{x}, \boldsymbol{y}$ if and only if $H(\boldsymbol{x})$ is positive semi-definite everywhere. □

This marks the end of the results adapted from the course *Optimization (MM7028)* by Y.Zhou.(Zhou, 2024)

Theorem 2.5 (Jensen's inequality)**.** *If $f(x)$ is a convex function then*

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$$

provided that the expectations exist and are finite.

Proof. We refer to Ross (2019, Page 427) for a proof. □

Proposition 2.12. *A function $f : \mathbb{R}^n$ on the form $f(\boldsymbol{x}) = \boldsymbol{x}^T A \boldsymbol{x}$ where $A \in \mathbb{R}^{n \times n}$ is convex if A is symmetric and positive semi-definite.*

Proof. Consider $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in $\mathbb{R}^n$ and suppose A is symmetric and positive semi-definite. By Proposition 2.11 the hessian of f is $H(\boldsymbol{x}) = 2A$, and since A is positive semi-definite is convex by Theorem 2.4. □

2.4 Fixed Point Theorem and Contraction Mappings

The Banach fixed point theorem is a result from real analysis which has important implications for our discussion of reinforcement learning. In particular, it provides the theoretical foundation for guaranteeing convergence of some algorithms which supports reinforcement learning as a feasible method for solving portfolio optimization problems. We begin by presenting some basic definitions in real analysis from Rudin (1976) and build up to proving the theorem.

Definition 2.7 (Metric spaces)**.** *A set X, whose elements we shall call **points**, is said to be a **metric space** if with any two points p and q of X there is associated a real number $d(p,q)$, called the **distance** from p to q, such that*

a) $d(p,q) > 0$ if $p \neq q$; $d(p,p) = 0$;

b) $d(p,q) = d(q,p)$;

c) $d(p,q) \leq d(p,r) + d(r,q)$, for any $r \in X$.

*Any function with these three properties is called a **distance function**, or a **metric**.(Rudin, 1976, Page 32)*

Definition 2.8 (Convergent sequences)**.** *A sequence $\{p_n\}$ in a metric space X is said to **converge** if there is a point $p \in X$ with the following property: For every $\varepsilon > 0$ there is an integer N such that $n \geq N$ implies that $d(p_n, p) < \varepsilon$. (Here d denotes the distance in X.)(Rudin, 1976, Page 47)*

Definition 2.9 (Cauchy sequences)**.** *A sequence $\{p_n\}$ in a metric space X is said to be a **Cauchy sequence** if for every $\varepsilon > 0$ there is an integer N such that $d(p_n, p_m) < \varepsilon$ if $n \geq N$ and $m \geq N$.(Rudin, 1976, Page 52)*

Definition 2.10 (Complete metric spaces)**.** *A metric space in which every Cauchy sequence converges is said to be complete.(Rudin, 1976, Page 54)*

Definition 2.11 (Contraction mappings)**.** *Let X be a metric space, with metric d. If φ maps X into X and if there is a number $c < 1$ such that*

$$d\big(\varphi(x), \varphi(y)\big) \leq cd(x,y)$$

*for all $x, y \in X$, then φ is said to be a **contraction** of X into X.(Rudin, 1976, Page 220)*

Theorem 2.6 (Uniqueness and existence of a fixed point)**.** *Let (X,d) be a complete metric space and let a function $f : X \to X$ be a contraction of X into X. Then f has a unique fixed point $x^* \in X$, i.e. a unique point such that $f^n(x) := \underbrace{f\big(f(\cdots f(x))\big)}_{n \text{ times}} \to x^*$ as $n \to \infty$.*

Proof. To show uniqueness we assume by contradiction that there are two distinct fixed points $x_1^* \neq x_2^*$ such that $f(x_1^*) = x_1^*$ and $f(x_2^*) = x_2^*$. This gives us $d\big(f(x_1^*), f(x_2^*)\big) = d(x_1^*, x_2^*)$, but since f is a contraction this is impossible. Our assumption must have been false and $x_1^* = x_2^*$.

To show the existence of the fixed point x^*we let $x_n := f^n(x)$. We want to first show that $\{x_n\}$ is a Cauchy sequence. By the triangle inequality we have

$$\begin{aligned} d(x_m, x_n) &\leq d(x_m, x_{m-1}) + d(x_{m-1}, x_n) \\ &\leq d(x_m, x_{m-1}) + d(x_{m-1}, x_{m-2}) + d(x_{m-2}, x_n) \\ &\leq d(x_m, x_{m-1}) + \cdots + d(x_{n+1}, x_n) \\ &= d\big(f^m(x), f^{m-1}(x)\big) + \cdots + d\big(f^{n+1}(x), f^n(x)\big). \end{aligned}$$

By contraction we get

$$d\big(f^m(x), f^{m-1}(x)\big) \leq c \cdot d\big(f^{m-1}(x), f^{m-2}(x)\big) \leq \cdots \leq c^{m-1} \cdot d\big(f(x), x\big).$$

Similarly we find

$$\begin{aligned} d\big(f^{m-1}(x), f^{m-2}(x)\big) &\leq c^{m-2} \cdot d\big(f(x), x\big), \\ &\vdots \\ d\big(f^{n+1}(x), f^{n}(x)\big) &\leq c^{n} \cdot d\big(f(x), x\big). \end{aligned}$$

Through this we now get

$$\begin{aligned} d(x_m, x_n) &\leq c^{m-1} \cdot d\big(f(x), x\big) + \cdots + c^{n} \cdot d\big(f(x), x\big) \\ &= \sum_{k=n}^{m-1} c^k \cdot d\big(f(x), x\big) \\ &= c^n \sum_{k=0}^{m-n-1} c^k \cdot d\big(f(x), x\big) \\ &\leq \frac{c^n}{1-c} d\big(f(x), x\big). \end{aligned}$$

Choosing $n = N$ sufficiently large we have $d(x_m, x_n) < \varepsilon$ for all $m, n \geq N$ and $\{x_n\}$ is a Cauchy sequence. Since X is complete the sequence converges and $x^* = \lim_{n\to\infty} x_n$ exists. □

2.5 Financial Concepts

In finance we often look at how the prices of financial instruments develop and the study of publicly traded financial instruments such as stocks and bonds is central. While many people have heard of or have some intuition for the notions of risk and return it is beneficial to define these formally. In financial mathematics the return of a financial instrument α_i over one period ending at period t is defined as the change in price over the period.

Definition 2.12 (Returns)**.** *The net return of a financial instrument α_i over the period from t to $t+1$ is defined as*

$$R_{t+1}(\alpha_i) := \frac{P_{t+1}(\alpha_i) - P_t(\alpha_i)}{P_t(\alpha_i)},$$

and expresses the percentage change of the asset price $P(\alpha)$ for asset α_i, referred to as the return of the asset.

Risk in general is a broad subject in financial mathematics but when discussing publicly traded financial instruments risk often refers to the volatility of the instrument, measured as the standard deviation of returns over a period of time. An example of this is the treatment of risk in Markowitz (1952).

Definition 2.13 (Volatility)**.** *The volatility of a financial instrument α_i over T periods is defined as*

$$V(\alpha_i) = \sqrt{\sum_{t=1}^{T} \Big(R_t(\alpha_i) - \mathbb{E}[R(\alpha_i)]\Big)^2}.$$

While the length of periods and total number of periods considered when calculating volatility is not explicit in the definition, different choices can provide different insights and should be made explicit or be clear from the context.

Remark 2.4. *Here it is worth noting a characteristic of financial mathematics that may seem unusual to a reader with a pure mathematics background. Unlike in mathematics, certain parameters are often left deliberately unspecified in financial mathematics. This is most often not an oversight but rather reflects the flexibility finance practitioners have in real-world applications, where parameters are chosen based on specific investment objectives or empirical considerations, rather than on mathematical necessity. This is the source of many limitations but also of creativity in how problems are approached, which will hopefully become clear throughout this paper.*

Another important concept for us to consider is portfolio allocation, which simply put is the way we choose to invest our money. Letting w_t denote our wealth at period t, the allocation $x_i \in \mathbb{R}$ corresponds to the percentage allocation of our wealth w_t to the asset α_i. Taking all allocations together we end up with the column vector

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix},$$

representing our allocation of funds. We note that x_i is unconstrained and can take on both negative values and values greater than 1. While this might not seem intuitive at first the use of short-selling and leverage, or betting against a financial instrument and using borrowed funds to invest, makes this feasible in practice under some constraints. When buying a stock we say we are taking a long position while when betting against a stock we are taking a short position.

We now extend the notation $R_t(\alpha_i)$ to apply also to portfolios of assets, where we let $\boldsymbol{x}_t$ denote the vector containing our portfolio allocation from period t to period $t+1$ and let $\boldsymbol{r}_{t+1}$ denote the return of assets for the period t until $t+1$. The realized return of a portfolio π over the period ending in time $t+1$ will then be $R(\pi) = \boldsymbol{x}_t^T \boldsymbol{r}_{t+1}$ and the evolution of our wealth will be given by $w_{t+1} = (1 + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1}) w_t$.

Definition 2.14 (Covariance matrix)**.** *We define the covariance matrix of N assets as the matrix Σ with entries*

$$\Sigma_{ij} = \operatorname{Cov}[R(\alpha_i), R(\alpha_j)],$$

and note that Σ is symmetric by construction.

Proposition 2.13. *The variance of the returns of a portfolio π over a single period is given by*

$$\operatorname{Var}(\pi) = \boldsymbol{x}^T \Sigma \boldsymbol{x}.$$

Proof. We apply Definition 2.14 and note that

$$\boldsymbol{x}^T \Sigma \boldsymbol{x} = \boldsymbol{x}^T \begin{pmatrix} \sum_{j=1}^N x_j \operatorname{Cov}[R(\alpha_1), R(\alpha_j)] \\ \sum_{j=1}^N x_j \operatorname{Cov}[R(\alpha_2), R(\alpha_j)] \\ \vdots \\ \sum_{j=1}^N x_j \operatorname{Cov}[R(\alpha_N), R(\alpha_j)] \end{pmatrix} = \sum_{i=0}^N \sum_{j=0}^N x_i x_j \operatorname{Cov}[R(\alpha_N), R(\alpha_j)].$$

We separate the variance and covariance terms to get

$$\boldsymbol{x}^T \Sigma \boldsymbol{x} = \sum_{i=0}^N x_i^2 \operatorname{Var}[R(\alpha_i)] + \sum_{i \neq j} x_i x_j \operatorname{Cov}[R(\alpha_i), R(\alpha_j)],$$

which is precisely the variance of a linear combination as shown in Proposition 2.2. □

Definition 2.15 (Risk-free returns)**.** *We say that the return of an asset is a risk-free if the returns are deterministic and thus have no variance.*

Remark 2.5. *In practice there is no such thing as a true risk-free return, but government bonds of developed countries, often the United States, are usually used as a proxy for the risk-free return. We will assume that the risk-free return is unique as not doing so would allow for trivial solutions. To see this we let x be our allocation to the first risk-free asset and let the two risk-free assets have returns $r_1 \neq r_2$. We can then produce any risk-free return r^* by*

$$r^* = x r_1 + (1 - x) r_2 = (r_1 - r_2) x + r_2 \iff x = \frac{r^* - r_2}{r_1 - r_2}.$$

Definition 2.16 (Risky assets)**.** *We consider an asset risky if its returns have a strictly positive variance and a set of assets as collectively risky if we cannot construct a risk-free return by using a linear combination of the assets.*

Proposition 2.14. *For a portfolio of risky assets the covariance matrix Σ is positive definite.*

Proof. Since any portfolio allocation $\boldsymbol{x}$ will create a risky portfolio with a strictly positive variance. By Proposition 2.13 this means $\boldsymbol{x}^T\Sigma\boldsymbol{x} > 0$ for all $\boldsymbol{x} \neq 0$ and Σ is positive definite by Definition 2.1. □

Remark 2.6. *While the covariance matrix Σ is positive definite by our assumption of dealing with risky assets, estimating Σ can be challenging. As described in Ledoit and Wolf (2004, 2012) the sample covariance matrix is often ill-conditioned or even singular, particularly when the number of assets is large relative to the number of observations. This poses a problem for portfolio optimization which often relies on the inversion of Σ. In their paper we find proposed solutions but a closer examination of these goes beyond the scope of this paper. We assume the existence of a well-conditioned and invertible estimate of Σ while acknowledging its estimation is nontrivial in practice.*

Remark 2.7. *The notation $R(\alpha)$ denoting asset returns used in this section will now be dropped as it has served its purpose of helping define the returns vector $\boldsymbol{r}$. We make this explicit to avoid a clash in notation when covering reinforcement learning in Section 5.*

3 Single-Period Mean-Variance Optimization

The foundation of modern portfolio theory was laid in the 1950s with the introduction of the mean-variance framework by Harry Markowitz in Markowitz (1952), where he challenges the prevailing notion that an investor should only consider the discounted expected value of future returns. Instead he lays out an argument for the idea that optimal investment decisions should reflect a trade-off between the expected risk and expected return.(Markowitz, 1952)

The work of Markowitz extends the idea of diversification through providing a mathematical framework that provides an intuitive understanding of optimal choices in portfolio allocation. The model assumes that an investor chooses a portfolio at the beginning of a single period and evaluates the outcome at the end, where asset returns are treated as random variables and risk is given by the variance of portfolio returns.

In this setting Markowitz derives what he calls the "expected returns - variance of returns rule", and the following quote from the beginning of his paper helps put his rule and the problem at hand into context.

> *The process of selecting a portfolio may be divided into two stages. The first stage starts with observation and experience and ends with beliefs about the future performances of available securities. The second stage starts with the relevant beliefs about future performances and ends with the choice of portfolio. This paper is concerned with the second stage.*
>
> — Harry Markowitz, 1952.(Markowitz, 1952)

The same is true for this paper. In this section we formally define the optimization problem, derive closed-form solutions under different assumptions, and examine the limitations and practical challenges of the model, much inspired by Steinbach (2001).

3.1 Constrained Optimization and Lagrange Relaxation

A constrained optimization problem can become much harder to solve than its unconstrained counterpart, but through relaxation we can sometimes solve the difficult problem by a simpler one. Here we will motivate the use of Lagrange relaxation to solve such problems following the lectures and notes of Y.Zhou of Stockholm University for the course *Optimization (MM7028)* in the fall semester of 2024.(Zhou, 2024) Let us consider the general optimization problem

$$(G)\ \begin{cases} \min & f(x) \\ \text{s.t.} & x \in S, \end{cases}$$

and the problem

$$(G_r)\ \begin{cases} \min & f_r(x) \\ \text{s.t.} & x \in S_r, \end{cases}$$

for some functions f and f_r and some sets S and S_r. We call the second problem (G_r) a relaxation of (G) if $S \subseteq S_r$ and $f_r(x) \leq f(x)$ for all $x \in S$, but not necessarily for all $x \in S_r$. We now make the assumption that the optimum is indeed obtained for some point in S, motivated by a desire to maintain a clean formulation of the theory. We note that this will not pose a problem for our use-case since the problems we will encounter are relatively well-behaved.

