B.Sc. Dissertation

Robo-advising: automated quantitative algorithms for risk profile estimation and portfolio optimization

Yiqiu Liu

Supervised by: Prof Chao Zhou

National University of Singapore
March 2023

Thesis submitted to
Department of Mathematics, Faculty of Science
in partial fulfillment of the requirements for
Bachelor of Science (Honours) in Quantitative Finance

Academic Year 2022/2023

Abstract[1]

This is an abstract

Keywords:

robo-advising, portforlio optimization

[1]This abstract serves the purpose of the report summary required by the Department.

Acknowledgements

This is acknowledgement.

Contents

Chapter 1

Introduction

1.1 Background

1.2 Motivation and challenges

1.3 Contributions

1.4 Report organization

Chapter 2

Literature review

2.1 Preliminaries on robo-advisors

The inception of robo-advisors dates back to the early 2000s when the first online investment platforms, such as Betterment and Wealthfront, came to the fore. These platforms leveraged algorithms to determine asset allocation and rebalancing strategies, providing automated investment advice and portfolio management services to individual investors at a low cost. With the advancement of technology, traditional financial institutions started offering robo-advisory services by developing their own platforms or collaborating with existing FinTech firms.

The financial crisis of 2008 was a turning point in investment management. The crisis led to a loss of trust in traditional financial institutions, prompting investors to explore alternative investment strategies. This environment facilitated the emergence of robo-advising, which offers investors an automated investment platform that uses quantitative algorithms to manage their portfolios (Capponi, Olafsson, & Zariphopoulou, 2022). This platform is easily accessible to clients online, allowing investors to create and manage their portfolios. The robo-advisor analyzes the investor's risk tolerance, investment goals, and financial situation using algorithms to develop a personalized investment plan for the investor (Beketov, Lehmann, & Wittke, 2018). The platform then uses these algorithms to execute trades and manage the investor's portfolio on an ongoing basis.

The robo-advisory industry is relatively new, but it has grown rapidly since its inception in response to the increasing demand from investors for more transparent and low-cost investment services. Using robo-advisors provides several benefits to financial institutions and investors alike. One of the main benefits of using robo-advisors is that they save on fixed costs for financial institutions. The traditional model of investment advisory services involves human advisors who

Figure 2.1: Pipeline of a typical robo-advisor (Beketov et al., 2018).

charge high fees for their services. In contrast, robo-advisors offer similar investment advice at a fraction of the cost, making it more accessible and affordable to a broader range of investors (Beketov et al., 2018).

Furthermore, robo-advisors provide systematic and transparent advice to a new generation of clients (Alsabah, Capponi, Ruiz Lacedelli, & Stern, 2021; Beketov et al., 2018). They use algorithms to create personalized investment plans tailored to the investor's risk tolerance and financial situation. This approach provides investors with more transparent and objective advice than traditional investment advisory services, which can be influenced by the advisor's biases.

The robo-advisory industry has grown rapidly since its inception, and this trend is expected to continue in the future. As more investors turn to digital investment platforms, the number of robo-advisors on the market continues to increase, and established players are expanding their offerings to remain competitive. Many robo-advisors are also becoming more sophisticated, offering a wider range of investment options such as socially responsible investing and alternative investments. Assets under management in the robo-advisory segment are projected to reach US$2.76 trillion in 2023 (Statista, 2023). The growth of the industry is being driven by several factors, including increased investor demand for low-cost and transparent investment services and the proliferation of new technology that makes robo-advisors more accessible to a broader range of investors.

2.2 Pipeline of a robo-advisor

As per (Beketov et al., 2018), a typical robo-advisor has the following blocks, as shown in Figure 2.1.

1. **Universal asset selection.** This block involves providing a set of investment instruments that the robo-advisor can select from based on their corresponding risk and return information. The set of instruments is usually predetermined and provided by human financial advisors to the robo-advisor to choose from, rather than being chosen by the advisor itself. This approach allows for greater consistency in investment options across clients and helps to ensure that all clients have access to the same investment opportunities.

2. **Investor profile identification.** In this block, the robo-advisor models and infers the client's risk profile and investment preferences. This can be done through various means, such as questionnaires (Tertilt & Scholz, 2018), portfolio choices (Alsabah et al., 2021), or client interactions (Capponi et al., 2022). The goal is to understand the client's risk tolerance, investment goals, and other factors that may impact their investment decisions. By understanding the client's profile, the robo-advisor can provide personalized investment advice that is tailored to the client's specific needs.

3. **Portfolio optimization.** This block involves the optimization of the client's portfolio selection based on their risk profile and investment preferences. The robo-advisor uses algorithms to determine the optimal investment mix for the client based on factors such as risk tolerance, expected return, and investment horizon. Robo-advisors may use different optimization objectives to perform optimal asset allocation, such as to maximize mean-variance objective (Capponi et al., 2022; Alsabah et al., 2021; Wang & Yu, 2021) or to minimize risk with a constraint on the target mean return (Wang & Zhou, 2020). This allows for the efficient management of client portfolios and can help to maximize returns while minimizing risk.