Theorem 3.1. *Assume that (G_r) is a relaxation of (G). If $\bar{x}_r$ and $\bar{x}$ are optimal solutions of (G_r) and (G) respectively, then*

$$f_r(\bar{x}_r) \leq f(\bar{x}) \leq f(\bar{x}_r).$$

Proof.

$$\begin{aligned} f_r(\bar{x}_r) &= \{\bar{x}_r \text{ is optimum of } (G_r)\} \leq f_r(\bar{x}) \\ &= \{\text{by definition of the relaxation}\} \leq f(\bar{x}) \\ &= \{\bar{x} \text{ is optimum of (G)}\} \leq f(\bar{x}_r) \end{aligned}$$

□

Corollary 3.1. *Assume that (G_r) is a relaxation of (G) and $\bar{x}_r$ is optimum to (G_r). If*

1. $\bar{x}_r \in S$,
2. $f_r(\bar{x}_r) = f(\bar{x}_r)$

Then $\bar{x}_r$ is optimum to (G).

Proof. Let x be an arbitrary feasible point to (G). Then

$$f_r(\bar{x}_r) \leq f_r(x) \leq f(x).$$

But $\bar{x}_r$ is also feasible, showing that $\bar{x}_r$ is optimum to (G), for x is arbitrary. □

Now consider the equality constrained problem

$$(M) \begin{cases} \min & f(x) \\ \text{s.t.} & x \in X \\ & g_i(x) = 0,\ i = 1, \ldots, m, \end{cases}$$

and construct the Lagrange relaxation

$$(M_\lambda) \begin{cases} \min & f(x) + \sum_{i=1}^m \lambda_i g_i(x) \\ \text{s.t.} & x \in X. \end{cases}$$

As lifting the constraint $g_i(x) = 0$ from the feasible set of (M) yields a less restricted set we get $\{x \in X \mid g_i(x) = 0\} \subseteq X$. It is also clear that if x is in the feasible set of (M) then each term $\lambda_i g_i(x) = 0$ so $f(x) + \sum_{i=1}^m \lambda_i g_i(x) \leq f(x)$ for x in the feasible set of (M), in fact equality will hold. Combining these two facts we see that (M_λ) indeed is a relaxation of (M).

Important to note is that the so called Lagrange multipliers λ_i in (M_λ) are not variables but parameters, each giving rise to an optimization problem (M_λ) over the variable x. The objective function of (M_λ) is called the Lagrange function to (M) and is often denoted by $L(x, \lambda)$.

Theorem 3.2 (KKT conditions)**.** *Consider the convex optimization problem*

$$(P) \begin{cases} \min f(\boldsymbol{x}) \\ s.t.\ g_i(\boldsymbol{x}) = 0,\ i = 1, \ldots, m, \end{cases}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex and continuously differentiable, and each $g_i : \mathbb{R}^n \to \mathbb{R}$ is an affine function. Then any feasible solution $\boldsymbol{x}^*$ and $\lambda_i \in \mathbb{R}$ satisfying

1. $\nabla_{\boldsymbol{x}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \nabla_{\boldsymbol{x}} f(\boldsymbol{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla_{\boldsymbol{x}} g_i(\boldsymbol{x}^*) = 0$ *(Stationarity), and*
2. $g_i(\boldsymbol{x}^*) = 0$ *for all* $i = 1, \ldots, m$,

is a globally optimal solution to P. Moreover if f is strictly convex then the minimizer is unique.

Proof. This is a narrow formulation of the KKT conditions adapted from Zhou (2024). For brevity we refer to Bazaraa et al. (2006, Page 207) or other popular sources for a full proof as this is a common result. □

Having introduced Lagrange relaxation and provided the KKT conditions as a sufficient condition for optimality, we are now equipped with the proper tools to address the portfolio optimization problem in the single-period setting.

3.2 Portfolio Optimization with Risky Assets

With the mathematical foundation established we are now ready to approach the portfolio optimization problem. We recall that the goal of mean-variance optimization as presented in Markowitz (1952) is to achieve a balanced trade-off between return and risk, i.e. mean and variance. Inspired by Steinbach (2001) we achieve this by minimizing the variance of our portfolio under the linear constraint that our expected portfolio return $\boldsymbol{x}^T\boldsymbol{\mu}$ equals some target return r^*. Adding the condition that the sum of portfolio weights $\boldsymbol{x}^T\mathbf{1} = 1$ restricts our portfolio to a 100% net investment of the available capital while still allowing for larger gross exposures $\|\boldsymbol{x}\|_1$. This allows for short positions to finance long positions while ensuring we maintain a "net long" position the size of our invested capital. This constraint strikes a good balance between making solutions more realistic while still allowing for an elegant analytical solution.

The choice of constraints for the problem formulation involves a lot of implicit assumptions about how we bridge the gap between theory and practice as well as our investment preferences. We could for example consider a non-negativity constraint on the portfolio allocations which would prohibit short-selling. When combined with the restriction on 100% net investment this would restrict gross investment to 100% ensuring more moderate position sizes. Alternatively we could penalize short positions to reflect real world costs induced by such positions and improve practical relevance. Each modification bridges theory and practice in a different way, but a full discussion about such choices would extend beyond the scope of this paper. We reiterate that our choices aim to provide a balance between practical relevance and mathematical elegance. We now consider the allocation of funds to a set of risky assets under the assumptions we made and construct the problem

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} \quad \text{subject to} \quad \begin{cases} \boldsymbol{x}^T\boldsymbol{\mu} = r^*, \\ \boldsymbol{x}^T\mathbf{1} = 1, \end{cases} \tag{3}$$

a quadratic programming problem with linear constraints. With the assumption of risky assets the matrix Σ is positive definite by Proposition 2.14 and (3) is a convex optimization problem. As Σ is positive definite we can also define the following constants

$$\alpha := \mathbf{1}^T\Sigma^{-1}\mathbf{1}, \quad \beta := \mathbf{1}^T\Sigma^{-1}\boldsymbol{\mu}, \quad \gamma := \boldsymbol{\mu}^T\Sigma^{-1}\boldsymbol{\mu},$$

which will be used to condense notation, inspired by Steinbach (2001).

Proposition 3.1. *The unique primal-dual solution to problem (3) is given by*

$$\boldsymbol{x}^* = -\Sigma^{-1}\left(\frac{\alpha r^* - \beta}{\beta^2 - \alpha\gamma}\boldsymbol{\mu} + \frac{\gamma - \beta r^*}{\beta^2 - \alpha\gamma}\mathbf{1}\right), \qquad \lambda = \frac{\gamma - \beta r^*}{\beta^2 - \alpha\gamma}, \quad \nu = \frac{\alpha r^* - \beta}{\beta^2 - \alpha\gamma}.$$

Proof. Using Lagrange relaxation to solve for the optimal portfolio we lift the constraints into the objective function using the Lagrange multipliers $\nu, \lambda \in \mathbb{R}$ and get

$$\min L(\boldsymbol{x}, \nu, \lambda) = \min \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} + \nu(\boldsymbol{x}^T\boldsymbol{\mu} - r^*) + \lambda(\boldsymbol{x}^T\mathbf{1} - 1).$$

Since $L(\boldsymbol{x}, \nu, \lambda)$ is convex in $\boldsymbol{x}$ the minimum with respect to $\boldsymbol{x}$ is found where

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}, \nu, \lambda) = 0,$$

by Theorem 2.2. Using matrix calculus and the fact that Σ is symmetric we find

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}, \nu, \lambda) = \Sigma\boldsymbol{x} + \nu\boldsymbol{\mu} + \lambda\mathbf{1}.$$

Now, solving for the minimum yields

$$\begin{aligned} \Sigma\boldsymbol{x} + \nu\boldsymbol{\mu} + \lambda\mathbf{1} &= 0, \\ \boldsymbol{x} &= -\Sigma^{-1}(\nu\boldsymbol{\mu} + \lambda\mathbf{1}). \end{aligned}$$

Let this optimal solution be $\boldsymbol{x}^*$. We now seek to ensure that the original constraints are satisfied. We insert the optimal solution it back into the first constraint, and solve for ν to get

$$\begin{aligned}\left(-\Sigma^{-1}(\nu\boldsymbol{\mu}+\lambda\mathbf{1})\right)^T\boldsymbol{\mu} &= r^*\\ -\nu\boldsymbol{\mu}^T\Sigma^{-1}\boldsymbol{\mu}-\lambda\mathbf{1}^T\Sigma^{-1}\boldsymbol{\mu} &= r^*\\ -\nu\gamma-\lambda\beta &= r^*\\ \nu &= -\frac{\lambda\beta+r^*}{\gamma}.\end{aligned}$$

This gives us an updated form of the optimal solution

$$\boldsymbol{x}^* = -\Sigma^{-1}\left(\lambda\mathbf{1}-\frac{\lambda\beta+r^*}{\gamma}\boldsymbol{\mu}\right).$$

Using this form of $\boldsymbol{x}^*$ in the second constraint to solve for λ we get:

$$\begin{aligned}\left(-\Sigma^{-1}\left(\lambda\mathbf{1}-\frac{\lambda\beta+r^*}{\gamma}\boldsymbol{\mu}\right)\right)^T\mathbf{1} &= 1\\ \frac{\lambda\beta+r^*}{\gamma}\boldsymbol{\mu}^T\Sigma^{-1}\mathbf{1}-\lambda\mathbf{1}^T\Sigma^{-1}\mathbf{1} &= 1\\ \frac{\lambda\beta^2+\beta r^*}{\gamma}-\lambda\alpha &= 1\\ \lambda\beta^2+\beta r^*-\lambda\alpha\gamma &= \gamma\\ \lambda &= \frac{\gamma-\beta r^*}{\beta^2-\alpha\gamma}.\end{aligned}$$

This form of λ now solves ν and we find:

$$\nu = -\frac{\frac{\gamma-\beta r^*}{\beta^2-\alpha\gamma}\beta+r^*}{\gamma} = -\frac{\frac{\gamma\beta-\beta^2r^*+\beta^2r^*\alpha\gamma r^*}{\beta^2-\alpha\gamma}}{\gamma} = -\frac{\beta-\alpha r^*}{\beta^2-\alpha\gamma} = \frac{\alpha r^*-\beta}{\beta^2-\alpha\gamma}.$$

Finally the optimal solution is given by

$$\boldsymbol{x}^* = -\Sigma^{-1}(\nu\boldsymbol{\mu}+\lambda\mathbf{1}) = -\Sigma^{-1}\left(\frac{\alpha r^*-\beta}{\beta^2-\alpha\gamma}\boldsymbol{\mu}+\frac{\gamma-\beta r^*}{\beta^2-\alpha\gamma}\mathbf{1}\right),$$

through Theorem 3.2, with Lagrange multipliers

$$\lambda = \frac{\gamma-\beta r^*}{\beta^2-\alpha\gamma}, \quad \nu = \frac{\alpha r^*-\beta}{\beta^2-\alpha\gamma}.$$

□

Clearly we end up with an optimal portfolio that is a linear combination of the two portfolios $\Sigma^{-1}\boldsymbol{\mu}$ and $\Sigma^{-1}\mathbf{1}$. To understand the meaning of these portfolios we consider the two problems

$$\min\frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} \;\text{ s.t. }\; \boldsymbol{x}^T\mathbf{1}=1 \quad\text{and}\quad \max\boldsymbol{\mu}^T\boldsymbol{x} \;\text{ s.t. }\; \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x}=1,$$

minimizing the risk given a full allocation and maximizing the return given for a unit of risk, both of which can be solved using Lagrange relaxation. The first Lagrangian becomes $L(\boldsymbol{x},\lambda)=\frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x}+\lambda\boldsymbol{x}^T\mathbf{1}-\lambda$ which yields

$$\nabla_{\boldsymbol{x}}L = \Sigma\boldsymbol{x}+\lambda\mathbf{1}=0 \iff \boldsymbol{x}^*=\lambda\Sigma^{-1}\mathbf{1}.$$

Solving for λ through the first constraint yields

$$\lambda = \frac{1}{\mathbf{1}^T\Sigma^{-1}\mathbf{1}} \text{ and } \boldsymbol{x}^* = \frac{1}{\mathbf{1}^T\Sigma^{-1}\mathbf{1}}\Sigma^{-1}\mathbf{1}.$$

This is obviously proportional to the $\Sigma^{-1}\mathbf{1}$ component we identified in the solution to Proposition 3.1. Repeating the process for the second problem

$$\max \boldsymbol{\mu}^T \boldsymbol{x} \quad \text{s.t.} \quad \frac{1}{2}\boldsymbol{x}^T \Sigma \boldsymbol{x} = 1,$$

we find the optimal solution as

$$\boldsymbol{x}^* = -\frac{1}{\boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu}} \Sigma^{-1} \boldsymbol{\mu},$$

again proportional to our second component of the solution to Proposition 3.1.

Taken together we can conclude that the solution to the mean-variance optimization comes down to a trade-off between minimizing the risk and maximizing the return. Thus we have recovered the initial intuition behind the approach and revealed its structure in the solution. We also note that the relative emphasis on risk versus reward in the solution is guided by our target return r^* which is our input parameter.

The intuition of a more direct trade-off also extends to other equivalent ways of formulating the problem. For example through the trade-off formulation of the objective function

$$\min \frac{1}{2}\boldsymbol{x}^T \Sigma^{-1} \boldsymbol{x} - \gamma \boldsymbol{\mu}^T \boldsymbol{x},$$

which is equivalent to our choice of objective function under a certain choice of γ as shown in Steinbach (2001). An interesting note from the same paper is Theorem 1.7 of Steinbach (2001) where the author highlights that variance is a quadratic function of the return, the graph of which is called the efficient frontier.

3.3 Risky Assets and Risk-Free Cash

Extending the problem we consider a set of risky assets along with an additional risk-free asset. Investments in this risk-free asset will represent saving cash to yield the risk-free interest rate which we will denote by r^c, and the allocation to the asset by x^c. To avoid degenerate solutions we will assume that $\boldsymbol{\mu} \neq r^c \mathbf{1}$ and note that if our assumption fails to hold then we have the trivial solution of $x^c = 1$.

As the risk-free asset has no variance we let Σ still represent the covariance matrix of the risky assets. Rewriting the constraints and objective function to accommodate the added risk-free asset we formulate the problem

$$\min_{\boldsymbol{x}, x^c} \frac{1}{2} \begin{pmatrix} \boldsymbol{x}^T, & x^c \end{pmatrix} \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ x^c \end{pmatrix}, \quad \text{subject to} \quad \begin{cases} \boldsymbol{x}^T \boldsymbol{\mu} + x^c r^c = r^*, \\ \boldsymbol{x}^T \mathbf{1} + x^c = 1, \end{cases} \tag{4}$$

which again is a convex optimization problem. We can simplify problem (4) to

$$\min_{\boldsymbol{x}, x^c} \frac{1}{2}\boldsymbol{x}^T \Sigma \boldsymbol{x}, \quad \text{subject to} \quad \begin{cases} \boldsymbol{x}^T \boldsymbol{\mu} + x^c r^c = r^*, \\ \boldsymbol{x}^T \mathbf{1} + x^c = 1. \end{cases}$$

We define the expected excess return vector $\hat{\boldsymbol{\mu}} := \boldsymbol{\mu} - r^c \mathbf{1}$ which represents the expected return in excess of the risk-free return.

Proposition 3.2. *The unique primal-dual solution to problem (4) is given by*

$$\boldsymbol{x}^* = -\lambda \Sigma^{-1} \hat{\boldsymbol{\mu}}, \quad x^c = 1 - \lambda \hat{\boldsymbol{\mu}}^T \Sigma^{-1} \mathbf{1}, \quad \lambda = -\frac{r^* - r^c}{\hat{\boldsymbol{\mu}}^T \Sigma^{-1} \hat{\boldsymbol{\mu}}}.$$

Proof. We begin by rewriting the second constraint to get

$$x^c = 1 - \boldsymbol{x}^T \mathbf{1},$$

and use this in the first constraint to get

$$\begin{aligned}\boldsymbol{x}^T\boldsymbol{\mu} + (1 - \boldsymbol{x}^T\mathbf{1})r^c &= r^* \\ \boldsymbol{x}^T\boldsymbol{\mu} - \boldsymbol{x}^T(r^c\mathbf{1}) &= r^* - r^c \\ \boldsymbol{x}^T(\boldsymbol{\mu} - r^c\mathbf{1}) &= r^* - r^c \\ \boldsymbol{x}^T\hat{\boldsymbol{\mu}} &= r^* - r^c.\end{aligned}$$

This has the expected intuitive meaning that we now aim to ensure our expected excess returns meet our excess return target. We form the Lagrangian by lifting this combined constraint into the objective function to get

$$L(\boldsymbol{x}, \lambda) = \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} + \lambda\big(\boldsymbol{x}^T\hat{\boldsymbol{\mu}} - (r^* - r^c)\big).$$

We find the gradient with respect to $\boldsymbol{x}$ and find where it equals zero

$$\begin{aligned}\nabla_{\boldsymbol{x}} L(\boldsymbol{x}, \lambda) = \Sigma\boldsymbol{x} + \lambda\hat{\boldsymbol{\mu}} &= 0 \\ \boldsymbol{x}^* &= -\lambda\Sigma^{-1}\hat{\boldsymbol{\mu}}.\end{aligned}$$

Using $\boldsymbol{x}^*$ in the combined constraint to solve for λ yields

$$\begin{aligned}-\lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}} &= r^* - r^c \\ \lambda &= -\frac{r^* - r^c}{\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}}},\end{aligned}$$

and we have found an optimal solution by Theorem 3.2. The rewritten second constraint in the original problem formulation (4) finally yields

$$x^c = 1 - \boldsymbol{x}^T\mathbf{1} = 1 + \lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\mathbf{1}.$$

All together we have found

$$\boldsymbol{x}^* = -\lambda\Sigma^{-1}\hat{\boldsymbol{\mu}}, \quad x^c = 1 + \lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\mathbf{1}, \quad \lambda = -\frac{r^* - r^c}{\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}}}.$$

□

In contrast to the main results in Section 3.2 our optimal portfolio of risky assets is now proportional only to the portfolio $\Sigma^{-1}\hat{\boldsymbol{\mu}}$. The intuition is that we now have access to the risk-free asset instead of the minimum variance portfolio when we seek to lower the risk of our portfolio, and the risky part of the trade-off is now instead proportional to the maximum excess return portfolio.

Using the optimal solution to now calculate the variance σ^2 of this portfolio we see that

$$\sigma^2 = \boldsymbol{x}^T\Sigma\boldsymbol{x} = \lambda^2\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}},$$

and so the volatility of the portfolio will be $\sigma = |\lambda|\sqrt{\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}}}$. We recall the identity $\boldsymbol{\mu} = r^c\mathbf{1} + \hat{\boldsymbol{\mu}}$ calculate the return ρ of the portfolio we find

$$\begin{aligned}\rho &= r^c x^c + \boldsymbol{x}^T\boldsymbol{\mu} \\ &= r^c(1 + \lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\mathbf{1}) - \lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}(r^c\mathbf{1} + \hat{\boldsymbol{\mu}}) \\ &= r^c + r^c\lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\mathbf{1} - r^c\lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\mathbf{1} - \lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}} \\ &= r^c - \lambda\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}}.\end{aligned}$$

Most of the time we are taking on risk to target a return that is greater than the risk-free rate, and so $r^* > r^c$. Since Σ and therefore also Σ^{-1} are both positive definite we then also have $\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}} > 0$. Combining these

facts we find $\lambda < 0$ for the cases we are interested in. In such cases σ simplifies into $-\lambda\sqrt{\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}}}$ and we can substitute this σ into ρ to find

$$\rho = r^c + \sqrt{\hat{\boldsymbol{\mu}}^T\Sigma^{-1}\hat{\boldsymbol{\mu}}}\ \sigma.$$

What is interesting here is that our solution set is a straight line in risk/reward-space and where on the line we end up is determined by our target return r^*, with the solution set often known as the capital market line.

3.4 Sensitivity to Estimation Errors

In the previous two sections we derived closed-form solutions to two variants of the single-period mean-variance optimization problem. Both solutions depend directly on the expected return vector $\boldsymbol{\mu}$ and the estimated covariance matrix Σ which both represent the investor's belief in the future and are inherently prone to error. While it is common to estimate these parameters from historical data, this is known to generate poor performance as the estimation errors shift allocations.(Lai et al., 2011) Estimating both Σ and $\boldsymbol{\mu}$ is challenging but $\boldsymbol{\mu}$ is particularly unstable in practice, and we will show that this instability leads to high sensitivity in the resulting portfolio allocation.(Merton, 1980)

Naturally it is of interest to investigate just how sensitive the optimal portfolio $\boldsymbol{x}^*$ is to estimation errors, and we begin by recalling the unique dual solution to problem (3). The Lagrange multipliers associated with the return and budget constraints are

$$\lambda = \frac{\gamma - \beta r^*}{\beta^2 - \alpha\gamma}, \quad \nu = \frac{\alpha r^* - \beta}{\beta^2 - \alpha\gamma},$$

and the optimal portfolio is

$$\boldsymbol{x}^* = -\Sigma^{-1}\left(\nu\boldsymbol{\mu} + \lambda\mathbf{1}\right).$$

Because both the direction and magnitude of $\boldsymbol{x}^*$ are influenced by ν and λ, small perturbations in $\boldsymbol{\mu}$ can propagate nonlinearly through λ and ν. Seeing that the estimated covariance matrices Σ are often ill-conditioned Ledoit and Wolf (2012), this can result in large shifts in portfolio weights. If we consider a uniform estimation error of $\varepsilon\mathbf{1}$ in the vector $\boldsymbol{\mu}$ of estimated returns with $\varepsilon \in \mathbb{R}$ we see that

$$\begin{aligned}
\boldsymbol{x}_\varepsilon^* - \boldsymbol{x}^* &= -\Sigma^{-1}\left(\nu(\boldsymbol{\mu} - \varepsilon\mathbf{1}) + \lambda\mathbf{1}\right) - \left(-\Sigma^{-1}\left(\nu\boldsymbol{\mu} + \lambda\mathbf{1}\right)\right) \\
&= -\Sigma^{-1}\nu(\boldsymbol{\mu} - \varepsilon\mathbf{1}) - \Sigma^{-1}\lambda\mathbf{1} + \Sigma^{-1}\left(\nu\boldsymbol{\mu} + \lambda\mathbf{1}\right) \\
&= -\Sigma^{-1}(\nu\boldsymbol{\mu} + \lambda\mathbf{1}) + \Sigma^{-1}\nu\varepsilon\mathbf{1} + \Sigma^{-1}\left(\nu\boldsymbol{\mu} + \lambda\mathbf{1}\right) \\
&= \Sigma^{-1}\nu\varepsilon\mathbf{1},
\end{aligned}$$

showing how estimation errors may become amplified before affecting the optimal allocation through Σ^{-1} and ν, especially when Σ is ill-conditioned. To highlight the sensitivity of the optimal solution to estimation errors we take a look at an example based on real world estimates. We collect stock price data for the American corporations Visa(V), Mastercard(MA), Coca-Cola(KO), and PepsiCo(PEP), chosen qualitatively by the author based on the relatively high correlation in the first and second pair which should create a relatively ill-conditioned covariance matrix and offer interesting results. We collect data for the period 2019-01-01 to 2024-12-31, calculate their sample mean and sample covariance matrix, and annualize these results. We will use these as our expected return and covariance matrix estimates and the results are shown in Table 1a and 1b.

Solving (3) with a target return r^* of 15% and three different choices of inputs parameters should hopefully reveal the sensitivity to estimation errors we expect. The baseline solution will be created using our estimated returns and estimated covariance matrix $\boldsymbol{\mu}$ and Σ and we then solve (3) again where a small perturbation is applied to $\boldsymbol{\mu}$, and one where a small perturbation is applied to Σ. We introduce a 5 percentage point shift to Coca-Cola's metrics and define

$$\Sigma_\varepsilon := \Sigma + 5\% \cdot \boldsymbol{e}_1\boldsymbol{e}_1^T, \text{ and } \boldsymbol{\mu}_\varepsilon := \boldsymbol{\mu} + 5\% \cdot \boldsymbol{e}_1.$$

Table 1: Estimated Parameters for Select Stocks

(a) Estimated Returns

Ticker	Return
Coca-Cola (KO)	9.83%
Mastercard (MA)	21.96%
PepsiCo (PEP)	10.56%
Visa (V)	18.64%

(b) Estimated Covariance Matrix

	KO	MA	PEP	V
KO	4.09%	3.27%	3.12%	2.89%
MA	3.27%	8.77%	3.05%	7.10%
PEP	3.12%	3.05%	4.37%	2.77%
V	2.89%	7.10%	2.77%	7.03%

To test the sensitivity of our optimal solution to perturbations we solve (3) using the original Σ and $\boldsymbol{\mu}$, using the original Σ but the shifted $\boldsymbol{\mu}_\varepsilon$, and one with the shifted covariance matrix Σ_ε. Looking at the results in table 2 we see that a perturbation of 5 percentage points to the estimated mean or variance of Coca-Cola causes a significant shift in the optimal solution.

Table 2: Sensitivity of Optimal Weights to Estimation Errors Measured in Percentage Points

Ticker	Optimal	Δ Mean Shift	Δ Covariance Shift
KO	26.18%	+29.98%	−19.65%
MA	23.46%	−17.88%	−3.27%
PEP	26.16%	−5.28%	+20.10%
V	24.20%	−6.82%	+2.83%

3.5 Mitigating Estimation Errors through Regularization

As discussed in Remark 2.6, estimating Σ is difficult in practice and the sample covariance matrix is often ill-conditioned. The optimal solution $\boldsymbol{x}$ being sensitive to errors made in estimating Σ motivates the use of regularization techniques to stabilize the solution. One such approach presented in Bruder et al. (2013) is to penalize the norm of the allocation vector $\boldsymbol{x}$ in the objective function, discouraging extreme allocations. Using the L^2 norm to penalize the norm of the portfolio allocation vector in (3) of Section 3.2 we get the new regularized mean-variance optimization problem

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} + \frac{\alpha}{2}\|\boldsymbol{x}\|_2^2 \quad \text{subject to} \quad \begin{cases} \boldsymbol{x}^T\boldsymbol{\mu} = r^*, \\ \boldsymbol{x}^T\mathbf{1} = 1, \end{cases} \tag{5}$$

where $\alpha \in \mathbb{R}^+$ controls the strength of the regularization. Rewriting the objective function we note that

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} + \frac{\alpha}{2}\|\boldsymbol{x}\|_2^2 = \min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^T\left(\Sigma + \alpha I\right)\boldsymbol{x}.$$

Introducing the penalization term we have essentially shifted the eigenvalues of the covariance matrix Σ as shown in Proposition 2.8, which in turns shrinks the eigenvalues of Σ^{-1} by Proposition 2.7 and thus shrinks the norm of the optimal solution $\boldsymbol{x}^*$, which depends on Σ^{-1}.

To better understand how this regularization affects the optimization outcome we repeat the sensitivity analysis from Table 2 using (5) and let $\alpha = 0.1$. The results in Table 3 show clearly how regularization helps reduce the impact of estimation errors while giving a similar optimal solution for the baseline case, with almost all allocations remaining within single digit percentage points of the original allocations after the perturbation is applied, a significant improvement over the unregularized case.
Another approach to regularization is to penalize the L^1-norm of the portfolio allocation vector instead of the L^2-norm which gives the problem

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^T\Sigma\boldsymbol{x} + \frac{\alpha}{2}\|\boldsymbol{x}\|_1 \quad \text{subject to} \quad \begin{cases} \boldsymbol{x}^T\boldsymbol{\mu} = r^*, \\ \boldsymbol{x}^T\mathbf{1} = 1, \end{cases} \tag{6}$$

Table 3: Sensitivity of Optimal Weights to Estimation Errors, L^2 Regularized (vs. Original)

Ticker	Optimal	Δ Mean Shift	Δ Covariance Shift
KO	26.18% (26.18%)	+5.79% (+29.98%)	−5.06% (−19.65%)
MA	23.46% (23.46%)	−10.78% (−17.88%)	−0.86% (−3.27%)
PEP	26.16% (26.16%)	+9.04% (−5.28%)	+5.17% (+20.10%)
V	24.20% (24.20%)	−4.06% (−6.82%)	+0.75% (+2.83%)

where $\alpha \in \mathbb{R}^+$ again controls the strength of the regularization. Unlike the L^2-norm which shrinks all weights uniformly, the L^1-norm promotes sparsity in the solution by encouraging weights to be exactly zero. We see the intuition behind this through the example in Figure 1 comparing the L^1, L^2, and L^∞-norms, where solutions constrained by the L^1 norm are more likely to end up on the axes.