4. **Monitoring and rebalancing.** The robo-advisor must continually monitor the performance of the selected investment policy and intervene when necessary. This may involve re-estimating the client's risk profile, adjusting the investment mix, or rebalancing the portfolio to ensure that it remains aligned with the client's investment goals. The robo-advisor may re-estimate the user's risk profile because (i) it finds its previous estimation not accurate; (ii) the client's risk profile may change from time to time. By providing ongoing monitoring and management, the robo-advisor can help clients stay on track to achieve their investment objectives.

5. **Performance report.** Finally, the robo-advisor should provide performance reports to clients to keep them informed about the performance of their investments, just like their human counterparts. This may include details on investment returns, fees, and other relevant information. By providing clients with regular updates, the robo-advisor can help to build trust and confidence in the investment management process.

In this report, we have a particular focus on the second, third and fourth blocks of a robo-advisor's pipeline – investor profile identification, portfolio optimization and monitoring and rebalancing. We aim to design a robo-advisor that models and estimates the risk profile of a client, exploits its estimation to optimize portfolio, and re-estimate the risk profile and rebalance

Model	Optimization Goal	Client Input	Interaction Model
Wang and Zhou (2020)	Risk minimization given target return. Not personalized.	Target return value.	No interaction.
Wang and Yu (2021)	Mean-variance objective with risk aversion.	Historical actions of the client where risk aversion is to be learnt through IRL.	No interactions.
Alsabah et al. (2021)	Mean-variance objective with risk aversion.	Historical and interactive actions of the client. Due to the bijection between risk aversion to action, this is equivalent to numerical risk aversion values.	Interactions with the client are initiated by the robo-advisor to receive updated risk aversion parameters with penalty on each interaction, resulting in exploration-exploitation trade-off.
Capponi et al. (2022)	Mean-variance objective with risk aversion.	Interactive behaviorally biased risk aversion parameters communicated by the client.	Interactions with the client are determined by a fixed interaction schedule and there's no explicit penalty on interactions. However, the behavioral bias at interaction times lead to trade-off between more interactions versus less bias.

Table 2.1: A taxomony of selected robo-advisors in the literature.

the portfolio when it finds necessary, through client interactions or actively asking client for intervening.

2.3 Existing robo-advisors and a taxonomy of them

Not only have robo-advisors drawn an emerging amount of attention in financial institutions and investors, but it has grown more and more popular in the quantitative finance research community as well. In this section, we study a few existing robo-advising models in the literature and give a taxonomy of robo-advisors based on their optimization goal, client input to the robo-advisor and the robo-advisor's interaction model. We focus on robo-advisors from the works of Capponi et al. (2022), Alsabah et al. (2021), Wang and Zhou (2020) and Wang and Yu (2021).

We classify any robo-advisor based on three dimensions.

1. **Optimization goal.** Regardless of what and how a robo-advisor may communicate with its client, it may take different optimization goals for portfolio selection. This is the core component of the third block, portfolio optimization, of a robo-advisor's pipeline as shown

in Figure 2.1. Since optimization goal is of particular importance to a robo-advisor, we first categorize robo-advisors based on their optimization goals. In general, there are two types of optimization goals:

(a) Personalizaed optimization goals. Robo-advisors whose optimization goals are determined by the client's risk profile and investment goals provide personalized investment advisory tailored to the client's needs. Robo-advisors with personalized optimization goals consider the risk aversion of different clients and make investment decisions based on it. Assuming the client's risk aversion is γ, then the most common optimization goal is to maximize the mean-variance objective function $\mathrm{E}[X] - \gamma/2\mathrm{Var}[X]$ where X is the client's wealth. In this setting, since the robo-advisor cannot always have complete information on the client's varying risk aversion over time (otherwise it would be equivalant to the client investing on herself), the advisor must estimate and update this value by modelling and interaction with the client. Capponi et al. (2022), Alsabah et al. (2021) and Wang and Yu (2021) can all be categorized as robo-advisors with personalized optimization goals.

(b) Non-personalized optimization goals. A simplified method is to model the mean variance portfolio selection problem as a variance minimization process with fixed target return. In this setting, the advisor only needs to ask the client for the target return, and then select portfolio with the minimum risk. Wang and Zhou (2020) can be categorized here where they use a reinforcement learning framework to solve the variance minimization problem. This is **not** personalized.

2. **Communicated client input.** In this dimension, we categorize robo-advisors based on *what* (i.e., the content) the client communicates with the advisor into two classes.

(a) The robo-advisor can directly obtain the required *parameter* (for example, client's risk aversion) from the client. This is a high level abstraction since the actual process of this may be more complicated (since the client is unable to give a numerical value for her risk aversion, but such value can be obtained from a questionnaire), which is out of our concern. Note that it is preferable to also consider and model the bias of this communicated risk aversion, as in Capponi et al. (2022) where they consider the behavioral bias of the client at interaction times due to trend-chasing mindset. We also classify (Alsabah et al., 2021) here because although the client input to the robo-advisor is her investment actions, they assume the function from risk aversion γ_n to the client's action is bijective and invertible, it is equivalent to the case where the client directly communicate her risk aversion to the robo-advisor. In (Alsabah et al., 2021), they do not consider the behavioral bias in the communicated risk aversion.