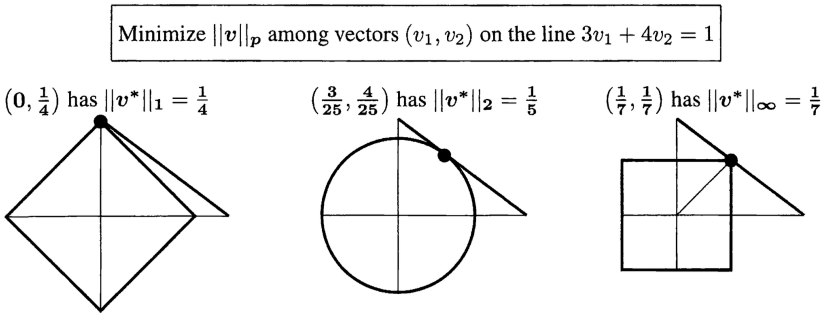

Figure 1: Geometric interpretation of L^1 vs L^2 regularization. Source: Strang (2019, Page 88).

As noted in Bruder et al. (2013), using the L^1-norm serves to improve sparsity and stability of the solution while also having a clear financial interpretation as it corresponds to the leverage of the portfolio, i.e. what multiple of the investor's own money is being put into the investment. While (6) does not have an analytical solution it remains a convex quadratic programming problem and is straight forward to solve using numerical methods.(Bruder et al., 2013) To keep the scope focused on mathematical development of these ideas we omit a full example for the L^1 case.

4 Solving Time-Inconsistency in the Multi-Period Problem

While Section 3 introduced us to the portfolio optimization problem and provided a solid mathematical framework, only considering a single period is a significant restriction. An investor does not exclusively allocate capital for a single period. The only practical limit to how frequently an investor can make decisions and reallocate funds is the speed of the platform they are investing through, which can be on the order of fractions of a second.(Wikipedia, 2025a) It is then natural to propose splitting up the single period into several smaller periods, allowing for decisions to be made throughout the investment horizon. Extending the portfolio optimization problem to a multi-period setting without modification leads to time-inconsistency however. Early works such as Merton (1980) and Samuelson (1969) drop the mean-variance formulation in favor of a utility based objective function, but we shall stay closer to the mean-variance formulation and provide a workaround.

4.1 Dynamic Programming and Bellman's Optimality Principle

Dynamic programming is a method for solving multi-period optimization problems where the core idea is to manage a trade-off between current and future rewards or costs. The optimization problem revolves around choosing some control that influences the state of some system such that the expected future cost or value is minimized or maximized. We will focus on a discrete-time formulation with potentially stochastic controls and stochastic transitions between the states of the system following the lectures and notes of Y.Zhou of Stockholm University for the course *Dynamic systems and optimal control theory (MM7027)* in the spring semester of 2025.(Zhou, 2025). To represent this formally we require a discrete-time set $\{0, 1, \dots, T\}$ containing our time indices t, a state space $\mathcal{X}_t$ containing the possible states at time t, and a control set $\mathcal{U}_{t,x}$ containing the possible controls at time t in state x.

The control $U_t \in \mathcal{U}_{t.x}$ is the possibly stochastic action taken at time t based on the information available at that time, which we denote by I_t. Allowing for stochastic controls essentially means letting the control U_t be determined through a probability density function over controls which we call a policy, instead of being deterministic. A policy is a function of the form

$$K_t(u_t, i_t) = \mathrm{P}(U_t = u_t \mid I_t = i_t).$$

We note that this formulation also includes deterministic policies as a special case where $K_t(u_t, i_t) = 1$ for some $u_t \in \mathcal{U}_{t,x}$. An important note here is that we assume that information is nested such that we remember past information as time goes by, i.e. $I_0 \subseteq I_1 \subseteq \cdots \subseteq I_T$.

To finish the setting we also require some cost or value function $g_t(X_t, U_t)$ for the stochastic state X_t and stochastic control U_t to measure the optimization objective, and assume that for each (i_t, u_t) we know the conditional probability distribution of X_{t+1} given $(I_t = i_t, U_t = u_t)$.

With the formal setting established we can now describe the dynamic programming objective as

$$J^*(i_0) = \min_{K_0,\dots,K_{T-1}} \mathbb{E}\left[\sum_{k=0}^{T-1} g_k(X_k, U_k) + g_T(X_T) \mid I_0 = i_0\right].$$

The value function, or the optimal cost-to-go function, can then be defined as

$$V_t(i_t) := \min_{K_t,\dots,K_{T-1}} \mathbb{E}\left[\sum_{k=t}^{T-1} g_k(X_k, U_k) + g_T(X_T) \mid I_t = i_t\right],$$

so we have $J^*(i_0) = V_0(i_0)$. The general dynamic programming objective $J^*(i_0)$ tells us to minimize the expected costs g_k, while the optimal cost-to-go function has the objective of minimizing the cost remaining from time t. With the setting formalized we are now ready to formally introduce Bellman's optimality principle.

Theorem 4.1 (Bellman's optimality principle)**.** *The value function $V_t(i_t)$ satisfies the recursion*

$$\begin{cases} V_T(i_T) = \mathbb{E}[g_T(X_N) \mid I_T = i_T], \\ V_t(i_t) = \min_u \mathbb{E}[g_t(X_t, U_t) + V_{t+1}(I_{t+1}) \mid I_t = i_t,\ U_t = u], t = 0, 1, \ldots, T-1. \end{cases}$$

There is a deterministic policy that achieves this.

Assumption: *The objective function can be decomposed into stage-wise additive components, allowing for the application of the law of iterated expectations $E[X] = E[E[X|Y]]$, which we will denote by $\circledast$.*

Proof. Define the recursion

$$\begin{cases} W_T(i_T) = \mathbb{E}[g_T(X_T) \mid I_T = i_T], \\ W_t(i_t) = \min_u \mathbb{E}[g_t(X_t, U_t) + W_{t+1}(I_{t+1}) \mid I_t = i_t,\ U_t = u], t = 0, 1, \ldots, T-1. \end{cases}$$

We note that $V_T = W_T$ and assume inductively that this holds for $k = t+1$. We now want to show that $V_t = W_t$. Consider a fixed set of policies $\{K_t, \ldots, K_{T-1}\}$.

$$\begin{aligned} V_t^{K_t,\ldots,K_{T-1}}(i_t) &= \mathbb{E}\left[\sum_{k=t}^{T-1} g_k(X_k, U_k) + g_T(X_T) \mid I_t = i_t\right] \\ &= \mathbb{E}\left[g_t(X_t, U_t) + \sum_{k=t+1}^{T-1} g_k(X_k, U_k) + g_T(X_T) \mid I_t = i_t\right]. \end{aligned}$$

We now use the property $\circledast$ to get

$$\begin{aligned} V_t^{K_t,\ldots,K_{T-1}}(i_t) &= \mathbb{E}\left[g_t(X_t, U_t) + \mathbb{E}\left[\sum_{k=t+1}^{T-1} g_k(X_k, U_k) + g_T(X_T) \;\middle|\; I_{t+1} = i_{t+1}\right] \;\middle|\; I_t = i_t\right] \\ &= \mathbb{E}\left[g_t(X_t, U_t) + V_{t+1}^{K_t,\ldots,K_{T-1}}(i_{t+1}) \;\middle|\; I_t = i_t\right] \\ \text{(I1)} &\geq \mathbb{E}\left[g_t(X_t, U_t) + V_{t+1}(i_{t+1}) \mid I_t = i_t\right] \\ &= \mathbb{E}\left[g_t(X_t, U_t) + W_{t+1}(i_{t+1}) \mid I_t = i_t\right] \\ &= \mathbb{E}\left[\mathbb{E}[g_t(X_t, U_t) + W_{t+1}(i_{t+1}) \mid U_t,\ I_t = i_t] \mid I_t = i_t\right] \quad \text{(Property } \circledast\text{)} \\ &= \sum_u \mathbb{E}[g_t(X_t, U_t) + W_{t+1}(i_{t+1}) \mid U_t,\ I_t = i_t] \cdot \mathrm{P}(U_t = u \mid I_t = i_t) \\ &= \sum_u \mathbb{E}[g_t(X_t, U_t) + W_{t+1}(i_{t+1}) \mid U_t,\ I_t = i_t] \cdot K_t(u, i_t) \\ \text{(I2)} &\geq \min_u \mathbb{E}[g_t(X_t, U_t) + W_{t+1}(i_{t+1}) \mid U_t,\ I_t = i_t] \\ &= W_t(i_t). \end{aligned}$$

Now to collapse inequality I1 we simply choose

$$K_{t+1:T-1} = \operatorname{argmin} V_t^{K_{t+1}:K_{T-1}}(i_{t+1}),$$

which will give us $V_{t+1}(i_{t+1})$ by definition. To collapse inequality I2 we simply choose $K_t = u_t$ for the minimizing u_t and get $W_t(i_t)$ by definition. □

Bellman's optimality principle tells us that an optimal policy has the property that, regardless of what state follows after an action, the remaining sequence of actions will remain optimal in expectation for that new state. This essentially means that what decisions are considered optimal is independent of which particular state we end up in. With the dynamic programming framework introduced in discrete time for stochastic control problems and proven Bellman's optimality principle, we are now ready to turn back to the portfolio optimization problem.

4.2 The Time-Inconsistency of Multi-Period Portfolio Optimization

Through dynamic programming it is now intuitive to extend the single period objective function (3) from Section 3.2 to a dynamic setting over multiple periods. We let w_t denote the wealth at time t starting at $w_0 = 1$, $\boldsymbol{x}_t$ be the portfolio allocation at time t, w^* be our target final wealth, and the system dynamics be dictated by $w_{t+1} = (1 + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1})w_t$. We note that the index of $\boldsymbol{x}_t$ and $\boldsymbol{r}_{t+1}$ differ as $\boldsymbol{x}$ represents a decision made going in to the period and $\boldsymbol{r}_{t+1}$ represents the observed outcome of that period. Now extending (**??**) we end up with the objective of minimizing

$$V_t(w_t, \boldsymbol{x}_t) = \frac{1}{2}\operatorname{Var}_t(w_T) \quad \text{subject to} \quad \begin{cases} \mathbb{E}_t[w_T] = w^*, \\ \boldsymbol{x_t}^T \mathbf{1} = 1. \end{cases}$$

While this formulation looks innocent enough the extension to multiple periods is not trivial. Minimizing variance over several periods introduces time-inconsistency in the optimal solution. Following Björk et al. (2021) this means that the optimal path changes depending on when in time we optimize and we violate Bellman's optimality principle presented in Theorem 4.1. We see why by expanding the variance term using Proposition 2.4 as

$$\operatorname{Var}_t(w_T) = \mathbb{E}_t[w_T^2] - \mathbb{E}_t[w_T]^2.$$

and see that the objective function is nonlinear in the expected value of the terminal wealth. As discussed in Björk et al. (2021, Pages 3-4) this is a common problem which violates the Bellman optimality principle and causes the time-consistency of dynamic programming problems to fail.

Using the conditional variance formula from Proposition 2.5 in the setting of sequential information we also see that

$$\operatorname{Var}_t(w_T) = \mathbb{E}_t[\operatorname{Var}(w_T)] + \operatorname{Var}_t(\mathbb{E}_{t+1}[w_T]),$$

where the second term $\operatorname{Var}_t(\mathbb{E}[w_T \mid I_{t+1}])$ represents the variance of our future expectation on the terminal wealth w_T. In the setting of financial returns this component is a function of the realized returns up to time t and of the portfolio weights chosen prior to t. We can convince ourselves that the second term will be strictly positive, as assuming $\operatorname{Var}_t(\mathbb{E}_{t+1}[w_T]) = 0$ would imply each period being void of uncertainty. As the naive extension of the mean-variance framework to a multi-period setting ends up violating Bellman's optimality principle breaking time consistency we are forced to reconsider our approach.

4.3 Constant Relative Risk Aversion

To overcome the the issue of time-inconsistency in the mean-variance formulation Merton (1969) and Samuelson (1969) employ the notion of constant relative risk aversion, where an investor remains willing to risk a certain percentage of their wealth regardless of their current wealth. Equivalently, their appetite for risk expresses itself in relative instead of absolute terms and is measured as a percentage of their total wealth. This behavior gives rise to the concave utility function

$$U(W) = \begin{cases} \frac{W^{1-\gamma}}{1-\gamma}, & \text{for } \gamma > 0, \gamma \neq 1 \\ \ln(W), & \text{for } \gamma = 1, \end{cases} \tag{7}$$

where γ represents the investor's degree of risk-aversion. While we no longer explicitly punish the variance of returns as we did in the mean-variance formulation, variance in outcome is implicitly punished through Jensen's inequality. We make this clear using an example. Suppose $\mathbb{E}[W] = \mu$, then Jensen's inequality of Theorem 2.5 gives

$$\mathbb{E}[U(W)] \leq U(\mathbb{E}[W]) = U(\mu)$$

since U is convex. That is, a terminal wealth of μ achieved with certainty will be preferred over an expected terminal wealth of μ with some variance.

Using this framework, Merton and Samuelson derive analytical and time-consistent solutions to the portfolio optimization problem in continuous and discrete time. While the solutions are elegant they take us away

from our original and intuitive mean-variance formulation, favoring the utility based objective function. To keep closer to the original mean-variance objective function presented in Section 3 we shall instead find another way around time-inconsistency.

4.4 Mean-Variance Optimization with Partial Pre-Commitment

Another method to get around the problem of time-inconsistency is to introduce a partial pre-commitment strategy inspired by Huang et al. (2022), where we fix the expected terminal wealth $\mathbb{E}[w_T]$ at time $t = 0$ and stick to this estimate throughout the investment horizon. Through this partial pre-commitment we avoid the issue of chasing a moving target in the form of an evolving $\mathbb{E}[w_T]$ and we can form a time-consistent objective function. We will first show that this formulation remains close to our naive extension of the mean-variance framework that turned out time-inconsistent and recall that our terminal wealth will depend on our choices of $\boldsymbol{x}_t$.

Proposition 4.1. *The mean-variance portfolio optimization problem*

$$\min_{\boldsymbol{x}_0,\dots,\boldsymbol{x}_{T-1}} \operatorname{Var}(w_T) \quad \textit{subject to} \quad \begin{cases} \mathbb{E}[w_T] = w^*, \\ \boldsymbol{x}^T\mathbf{1} = 1, \ t = 0, 1, \dots, T-1, \end{cases} \tag{8}$$

can be extracted from the problem

$$\min_{\boldsymbol{x}_0,\dots,\boldsymbol{x}_{T-1}} \mathbb{E}[(w_T - \lambda)^2] - (\lambda - w^*)^2 \quad \textit{subject to} \quad \boldsymbol{x}_t^T\mathbf{1} = 1, \ t = 0, 1, \dots, T-1, \tag{9}$$

at $t = 0$ in the sense that they yield the same solution $\boldsymbol{x}_0, \dots, \boldsymbol{x}_{T-1}$ at that time if λ can be chosen such that $w^* = \mathbb{E}_{t=0}[w_T]$.

Proof. We begin by expanding the objective function of (9) and find

$$\begin{aligned} \mathbb{E}[(w_T - \lambda)^2] - (\lambda - w^*)^2 &= \mathbb{E}[w_T^2 - 2\lambda w_T + \lambda^2] - (\lambda^2 - 2\lambda w^* + (w^*)^2) \\ &= \mathbb{E}[w_T^2] - 2\lambda\mathbb{E}[w_T] + \lambda^2 - \lambda^2 + 2\lambda w^* - (w^*)^2 \\ &= \mathbb{E}[w_T^2] - 2\lambda\mathbb{E}[w_T] + 2\lambda w^* - (w^*)^2. \end{aligned}$$

Now choosing λ such that $\mathbb{E}[w_T] = w^*$ yields

$$\begin{aligned} \mathbb{E}[(w_T - \lambda)^2] - (\lambda - w^*)^2 &= \mathbb{E}[w_T^2] - 2\lambda\mathbb{E}[w_T] + 2\lambda\mathbb{E}[w_T] - \mathbb{E}[w_T]^2 \\ &= \mathbb{E}[w_T^2] - \mathbb{E}[w_T]^2 \\ &= \operatorname{Var}(w_T). \end{aligned}$$

We have achieved the same objective function as (8), the first of its constraints holds by our choice of λ, and the second constraint is also present in (9) and so we have recovered the mean-variance problem formulation. □

By Proposition 4.1 we have found a problem that is equivalent to our desired problem formulation at $t = 0$ but that we shall show is time-consistent. To this end we recall the vector based definition of the covariance-matrix

$$\Sigma_t = \mathbb{E}[(\boldsymbol{r}_{t+1} - \boldsymbol{\mu}_t)(\boldsymbol{r}_{t+1} - \boldsymbol{\mu}_t)^T]$$

and the relationship

$$\mathbb{E}[\boldsymbol{r}_{t+1}\boldsymbol{r}_{t+1}^T] = \Sigma_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^T.$$

We also note that Σ_t is positive definite by Proposition 2.14 while $\boldsymbol{\mu}_t\boldsymbol{\mu}_t^T$ is positive semi-definite by Proposition 2.6, so it follows that $Q_t := (\Sigma_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^T)$ is positive definite and invertible. We again introduce the coefficients

$$\zeta_t := \mathbf{1}^T Q_t^{-1}\mathbf{1}, \quad \beta_t := \boldsymbol{\mu}_t^T Q_t^{-1}\mathbf{1}, \quad \delta_t := \boldsymbol{\mu}_t^T Q_t^{-1}\boldsymbol{\mu}_t,$$

to condense notation as in Section 3.

Proposition 4.2 (A time-consistent solution)**.** *The dynamic programming problem*

$$\min_{\boldsymbol{x}_0,\dots,\boldsymbol{x}_{T-1}} \mathbb{E}[(w_T-\lambda)^2]-(\lambda-w^*)^2 \quad \textit{subject to} \quad \boldsymbol{x}_t^T\mathbf{1}=1 \textit{ for } t=0,1,\dots,T-1, \tag{10}$$

is time-consistent with a unique optimal feedback

$$\boldsymbol{x}_t^*(w)=Q^{-1}\left(\left(\frac{b_{t+1}\beta_t}{2a_{t+1}w\zeta_t}+\frac{1+\beta_t}{\zeta_t}\right)\mathbf{1}-\left(1+\frac{b_{t+1}}{2a_{t+1}w}\right)\boldsymbol{\mu}_t\right).$$

The problem's value-function is the quadratic function

$$V_t(w)=a_tw^2+b_tw+c_t.$$

where the coefficients satisfy the recursive relationship

$$\begin{cases} a_t=a_{t+1}+\frac{a_{t+1}\beta_t+a_{t+1}}{\zeta_t}, \\ b_t=b_{t+1}+\frac{b_{t+1}\beta_t^2+2b_{t+1}\beta_t-b_{t+1}\delta_t\zeta_t}{2\zeta_t}, \\ c_t=c_{t+1}+\frac{b_{t+1}^2\beta_t^2-b_{t+1}^2\delta_t\zeta_t}{4a_{t+1}\zeta_t}, \end{cases}$$

with the terminal state yielding

$$a_T=1, \quad b_T=-2\lambda, \quad c_T=2\lambda w^*-(w^*)^2.$$

Proof. As our proof contains some heavy algebra we omit a few steps here and refer to Appendix 7.2 for the full proof. We proceed with a proof by induction, where we verify the terminal case, make an ansatz that the value function is quadratic at $t+1$, and then show that this still holds at t. For $t=0,1,\dots,T$ and current wealth $w\geq 0$ we define the value-function

$$V_t(w)=\min_{\boldsymbol{x}_t,\dots,\boldsymbol{x}_{T-1}} \mathbb{E}\left[(w_T-\lambda)^2 \mid w_t=w\right]-(\lambda-w^*)^2.$$

At the final period $t=T$ no more decisions need to be made so

$$V_T(w)=(w-\lambda)^2-(\lambda-w^*)^2=w^2-2\lambda w+\left(2\lambda w^*-(w^*)^2\right).$$

We now aim to show by backward induction that the value function is indeed quadratic as at time $t=T$ for all time-steps, such that

$$V_t(w)=a_tw^2+b_tw+c_t, \tag{11}$$

and that the optimal feedback x_t satisfies the Bellman equation. Now for $t<T$ the Bellman principle gives

$$V_t(w)=\min_{\boldsymbol{x}_t}\mathbb{E}[V_{t+1}(w) \mid w_t=w] \quad \text{subject to} \quad \boldsymbol{x}^T\mathbf{1}=1.$$

Now substituting the wealth dynamics $w_{t+1}=w_t(1+\boldsymbol{x}_t^T\boldsymbol{r}_{t+1})$ and the quadratic form (16) into the Bellman equation we get

$$\begin{aligned} V_t(w) &= \min_{\boldsymbol{x}_t \text{ s.t } \boldsymbol{x}^T\mathbf{1}=1} a_{t+1}w^2\mathbb{E}[(1+\boldsymbol{x}_t^T\boldsymbol{r}_{t+1})^2]+b_{t+1}w\mathbb{E}[1+\boldsymbol{x}_t^T\boldsymbol{r}_{t+1}]+c_{t+1} \\ &= \min_{\boldsymbol{x}_t \text{ s.t } \boldsymbol{x}^T\mathbf{1}=1} a_{t+1}w^2\mathbb{E}[2\boldsymbol{x}_t^T\boldsymbol{r}_{t+1}+\boldsymbol{x}_t^T\boldsymbol{r}_{t+1}\boldsymbol{r}_{t+1}^T\boldsymbol{x}_t]+b_{t+1}w\mathbb{E}[\boldsymbol{x}_t^T\boldsymbol{r}_{t+1}]+(a_{t+1}w^2+b_{t+1}w+c_{t+1}) \\ &= \min_{\boldsymbol{x}_t \text{ s.t } \boldsymbol{x}^T\mathbf{1}=1} a_{t+1}w^2\boldsymbol{x}_t^TQ_t\boldsymbol{x}_t+(2a_{t+1}w^2+b_{t+1}w)\boldsymbol{x}_t^T\boldsymbol{\mu}+(a_{t+1}w^2+b_{t+1}w+c_{t+1}). \end{aligned}$$

We form the Lagrangian using γ as the Lagrange multiplier for $\boldsymbol{x}_t^T\mathbf{1}=1$ to get

$$\mathcal{L}(\boldsymbol{x},\gamma)=a_{t+1}w^2\boldsymbol{x}_t^TQ_t\boldsymbol{x}_t+(2a_{t+1}w^2+b_{t+1}w)\boldsymbol{x}_t^T\boldsymbol{\mu}+c_{t+1}+\gamma(\boldsymbol{x}_t^T\mathbf{1}-1).$$

Through the KKT conditions we now get

$$\begin{cases} \nabla_{\boldsymbol{x}_t} : 2a_{t+1}w^2 Q\boldsymbol{x}_t + (2a_{t+1}w^2 + b_{t+1}w)\boldsymbol{\mu}_t + \gamma\mathbf{1} = 0, \text{ and} \\ \partial_\gamma : \boldsymbol{x}_t^T\mathbf{1} = 1. \end{cases}$$

As $Q \succ 0$ the first condition gives

$$\boldsymbol{x}_t^*(w) = -Q^{-1}\left(\frac{2a_{t+1}w^2 + b_{t+1}w}{2a_{t+1}w^2}\boldsymbol{\mu}_t + \frac{\gamma}{2a_{t+1}w^2}\mathbf{1}\right).$$

Solving for γ we insert this optimal solution into the second condition and find

$$\gamma = -\left(\frac{2a_{t+1}w^2(1+\beta_t) + b_{t+1}w\beta_t}{\zeta_t}\right),$$

which after some algebra gives the optimal control

$$\boldsymbol{x}_t^*(w) = Q^{-1}\left(\left(\frac{b_{t+1}\beta_t}{2a_{t+1}w\zeta_t} + \frac{1+\beta_t}{\zeta_t}\right)\mathbf{1} - \left(1 + \frac{b_{t+1}}{2a_{t+1}w}\right)\boldsymbol{\mu}_t\right).$$

This optimal control is unique as $Q \succ 0$. Now to calculate the recursive coefficients a_t, b_t, and c_t we will employ the optimal control in the value-function. After some algebra we collect the terms by w^2, w, and constants and get

$$V_t(w) = w^2\left(a_{t+1} + \frac{a_{t+1}\beta_t + a_{t+1}}{\zeta_t}\right) + w\left(b_{t+1} + \frac{b_{t+1}\beta_t^2 + 2b_{t+1}\beta_t - b_{t+1}\delta_t\zeta_t}{2\zeta_t}\right) + \left(c_{t+1} + \frac{b_{t+1}^2\beta_t^2 - b_{t+1}^2\delta_t\zeta_t}{4a_{t+1}\zeta_t}\right).$$

This yields the recursive formula

$$\begin{cases} a_t = a_{t+1} + \frac{a_{t+1}\beta_t + a_{t+1}}{\zeta_t}, \\ b_t = b_{t+1} + \frac{b_{t+1}\beta_t^2 + 2b_{t+1}\beta_t - b_{t+1}\delta_t\zeta_t}{2\zeta_t}, \\ c_t = c_{t+1} + \frac{b_{t+1}^2\beta_t^2 - b_{t+1}^2\delta_t\zeta_t}{4a_{t+1}\zeta_t}, \end{cases}$$

where the terminal value-function $V_T(w) = w^2 - 2\lambda w + \left(2\lambda w^* - (w^*)^2\right)$ yields

$$a_T = 1, \quad b_T = -2\lambda, \quad c_T = 2\lambda w^* - (w^*)^2.$$

By induction we have shown that the value function remains quadratic and have found the unique optimizer at each step. □

While we have managed to formulate a time-consistent mean-variance problem over several periods and even find an analytical solution, we are again faced with the problem of our optimal solution being heavily reliant on parameter estimation. What makes problem (10) time consistent is that we minimize the deviation of terminal wealth from a predetermined target level λ that we fix at $t = 0$. This is in contrast to simply minimizing the variance of terminal wealth where the reference point of deviation measurement, $\mathbb{E}[w_T]$, is continuously being updated. While this provides the time-consistency we sought it also places a lot of importance in the initial estimation of λ which adds to the parameter estimation problem.

5 Reinforcement Learning

Throughout the preceding chapters we have discussed analytical solutions to the portfolio optimization problem in a single and multi-period setting. We explored the mean-variance analysis introduced by Harry Markowitz and extended this framework to a dynamic multi-period problem, where we dealt with the issue of time-inconsistency. While this allowed us to derive analytical solutions, the probability distributions of asset returns we have taken for granted are difficult to estimate in practice. Through a sensitivity analysis we saw that our solutions are sensitive to perturbations in these estimates and the solutions can therefore become fragile. The natural question then arises; can we solve this problem even when we don't know the underlying probabilities?

Reinforcement learning provides precisely such a framework. As summarized in Sutton and Barto (2018, Chapter 1.6) reinforcement learning involves understanding and decision-making without requiring complete models of the environment. The aim of this chapter is to reinterpret the portfolio optimization problem from Section 4.4 through the lens of reinforcement learning. We do this by interpreting the problem as a Markov decision process, defining the relevant state and action spaces, and reformulating the objective function as a reward function. By doing this we bridge the gap between the analytical model-based optimization with this modern model-free learning approach, skirting the need for difficult parameter estimation and hopefully obtaining better and more robust solutions.

5.1 Portfolio Optimization as a Markov Decision Process

To apply reinforcement learning techniques to the portfolio optimization problem we must first formalize the problem within the reinforcement learning framework of Markov decision processes, where we follow the standard formulation in Sutton and Barto (2018, Chapter 3.1-3) and Szepesvári (2010, Chapter 1.2). We can define a Markov decision process by the tuple

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}),$$

where $\mathcal{S}$ and $\mathcal{A}$ are the sets of possible states and actions, $\mathcal{P}$ is the transition probability kernel assigning state transition probabilities conditional on the current state and action (s, a). When $\mathcal{S}$ and $\mathcal{A}$ are both finite we say this is a finite Markov decision process.

To the tuple $\mathcal{M}$ we can associate a reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ which gives the expected immediate reward when taking action a in state s. The reward itself is a random variable R so our reward function becomes

$$r(s, a) = \mathbb{E}[R \mid S = s, A = a],$$

to which we can associate a discount factor $\gamma \in [0, 1]$. If we fix some policy for how actions are taken the return of that policy is given by

$$\mathcal{R} = \sum_{t=0}^{\infty} \gamma^t R_{t+1}.$$

This of course assumes that the decision process continues forever, but if we assume that the process stops then we say it is a finite-horizon Markov decision process. Furthermore it is called discounted or undiscounted depending on whether $\gamma < 1$ or $\gamma = 1$. From here on out we will assume that R is bounded and note that if the Markov decision process is discounted then so is $\mathcal{R}$.

For our portfolio optimization problem we will consider a finite-horizon but undiscounted Markov decision process. To match the setting of our multi-period problem in Section 4 we let the state be given by the current wealth $w_t \in \mathbb{R}_+$ and time $t \in \{0, 1, \ldots, T\}$ which yields

$$\mathcal{S} := \mathbb{R}_+ \times \{0, 1, \ldots, T\}.$$

The action we are able to take at each time t is choosing some valid portfolio allocation $\boldsymbol{x}_t$ from the set

$$\Delta^n := \{\boldsymbol{x}_t \in \mathbb{R}^n \mid \boldsymbol{x}_t^T \mathbf{1} = 1\}.$$

This reflects the constraint that our full wealth must be allocated at each step as we have done throughout the previous problem formulations. One can restrict the set of valid portfolios further by for example prohibiting short-selling. Our action space simply becomes

$$\mathcal{A} := \Delta^n.$$

The state transition dynamics will be determined by the realized returns vector and our current wealth through the equation

$$w_{t+1} = w_t(1 + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1}),$$

where the return vectors $\boldsymbol{r}_t$ are drawn from some unknown probability distribution. This gives us the formal state transition probabilities as

$$P\left(W' \mid W, \boldsymbol{x}\right) := P\left(W' = W(1 + \boldsymbol{x}^T \boldsymbol{r})\right),$$

While the current formulation of states, actions, and rewards is not finite in favor of a clean presentation, we shall now make them finite through discretization. We do this by rounding weights, wealth, and the reward to a fixed number of decimal points. This is motivated by practical limitations, where wealth and portfolio allocations are ultimately measured in some currency which will have a smallest increment, and where measures of variance are necessarily rounded to the precision of floating point numbers in digital machines. This discretization both brings our theoretical foundation closer to the real-world setting we seek to apply our methods to, and the finite setting matches the one where we shall prove the existence of optimal policies.

For the reward function we look to the objective function in our time consistent multi-period problem (10) from Section 4.4, which provides a sparse reward only generated at the terminal time step T. Since the objective is to minimize (10) we multiply the objective by -1 to fit the terminology of positive rewards and construct the reward function

$$R_t := \begin{cases} 0, & \text{if } t < T, \\ -(w_T - \lambda)^2 & \text{if } t = T. \end{cases} \tag{12}$$

We remind ourselves that (10) involved minimizing the deviation of our final wealth from some predetermined target λ, which we treat as an input parameter.

5.2 Policies, Value Functions, and the Optimization Objective

Having formulated the multi-period problem as a Markov decision process we are now ready to approach the problem of finding optimal portfolio allocations, and in the context of reinforcement learning this involves finding an optimal policy. Following Sutton and Barto (2018, Chapter 3.5) we define a value function $V : \mathcal{S} \to \mathbb{R}$ with an associated policy as the expected return as measured from some timestep t following that policy. A policy maps a state to probabilities of selecting each possible action in that state, where we call a policy that maps a state directly to an action deterministic. For a finite-horizon Markov decision process we formally define the value function following the policy π as

$$V_\pi(s) := \mathbb{E}_\pi \left[\sum_{k=t+1}^{T} \gamma^{t-k+1} R_k \,\middle|\, S_t = s \right], \tag{13}$$

where $\mathbb{E}_\pi[\cdot]$ is the expected value given that the policy π is followed. We can extend this to define the closely related action-value function as

$$Q_\pi(s, a) := \mathbb{E}_\pi \left[\sum_{k=t+1}^{T} \gamma^{t-k+1} R_k \,\middle|\, S_t = s,\ A_t = a \right],$$

where we measure the expected value of taking the action a from state s. For a Markov decision process with an infinite horizon we simply let the above sums run to infinity.

Naturally we want to maximize the value generated by our policy, and solving a reinforcement learning task involves finding such a policy. Following along chapter 3.6 of Sutton and Barto (2018) we can define the notion of good and bad policies by defining an ordering of policies through the value function.

Definition 5.1. *We say that the policy π_1 is better than policy π_2, denoted by $\pi_1 \geq \pi_2$, if the value of a state derived using π_1 is greater than or equal to the value of a state derived using π_2 for every state in the environment, i.e.*

$$V_{\pi_1}(s) \geq V_{\pi_2}(s), \ \forall s \in \mathcal{S}.$$

Definition 5.2 (Bellman optimality operator)**.** *We consider a finite state space $\mathcal{S}$ and a finite action space $\mathcal{A}$. Now consider the set of all value functions $X := \{V : \mathcal{S} \to \mathbb{R} \mid \|V\|_\infty < \infty\}$, where $\|V\|_\infty = \max_{s\in\mathcal{S}} |V(s)|$, and define the Bellman optimality operator $B : X \to X$ for any $V \in X$ as*

$$BV(s) := \max_{a\in\mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a) V(s') \right\},$$

where we recall that $r(s,a)$ is the expected return taking action a in state s, γ is the discount rate for future returns, and $\mathrm{P}(s' \mid s,a)$ denotes the transition probability from state s to s' given that action a is taken.

As defined, we see that B is a recursive operator and thus generates a sequence of value functions. We now define the finite metric space (X, d_∞) with X as in Definition 5.2 and

$$d_\infty(V_1, V_2) = \|V_1 - V_2\|_\infty = \max_{s\in\mathcal{S}} |V_1(s) - V_2(s)|.$$

Lemma 5.1. *The metric space (X, d_∞) is complete.*

Proof. We let $\{V_n\}_{n=0}^\infty$ be any Cauchy sequence in (X, d_∞) and fix any $s \in \mathcal{S}$. The sequence $\{V_n(s)\}_{n=0}^\infty$ will then be a Cauchy sequence over the real numbers. Since $\mathbb{R}$ is complete we have

$$V(s) := \lim_{n\to\infty} V_n(s)$$

defining a function $V : \mathcal{S} \to \mathbb{R}$. Since $\{V_n\}$ is Cauchy $\|V(s)\|_\infty$ will be bounded and therefore $V(s)$ will lie in X. Now for any $\varepsilon > 0$ there exists some N such that for all $m, n \geq N$

$$\|V_m - V_n\|_\infty < \varepsilon.$$

This means

$$\max_{s\in\mathcal{S}} \big|V_m(s) - V_n(s)\big| < \varepsilon.$$

Now consider any $n \geq N$ and each $s \in \mathcal{S}$,

$$\big|V_n(s) - V(s)\big| = \big|V_n(s) - \lim_{m\to\infty} V_m(s)\big| = \lim_{m\to\infty} \big|V_n(s) - V_m(s)\big|,$$

where we pass the limit outside the absolute value thanks to continuity of $|\cdot|$. Taking the maximum over s yields

$$\|V_n - V\|_\infty < \varepsilon$$

and so $\{V_n\}$ converges to V. Any Cauchy sequence in (X, d_∞) converges to V which is in X, and so (X, d_∞) is complete. □

Theorem 5.1 (The Bellman operator is a contraction)**.** *The Bellman optimality operator B is a contraction mapping the finite metric space (X, d_∞).*

Proof. Let V_1 and V_2 be two value functions and assume a discount rate $\gamma \in [0, 1)$. We then have by definition

$$|BV_1(s) - BV_2(s)| = \left| \max_{a\in\mathcal{A}} \Big(r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a) V_1(s') \Big) - \max_{a\in\mathcal{A}} \Big(r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a) V_2(s') \Big) \right|.$$

From 4.1 of Schmidt (2016) we get the inequality

$$\begin{aligned}|BV_1(s) - BV_2(s)| &\leq \max_{a\in\mathcal{A}} \left| r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)V_1(s') - r(s,a) - \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)V_2(s') \right| \\ &= \max_{a\in\mathcal{A}} \left| \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)V_1(s') - \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)V_2(s') \right| \\ &= \gamma \max_{a\in\mathcal{A}} \left| \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)\big(V_1(s') - V_2(s')\big) \right|,\end{aligned}$$

which in turn gives us the inequality

$$|BV_1(s) - BV_2(s)| \leq \gamma \max_{a\in\mathcal{A}} \left| \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)\big(V_1(s') - V_2(s')\big) \right|. \tag{14}$$

Since we are working with a weighted sum we have for any $a \in \mathcal{A}$ that

$$\gamma \left| \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)\big(V_1(s') - V_2(s')\big) \right| \leq \gamma \max_{s\in\mathcal{S}} |V_1(s) - V_2(s)| = \gamma \|V_1 - V_2\|_\infty. \tag{15}$$

Putting (14) and (15) together finally yields

$$\|BV_1 - BV_2\|_\infty = \max_{s\in\mathcal{S}} |BV_1(s) - BV_2(s)| \leq \gamma \|V_1 - V_2\|_\infty.$$

If $\gamma \in [0,1)$ then B is a contraction on the complete metric space (X, d_∞). □

Corollary 5.1 (An optimal policy exists)**.** *For any finite Markov decision process with $\gamma \in [0,1)$ and bounded rewards there exists an optimal policy π^* such that it is better than or equal to every other possible policy π.*

Proof. By Theorem 2.6 the contraction B has a unique fixed point $V^* \in X$ and the iteration $V_{k+1} = BV_k$ will converge to V^*. We find an optimal policy π^*, which is not necessarily unique, by

$$\pi^*(s) := \arg\max_{a\in\mathcal{A}} r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathrm{P}(s' \mid s,a)V^*(s').$$

□

While we have shown the existence of an optimal policy for finite Markov decision processes with an infinite time-horizon through Corollary 5.1, our problem formulation has a finite horizon. We continue by showing similar results hold for finite horizons without the need for a discount factor.

Definition 5.3 (The time-indexed Bellman operator)**.** *We recall the Bellman optimality operator from Definition 5.2 and attach a time index $t \in \mathbb{N}$ to define*

$$(B_t V)(s) := \max_a r_{t+1}(s,a) + \gamma \sum_{s'} P(s' \mid s,a)V(s').$$

Along the same lines we can attach a time-index t to our value function, denoting the fact that we are evaluating at time t to emphasize that we only have $T - t$ steps left.

Proposition 5.1 (An optimal policy exists for a finite horizon problem)**.** *For any finite Markov decision process with a finite horizon, discount rate $\gamma \in [0,1]$, and bounded rewards there exists an optimal policy π^* such that it is better than or equal to every other possible policy π.*

Proof. As the Markov decision process terminates at $t = T$ we begin by noting that the expected future return will be $V_T^* = 0$. We now assume inductively that V_{t+1}^* is uniquely determined and let

$$V_t^*(s) = (B_t V_{t+1}^*)(s) = \max_a \left\{ r_{t+1}(s,a) + \gamma \sum_{s'} P(s' \mid s,a) V_{t+1}^*(s') \right\}.$$

As $\mathcal{A}$ and $\mathcal{S}$ are finite this maximum is attained and V_t^* is also uniquely determined. By induction we will have a sequence of uniquely determined optimal value functions $\{V_t^*\}_{t=0}^T$ from which we can extract an optimal sequence of policies $\{\pi_t^*\}_{t=0}^T$ defined by

$$\pi_t^*(s) \in \arg\max_a \left\{ r_{t+1}(s,a) + \gamma \sum_{s'} P(s' \mid s,a) V_{t+1}^*(s') \right\}.$$

Following the policies π_t^* from step $t = 0$ will guarantee we achieve V_t^* at each step. We also note that we converge to this optimal solution in T steps. □

Having proven the existence of an optimal policy for both infinite and finite-horizon Markov decision processes we have guaranteed that our portfolio optimization problem is solvable using reinforcement learning. Looking back to our reward function (12) from Section 5.1 we note that it is only non-zero at time T, and the value function simplifies to

$$V_{t,\pi}(s) = \mathbb{E}_\pi \left[-(w_T - \lambda)^2 \mid S_t = s \right].$$

The action-value function also simplifies to

$$Q_{t,\pi}(s,a) = \mathbb{E}_\pi \left[-(w_T - \lambda)^2 \mid S_t = s,\ A_t = a \right].$$

Since finding an optimal policy involves maximizing the expected return it generates we arrive at solving the problem

$$\pi^* = \arg\max_\pi \ \mathbb{E}_\pi \left[-(w_T - \lambda)^2 \right].$$

As our problem has a finite horizon we end up with finding a sequence of optimal policies $\{\pi_t^*\}_{t=0}^T$ maximizing the corresponding value function from $\{v_t^*\}_{t=0}^T$. The actions prescribed by the optimal policy are portfolio allocations $\boldsymbol{x}_t$, and by multiplying the return with -1 we can rewrite the problem as

$$\min_{\boldsymbol{x}_0,\ldots,\boldsymbol{x}_{T-1}} \mathbb{E}\left[(w_T - \lambda)^2 \right] \text{ subject to } \boldsymbol{x}_t \in \Delta^n \text{ for } t = 0, 1, \ldots, T-1.$$

This is precisely our time-consistent dynamic programming formulation (10) which we have successfully transferred to the reinforcement learning setting. Instead of solving the problem analytically based on estimates of probability distributions, solutions which we have seen can be sensitive to estimation errors, we can now find optimal allocations through trial and error without relying on knowing the underlying probability distributions of asset returns.

5.3 Learning from Experience and Finding an Optimal Policy

A concern when finding an optimal solution through dynamic programming is the reliance on estimates of the probability distribution of returns. We saw in Remark 2.6 that finding such estimates is nontrivial, and in section 3.4 we saw that errors in estimates can degrade the quality of our solution.

An alternative is to estimate an optimal solution from collected experience. As described in Sutton and Barto (2018, Chapter 6) temporal difference learning provides such a method, combining learning from experience as seen in Monte-Carlo methods with continually updating estimates as seen in dynamic programming, all without the need for a model of the dynamics of the environment. Temporal difference learning focuses on finding the value function v_π for a given policy π through sequentially making and refining estimates V of the true value function v_π. The general idea is to evaluate the estimate of V at time t against the actual reward R_{t+1} observed the next step along with the current estimate of the remaining reward. The simplest

temporal difference method TD(0) collects an experience tuple (S_t, R_{t+1}, S_{t+1}) representing a sequence of state, reward, and state, and then makes the update

$$V_{new}(S_t) \leftarrow V(S_t) + \alpha\Big[\big(R_{t+1} + \gamma V(S_{t+1})\big) - V(S_t)\Big],$$

where γ is the discount rate and α the rate at which we update our estimate.(Sutton and Barto, 2018, Page 120) As shown in Jaakkola et al. (1993) the tabular version of this algorithm will indeed converge with probability 1 under some moderate assumptions outlined in the paper as well as in Sutton and Barto (2018, Chapter 6.2).

Extending TD(0) to instead estimate the action-value function $q_\pi(s, a)$ for a current policy π we can find a function associating actions to estimated rewards, from which we can generate a policy which learns using temporal difference. Letting Q represent our current estimate this leads to the algorithm

$$Q_{new}(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\Big[\big(R_{t+1} + \gamma Q(S_{t+1}, A_{t+1})\big) - Q(S_t, A_t)\Big],$$

with γ and α as before.(Sutton and Barto, 2018, Page 129) This method is known as Sarsa, stemming from the sequence of state, action, reward, state, action that it collects and learns from. Sarsa is what is known as an on-policy method where we continually improve upon the policy used to generate new data providing an instant feedback loop between experience and learning. In contrast, off-policy methods such as DQN(Mnih et al., 2013) instead collect a large buffer of past experience, possibly generated by a now outdated policy, and still learns from old and new experience alike. Actual implementations of reinforcement learning algorithms also rely on some trade-off between exploration and exploitation, for example by balancing random actions with greedy actions, i.e. where $\arg\max_a Q(s, a)$ is always chosen, which is known as ε-greedy exploration. Exploration is essential to allow the algorithm time to collect a diverse set of experiences to help pick out a policy which effectively solves the problem at hand. On the other hand, exploring too aggressively may prohibit the agent from actually learning positive behaviors. While ε-greedy is the common starting point to implement exploration more sophisticated methods are steadily proposed, for example the Bayesian approach of Rozanov (2024) or the curiosity driven approach proposed in Pathak et al. (2017).

As we get deeper into the discussion of reinforcement learning we may naturally start to wonder about the true effectiveness of these algorithms. Indeed the tabular version of the Sarsa algorithm does converge with probability one under some moderate assumptions and given sufficient exploration, where all state-action pairs are visited an infinite amount of times, and that the policy converges to the greedy policy.(Sutton and Barto, 2018; Singh et al., 2000) Beyond theoretical convergence guarantees we also have a plethora of promising results in the literature, each of which brings a unique perspective that helps us understand the developments in and promise of the field as a whole. The early work of Mnih et al. (2013) introduced the DQN algorithm and successfully learned to play Atari games using deep learning, with the later work Hessel et al. (2017) compiling several improvements to the original algorithm. Building off of the success of what became known as Deep Q-learning Lillicrap et al. (2019) developed the off-policy algorithm DDPG, compatible with continuous action spaces and which adopted the so called actor-critic framework where the responsibility of action selection and value estimation are separated. The TRPO algorithm developed around the same time is an on-policy alternative also adopting the actor-critic framework which offered some theoretical guarantees, and which also gave rise to the PPO algorithm of Schulman et al. (2017), which remains a very popular algorithm today.

By now it is clear that reinforcement learning can provide a powerful model-free framework to sequential decision-making problems under uncertainty. To explore this further we translated the portfolio optimization problem to the reinforcement learning setting by re-framing it as a Markov decision process and introducing temporal difference methods. Providing simple algorithms with convergence guarantees under reasonable assumptions and outlining the vast progress made in the field in recent years we take the first step towards finding a competitive solution to the portfolio allocation problem using reinforcement learning. While there is a wide range of algorithms and techniques to explore we limit ourselves to laying the groundwork, and point to Sutton and Barto (2018) for a more thorough treatment of the subject as a whole. We instead

turn to a review of empirical studies that showcase the effectiveness of reinforcement learning based portfolio optimization strategies.

5.4 The Effectiveness of Reinforcement Learning for Portfolio Optimization

Recent work in reinforcement learning suggests that this data-driven approach can outperform static allocation rules in portfolio optimization. While numerical simulations and test on real-world market data extend beyond the scope of this paper, we shall explore these recent results. These studies span a broad set of asset classes, algorithms, and time-frames and all show encouraging results that reinforcement learning may be an effective tool for portfolio allocation. Common benchmarks for the evaluating the performance of algorithms are fixed allocations, such as setting all weights to an equal value, or naive strategies that follow a simple heuristic rule, e.g. investing in the stocks that performed best over some recent period.

Huang et al. (2022) construct a novel reinforcement learning algorithm based from the well known algorithms SAC(Haarnoja et al., 2018) and A2C(Mnih et al., 2016) that beats the performance of the market index, the plug-in estimation method we set forth in section 4, and the well-known reinforcement learning algorithms PPO(Schulman et al., 2017) and DDPG(Lillicrap et al., 2019). The paper implements the methods on a subset of the US stock market and covers both daily and monthly trading frequencies, indicating the method may be robust across different time resolutions.

Constructing a multi-reward approach, where several agents are trained under different rewards and then combined, Choudhary et al. (2025) also finds strong results relative to benchmarks. They find that their framework outperforms mean-variance optimization, equal weighting of instruments, and the actual underlying stock across four different markets.

Other studies explore more exotic markets and a faster pace of trading. In Jiang et al. (2017) the authors construct complex reinforcement learning algorithms using deep learning and manage to achieve a strong performance relative to naive benchmarks and previous methods. They do this trading on a subset of 12 cryptocurrencies and using a trading frequency of 30 minutes which provides further evidence that reinforcement learning methods may be effective in portfolio optimization. To this end we also look to a comprehensive study of different algorithms and techniques in Espiga-Fernández et al. (2024) with more promising results, where several algorithms beat the benchmark performance.

Taken together these results demonstrate the capacity of reinforcement learning to learn adaptive portfolio allocation rules that can beat classical methods. At the same time the practical challenges we have encountered, particularly the difficulty of estimating the input parameters to classical methods, mean that the intersection of reinforcement learning and portfolio optimization remains a promising and open field of research.

6 Conclusion

In this paper we have explored portfolio optimization in discrete time, starting at the traditional mean-variance formulation over a single period, which we extended to a multi-period setting, finally leading us to explore reinforcement learning as a modern alternative. We began by introducing the mean-variance problem in a single-period setting as a quadratic programming problem with linear constraints, solvable through standard optimization techniques. While elegant, investigating the analytical solutions and performing numerical tests we demonstrated that the solutions have a high sensitivity to estimation errors in the input parameters.

To tackle the sensitivity to estimation errors we examined regularization techniques based on penalizing the L^1 and L^2-norm of the portfolio allocation vector. We discussed the merits of such techniques and showed through numerical examples that they indeed can reduce sensitivity to estimation errors, providing a potential solution to the fragility of the analytical solutions.

We extended the single-period framework to a multi-period setting by allowing for reallocation decisions to be made throughout the investment horizon. This uncovered a time-inconsistency problem when extending the traditional mean-variance formulation to several periods. Describing the nature of this inconsistency we proposed an alternative time-consistent problem formulation based on partial pre-commitment. We showed that this alternative formulation is identical to the original formulation at the initial time-step and derived an analytical solution using dynamic programming. While this pre-commitment formulation was solvable it still suffered from the need of estimating input parameters, prompting a further search for solutions

In the final part of this paper we introduced reinforcement learning as an alternative approach that bypasses the need for an explicit model of probability distributions, with the hope of avoiding at least some of the issues previously encountered. We reinterpreted the multi-period problem as a Markov decision process and introduced the terminology of reinforcement learning. We proved that an optimal solution to our reformulated problem exists, introduced simple algorithms with convergence guarantees, and covered some more recent algorithms in reinforcement learning to set the stage. Reviewing a set of empirical studies that employ reinforcement learning to real-world financial data we then show that this indeed is a promising solution to the problem, finding encouraging results across several algorithms, markets, and time-frames.

While the results in this paper are largely theoretical, they showcase both the elegance and limitations of traditional methods in portfolio optimization. By extending these traditional methods using modern model-free techniques we provide a way forward in spite of these many limitations. Rather than being an entirely linear work, this paper aims to provide a map of problems, techniques, and limitations. To offer clarity while sparking curiosity by both answering questions that arise and raising unanswered ones that the reader may pursue independently. It is also intended to reveal the beautiful fractal richness that can emerge in both finance and mathematics.

References

Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley-Interscience, 3rd edition, 2006. ISBN 978-0471486008.

Tomas Björk, Mariana Khapko, and Agatha Murgoci. *Time-Inconsistent Control Theory with Finance Applications*. Springer Cham, 1st edition, 2021. ISBN 978-3-030-81842-5. doi: https://doi.org/10.1007/978-3-030-81843-2.

Benjamin Bruder, Nicolas Gaussel, Jean-Charles Richard, and Thierry Roncalli. Regularization of portfolio allocation. https://ssrn.com/abstract=2767358, June 2013. SSRN working paper.

Himanshu Choudhary, Arishi Orra, Kartik Sahoo, and Manoj Thakur. Risk-adjusted deep reinforcement learning for portfolio optimization: A multi-reward approach. *International Journal of Computational Intelligence Systems*, 18(1):126, 2025.

Francisco Espiga-Fernández, Álvaro García-Sánchez, and Joaquín Ordieres-Meré. A systematic approach to portfolio optimization: A comparative study of reinforcement learning agents, market signals, and investment horizons. *Algorithms*, 17(12), 2024. ISSN 1999-4893. doi: 10.3390/a17120570. URL https://www.mdpi.com/1999-4893/17/12/570.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, 2018. URL https://arxiv.org/abs/1801.01290.

Ankur Handa. Matrix calculus: Useful identities for derivatives. https://www.doc.ic.ac.uk/~ahanda/referencepdfs/MatrixCalculus.pdf, 2011. Accessed: 2025-07-22.

Matteo Hessel, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning, 2017. URL https://arxiv.org/abs/1710.02298.

Yilie Huang, Yanwei Jia, and Xunyu Zhou. Achieving mean–variance efficiency by continuous-time reinforcement learning. In *Proceedings of the Third ACM International Conference on AI in Finance*, ICAIF '22, page 377–385, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393768. doi: 10.1145/3533271.3561760. URL https://doi.org/10.1145/3533271.3561760.

Tommi Jaakkola, Michael I. Jordan, and Satinder P. Singh. Convergence of stochastic iterative dynamic programming algorithms. In *Proceedings of the 7th International Conference on Neural Information Processing Systems*, NIPS'93, page 703–710, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.

Zhengyao Jiang, Dixing Xu, and Jinjun Liang. A deep reinforcement learning framework for the financial portfolio management problem, 2017. URL https://arxiv.org/abs/1706.10059.

Tze Leung Lai, Haipeng Xing, and Zehao Chen. Mean–variance portfolio optimization when means and covariances are unknown. *The Annals of Applied Statistics*, 5(2A), June 2011. ISSN 1932-6157. doi: 10.1214/10-aoas422. URL http://dx.doi.org/10.1214/10-AOAS422.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004. ISSN 0047-259X. doi: https://doi.org/10.1016/S0047-259X(03)00096-4. URL https://www.sciencedirect.com/science/article/pii/S0047259X03000964.

Olivier Ledoit and Michael Wolf. Nonlinear shrinkage estimation of large-dimensional covariance matrices. *The Annals of Statistics*, 40(2), April 2012. ISSN 0090-5364. doi: 10.1214/12-aos989. URL http://dx.doi.org/10.1214/12-AOS989.

Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning, 2019. URL https://arxiv.org/abs/1509.02971.

Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. doi: https://doi.org/10.2307/2975974.

Robert C. Merton. Lifetime portfolio selection under uncertainty: The continuous-time case. *The Review of Economics and Statistics*, 51(3):247–257, 1969. ISSN 00346535, 15309142. URL http://www.jstor.org/stable/1926560.

Robert C. Merton. On estimating the expected return on the market: An exploratory investigation. *Journal of Financial Economics*, 8(4):323–361, 1980. ISSN 0304-405X. doi: https://doi.org/10.1016/0304-405X(80)90007-0. URL https://www.sciencedirect.com/science/article/pii/0304405X80900070.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013. URL http://arxiv.org/abs/1312.5602.

Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning, 2016. URL https://arxiv.org/abs/1602.01783.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction, 2017. URL https://arxiv.org/abs/1705.05363.

Sheldon Ross. *First Course in Probability, A, Global Edition*. Pearson Education, 10th edition, 2019. ISBN 9781292269238. URL https://books.google.se/books?id=4P07ygEACAAJ.

Nikolai Rozanov. Efficient exploration in deep reinforcement learning: A novel bayesian actor-critic algorithm, 2024. URL https://arxiv.org/abs/2408.10055.

Walter Rudin. *Principles of mathematical analysis*. McGraw-Hill, New York, 3rd edition, 1976. ISBN 9780070856134.

Paul A. Samuelson. Lifetime portfolio selection by dynamic stochastic programming. *The Review of Economics and Statistics*, 51(3):239–246, 1969. ISSN 00346535, 15309142. URL http://www.jstor.org/stable/1926559.

Mark Schmidt. Argmax and max calculus, 2016. URL https://www.cs.ubc.ca/~schmidtm/Courses/Notes/max.pdf. Accessed: 2025-07-29.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Satinder Singh, Tommi Jaakkola, Michael L. Littman, and Csaba Szepesvári. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine Learning*, 38(3):287–308, 2000. doi: 10.1023/A:1007678930559. URL https://doi.org/10.1023/A:1007678930559.

Marc C. Steinbach. Markowitz revisited: Mean-variance models in financial portfolio analysis. *SIAM Review*, 43(1):31–85, 2001. doi: https://doi.org/10.1137/S0036144500376650.

Gilbert Strang. *Linear Algebra and Learning from Data*. Wellesley-Cambridge Press, 2019. ISBN 9780692196380.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 2nd edition, 2018.

Csaba Szepesvári. *Algorithms for Reinforcement Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Springer Cham, 1st edition, 2010. ISBN 978-3-031-00423-0. doi: 10.1007/978-3-031-01551-9. URL https://doi.org/10.1007/978-3-031-01551-9. Springer Nature Switzerland AG, eBook ISBN: 978-3-031-01551-9, Published: 31 May 2022.

Wikipedia. High-frequency trading, 2025a. URL https://en.wikipedia.org/wiki/High-frequency_trading. Accessed: 2025-07-09.

Wikipedia. Modern portfolio theory. https://en.wikipedia.org/wiki/Modern_portfolio_theory, 2025b. Accessed: 2025-06-16.

Yishao Zhou. Lecture notes for optimization (mm7028). Lecture notes, distributed privately, 2024. Unpublished lecture notes, Stockholm University.

Yishao Zhou. Lecture notes for dynamic systems and optimal control theory (mm7027). Lecture notes, distributed privately, 2025. Unpublished lecture notes, Stockholm University.

7 Appendix

7.1 Code for Markowitz Sensitivity Analysis

```python
import yfinance as yf
import pandas as pd
import numpy as np
from scipy.optimize import minimize

# --- PARAMETERS ---
TICKERS = ["V", "MA", "KO", "PEP"]
START_DATE = "2019-01-01"
END_DATE = "2024-12-31"
TRADING_DAYS = 252
TARGET_RETURN = 0.15
ESTIMATION_ERROR = 0.05
ALPHA = 2.0  # Regularization term for L2 penalty

ALPHA = ALPHA * 0.5 # Adjust alpha to be in line with problem formulation

# --- DOWNLOAD DATA ---
def download_price_data(tickers, start, end):
    data = yf.download(tickers, start=start, end=end, interval="1d")
    return data["Close"].dropna(axis=1)

try:
    prices = pd.read_csv("sensitivity_analysis_prices.csv", index_col=0, parse_dates=True)
except (FileNotFoundError, ValueError):
    print("Downloading price data...")
    prices = download_price_data(TICKERS, START_DATE, END_DATE)
    prices.to_csv("sensitivity_analysis_prices.csv")
    print("Data saved to sensitivity_analysis_prices.csv.")

# --- CALCULATE RETURNS ---
returns = prices.pct_change().dropna()
mean_returns = returns.mean() * TRADING_DAYS
cov_matrix = returns.cov() * TRADING_DAYS

# --- OPTIMIZATION FUNCTION ---
def markowitz(mu, Sigma, target_return, alpha=0.0):
    Sigma_reg = Sigma + alpha * np.eye(len(Sigma))  # L2 regularization

    def objective(x): return 0.5 * x.T @ Sigma_reg @ x
    def constraint_sum(x): return np.sum(x) - 1
    def constraint_return(x): return x @ mu - target_return

    constraints = [
        {"type": "eq", "fun": constraint_sum},
        {"type": "eq", "fun": constraint_return}
    ]
    x0 = np.ones(len(mu)) / len(mu)
    result = minimize(objective, x0, constraints=constraints, method="SLSQP")

    if not result.success:
        return None

    x_opt = result.x
    realized_return = x_opt @ mu
    realized_risk = np.sqrt(x_opt.T @ Sigma @ x_opt)  # Use unregularized for actual risk
    return x_opt, realized_return, realized_risk

# --- SHIFTED ESTIMATES ---
mu_shift = mean_returns + ESTIMATION_ERROR * np.array([1, 0, 0, 0])
Sigma_shift = cov_matrix + ESTIMATION_ERROR * np.diag([1, 0, 0, 0])

# --- SOLVE ALL CASES ---
sol_base = markowitz(mean_returns, cov_matrix, TARGET_RETURN)
```

```python
sol_mu_shift = markowitz(mu_shift, cov_matrix, TARGET_RETURN)
sol_Sigma_shift = markowitz(mean_returns, Sigma_shift, TARGET_RETURN)

Sigma_reg = cov_matrix + ALPHA * np.eye(len(cov_matrix))
Sigma_shift_reg = Sigma_shift + ALPHA * np.eye(len(cov_matrix))

sol_reg = markowitz(mean_returns, Sigma_reg, TARGET_RETURN, alpha=ALPHA)
sol_mu_shift_reg = markowitz(mu_shift, Sigma_reg, TARGET_RETURN, alpha=ALPHA)
sol_Sigma_shift_reg = markowitz(mean_returns, Sigma_shift_reg, TARGET_RETURN, alpha=ALPHA)

# --- SENSITIVITY TABLE ---
def print_sensitivity_table(tickers, base, mu_shifted, Sigma_shifted, base_reg,
    mu_shifted_unreg, Sigma_shifted_unreg):
    print("\n--- Relative Sensitivity of Weights to Estimation Error (Regularized vs
        Unregularized) ---")
    header = f"{'Ticker':<10}{'Optimal':>12}{'$Mean Shift':>20}{'Cov Shift':>20}"
    print(header)
    print("-" * len(header))
    for i, ticker in enumerate(tickers):
        base_w = base[i]
        delta_mu = mu_shifted_unreg[i] - base_w
        delta_Sigma = Sigma_shifted_unreg[i] - base_w
        reg_w = base_reg[i]
        delta_mu_reg = mu_shifted[i] - reg_w
        delta_Sigma_reg = Sigma_shifted[i] - reg_w

        line = f"{ticker:<10}{reg_w*100:12.2f}%({base_w*100:.2f}%){delta_mu_reg*100:+12.2f}%
            ({delta_mu*100:+.2f}%)" \
              f"{delta_Sigma_reg*100:+12.2f}% ({delta_Sigma*100:+.2f}%)"
        print(line)

# --- DISPLAY RESULTS ---
tickers = mean_returns.index.tolist()
print_sensitivity_table(
    tickers,
    sol_base[0], sol_mu_shift_reg[0], sol_Sigma_shift_reg[0],
    sol_reg[0], sol_mu_shift[0], sol_Sigma_shift[0]
)
```

Listing 1: Portfolio Optimization with Sensitivity Analysis

7.2 Full proof for the time-consistent multi-period solution

Proof. We proceed with a proof by induction, where we verify the terminal case, make an ansatz that the value function is quadratic at $t+1$, and then show that this still holds at t. For $t = 0, 1, \ldots, T$ and current wealth $w \geq 0$ we define the value-function

$$V_t(w) = \min_{\boldsymbol{x}_t, \ldots, \boldsymbol{x}_{T-1}} \mathbb{E}\left[(w_T - \lambda)^2 \mid w_t = w\right] - (\lambda - w^*)^2.$$

At the final period $t = T$ no more decisions need to be made so

$$V_T(w) = (w - \lambda)^2 - (\lambda - w^*)^2 = w^2 - 2\lambda w + \left(2\lambda w^* - (w^*)^2\right).$$

We now aim to show by backward induction that the value function is indeed quadratic as at time $t = T$ for all time-steps, such that

$$V_t(w) = a_t w^2 + b_t w + c_t, \tag{16}$$

and that the optimal feedback x_t satisfies the Bellman equation. Now for $t < T$ the Bellman principle gives

$$V_t(w) = \min_{\boldsymbol{x}_t} \mathbb{E}[V_{t+1}(w) \mid w_t = w] \quad \text{subject to} \quad \boldsymbol{x}^T \mathbf{1} = 1.$$

Now substituting the wealth dynamics $w_{t+1} = w_t(1 + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1})$ and the quadratic form (16) into the Bellman equation we get

$$\begin{aligned}
V_t(w) &= \min_{\boldsymbol{x}_t \text{ s.t } \boldsymbol{x}^T \mathbf{1} = 1} a_{t+1} w^2 \mathbb{E}[(1 + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1})^2] + b_{t+1} w \mathbb{E}[1 + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1}] + c_{t+1} \\
&= \min_{\boldsymbol{x}_t \text{ s.t } \boldsymbol{x}^T \mathbf{1} = 1} a_{t+1} w^2 \mathbb{E}[2\boldsymbol{x}_t^T \boldsymbol{r}_{t+1} + \boldsymbol{x}_t^T \boldsymbol{r}_{t+1} \boldsymbol{r}_{t+1}^T \boldsymbol{x}_t] + b_{t+1} w \mathbb{E}[\boldsymbol{x}_t^T \boldsymbol{r}_{t+1}] + (a_{t+1} w^2 + b_{t+1} w + c_{t+1}) \\
&= \min_{\boldsymbol{x}_t \text{ s.t } \boldsymbol{x}^T \mathbf{1} = 1} a_{t+1} w^2 \boldsymbol{x}_t^T Q_t \boldsymbol{x}_t + (2a_{t+1} w^2 + b_{t+1} w) \boldsymbol{x}_t^T \boldsymbol{\mu} + (a_{t+1} w^2 + b_{t+1} w + c_{t+1}).
\end{aligned}$$

We form the Lagrangian using γ as the Lagrange multiplier for $\boldsymbol{x}_t^T \mathbf{1} = 1$ to get

$$\mathcal{L}(\boldsymbol{x}, \gamma) = a_{t+1} w^2 \boldsymbol{x}_t^T Q_t \boldsymbol{x}_t + (2a_{t+1} w^2 + b_{t+1} w) \boldsymbol{x}_t^T \boldsymbol{\mu} + c_{t+1} + \gamma(\boldsymbol{x}_t^T \mathbf{1} - 1).$$

Through the KKT conditions we now get

$$\begin{cases} \nabla_{\boldsymbol{x}_t} : 2a_{t+1} w^2 Q \boldsymbol{x}_t + (2a_{t+1} w^2 + b_{t+1} w) \boldsymbol{\mu}_t + \gamma \mathbf{1} = 0, \text{ and} \\ \partial_\gamma : \boldsymbol{x}_t^T \mathbf{1} = 1. \end{cases}$$

As $Q \succ 0$ the first condition gives

$$\boldsymbol{x}_t^*(w) = -Q^{-1} \left(\frac{2a_{t+1} w^2 + b_{t+1} w}{2a_{t+1} w^2} \boldsymbol{\mu}_t + \frac{\gamma}{2a_{t+1} w^2} \mathbf{1} \right).$$

Solving for γ we insert this optimal solution into the second condition and find

$$\begin{aligned}
1 &= -\frac{2a_{t+1} w^2 + b_{t+1} w}{2a_{t+1} w^2} \boldsymbol{\mu}_t^T Q^{-1} \mathbf{1} - \frac{\gamma}{2a_{t+1} w^2} \mathbf{1}^T Q^{-1} \mathbf{1} \\
\frac{\gamma}{2a_{t+1} w^2} \mathbf{1}^T Q^{-1} \mathbf{1} &= -\frac{2a_{t+1} w^2 + b_{t+1} w}{2a_{t+1} w^2} \boldsymbol{\mu}_t^T Q^{-1} \mathbf{1} - 1 \\
\gamma &= -\left(\frac{2a_{t+1} w^2 (\boldsymbol{\mu}_t^T Q^{-1} \mathbf{1}) + b_{t+1} w (\boldsymbol{\mu}_t^T Q^{-1} \mathbf{1}) + 2a_{t+1} w^2}{\mathbf{1}^T Q^{-1} \mathbf{1}} \right) \\
\gamma &= -\left(\frac{2a_{t+1} w^2 (1 + \beta_t) + b_{t+1} w \beta_t}{\zeta_t} \right)
\end{aligned}$$

After some algebra, this finally gives the optimal control

$$\boldsymbol{x}_t^*(w) = Q^{-1}\left(\left(\frac{b_{t+1}\beta_t}{2a_{t+1}w\zeta_t} + \frac{1+\beta_t}{\zeta_t}\right)\mathbf{1} - \left(1 + \frac{b_{t+1}}{2a_{t+1}w}\right)\boldsymbol{\mu}_t\right).$$

This optimal control is unique as $Q \succ 0$. Now to calculate the recursive coefficients a_t, b_t, and c_t we will employ the optimal control in the value-function and after some algebra arrive at

$$\begin{aligned}
V_t(w) = a_{t+1}w^2\Bigg[&\left(\frac{b_{t+1}^2\beta_t^2}{4a_{t+1}^2w^2\zeta_t^2} + \frac{1+2\beta_t+\beta_t^2}{\zeta_t^2} + \frac{b_{t+1}\beta_t(1+\beta_t)}{a_{t+1}w\zeta_t^2}\right)\zeta_t \\
&- 2\left(\frac{b_{t+1}^2\beta_t}{4a_{t+1}^2w^2\zeta_t} + \frac{b_{t+1}(1+\beta_t)}{2a_{t+1}w\zeta_t} + \frac{b_{t+1}\beta_t}{2a_{t+1}w\zeta_t} + \frac{1+\beta_t}{\zeta_t}\right)\beta_t \\
&+ \left(1 + \frac{b_{t+1}}{a_{t+1}w} + \frac{b_{t+1}^2}{4a_{t+1}^2w^2}\right)\delta_t\Bigg] \\
&+ (a_{t+1}w^2 + b_{t+1}w)\left[\left(\frac{b_{t+1}\beta_t}{2a_{t+1}w\zeta_t} + \frac{1+\beta}{\zeta_t}\right)\beta_t - \left(1 + \frac{b_{t+1}}{2a_{t+1}w}\right)\delta_t\right] \\
&+ (a_{t+1}w^2 + b_{t+1}w + c_{t+1}).
\end{aligned}$$

We simplify slightly to get

$$\begin{aligned}
V_t(w) = \Bigg[&\frac{\big(2a_{t+1}w(1+\beta) + b_{t+1}\beta_t\big)^2}{4a_{t+1}\zeta_t} \\
&- 2\left(\frac{\beta_t(2a_{t+1}w + b_{t+1})\big(2a_{t+1}w(1+\beta_t) + b_{t+1}\beta_t\big)\big)}{4a_{t+1}\zeta_t}\right) \\
&+ \frac{\delta_t(2a_{t+1}w + b_{t+1})^2}{4a_{t+1}}\Bigg] \\
&+ \left[\frac{\beta_t(a_{t+1}w + b_{t+1})\big(2a_{t+1}w(1+\beta_t) + b_{t+1}\beta_t\big)}{2a_{t+1}\zeta_t} - \frac{\delta_t(a_{t+1}w + b)(2a_{t+1}w + b)}{2a_{t+1}}\right] \\
&+ (a_{t+1}w^2 + b_{t+1}w + c_{t+1}).
\end{aligned}$$

Collecting the terms by w^2, w, and constants we get

$$V_t(w) = w^2\left(a_{t+1} + \frac{a_{t+1}\beta_t + a_{t+1}}{\zeta_t}\right) + w\left(b_{t+1} + \frac{b_{t+1}\beta_t^2 + 2b_{t+1}\beta_t - b_{t+1}\delta_t\zeta_t}{2\zeta_t}\right) + \left(c_{t+1} + \frac{b_{t+1}^2\beta_t^2 - b_{t+1}^2\delta_t\zeta_t}{4a_{t+1}\zeta_t}\right).$$

This yields the recursive formula

$$\begin{cases}
a_t = a_{t+1} + \frac{a_{t+1}\beta_t + a_{t+1}}{\zeta_t}, \\
b_t = b_{t+1} + \frac{b_{t+1}\beta_t^2 + 2b_{t+1}\beta_t - b_{t+1}\delta_t\zeta_t}{2\zeta_t}, \\
c_t = c_{t+1} + \frac{b_{t+1}^2\beta_t^2 - b_{t+1}^2\delta_t\zeta_t}{4a_{t+1}\zeta_t},
\end{cases}$$

where the terminal value-function $V_T(w) = w^2 - 2\lambda w + \big(2\lambda w^* - (w^*)^2\big)$ yields

$$a_T = 1, \quad b_T = -2\lambda, \quad c_T = 2\lambda w^* - (w^*)^2.$$

By induction we have shown that the value function remains quadratic and have found the unique optimizer at each step. □