(b) The robo-advisor may be not able to obtain the required parameter such as risk aversion, but the robo-advisor may be able to observe client's *strategies and actions* instead. Such strategies usually refer to client's asset allocation decisions made in a market with complete information. The behavior observed by the robo-advisor may be historical, i.e., client's actions before the robo-advising process as a reference, or interactive, i.e., the robo-advisor observes or asks for the client's decisions on an ongoing basis. In this case, the problem can be modelled as a inverse reinforcement learning problem (Ng & Russell, 2000) where one can only observe the client's actions to the environment and infer the client's internal parameters to be used for robo-advised investment later. Wang and Yu (2021) propose a deep reinforcement learning algorithm to infer the risk preference of the client from historical client's asset allocations.

3. **Interaction model** At last, robo-advisors can also be categorized based on their interaction model. First, they can be classified into those with and without interactions with the client.

   (a) Robo-advisors without interactions with the client obtain all the information required from the client in a single information exchange. The robo-advisor may obtain the client's target risk or risk aversion through direct communication of numerical values or can estimate the required parameters from historical allocations of the client. The robo-advisor in Wang and Zhou (2020) receive a single target return value from the client to perform risk minimization with constraint on target return value. Wang and Yu (2021) let the robo-advisor to receive historical asset allocations of the client to infer the client's risk aversion. Both do not involve interactive communications during the robo-advising process.

   (b) There are also robo-advisors that allow interactions with the client during the robo-advising process. In this case, the robo-advisor may interact with the client (explore for better estimation of the risk aversion parameter), and optimize based on the information received (exploit the current belief of the risk aversion to make decisions). Such interactions may come with an opportunity cost (i.e., a penalty), naturally leading to an exploration-exploitation trade-off where the robo-advisor cannot choose to interact with client all the time for perfect estimation of client's risk aversion. This is the case in (Alsabah et al., 2021). Capponi et al. (2022) takes an alternative approach where there is no explicit penalty to the reward function on interactions. However, as they also consider the client's behavioral bias that is only activated at interaction times due to trend-chasing mindset, there is an implicit penalty for too many interactions and hence also lead to a trade-off between better estimations and less behavioral bias. This is discussed in much detail in Section 4.3.

Table 2.1 concludes our taxonomy for robo-advisors and categorize each of the models we study into one of the categories in each dimension.

Chapter 3

A robo-advising algorithm

In this chapter, we elaborate in great detail on a robo-advising algorithm – its problem setting, optimal strategy, computation of such strategy and the estimation of personalized parameters. This chapter is based on and an extention to the robo-advising algorithm described by Capponi et al. (2022). We adopt the their problem setting, but provide a much more detailed description on the numerical computation of the optimal strategy (in Section 3.3), and demonstrate a method of estimating personalized parameters of the client (in Section 3.4), which is oversimplified and omitted in Capponi et al. (2022).

3.1 Problem setting

3.1.1 Market model

We adopt the market model of Capponi et al. (2022). We consider a period of time of length T, i.e., $n = 0, 1, \ldots, T-1$. At time n, the economy state $Y_n \in \mathcal{Y}$ is observable to all, where $\mathcal{Y} = \{1, 2\}$ denoting the two possible economy states. We assume the economy state variable Y_n to follow a Markov chain with transition matrix P, i.e.

$$\Pr[Y_{n+1} = y | Y_n] = P_{Y_n, y}, \quad n = 0, 1, \ldots T-1.$$

There exist two available assets, a risk-free asset $(B_n)_{n\geq 0}$ and a risky asset $(S_n)_{n\geq 0}$, whose returns depend on the economy state. At time n, the risk-free asset has a constant interest rate $r_{n+1} = r(Y_n)$ dependent on the current economy state, and the risky asset has a random return rate $Z_{n+1} \sim \mathcal{N}(\mu(Y_n), \sigma^2(Y_n))$ whose mean and variance also depend on the current economy state Y_n.

Functions r, μ, σ^2 of economy state are public knowledge to all. Hence, we have

$$\begin{aligned} B_{n+1} &= (1 + r_{n+1})B_n, \\ S_{n+1} &= (1 + Z_{n+1})S_n. \end{aligned} \tag{3.1}$$

Note that r_{n+1} and Z_{n+1} are indexed by $n+1$ as they are the return rates of the respective assets at timestamp $n+1$. At time n, given the economy state Y_n, only the distribution of Z_{n+1} is revealed, instead of the actual valuation of Z_{n+1}. Z_{n+1} is only determined at time $n+1$. For simplicity of our notations, let $\mu_{n+1} = \mu(Y_n)$ and $\sigma_{n+1} = \sigma(Y_n)$. In this way, Z_{n+1} will follow the distribution $\mathcal{N}(\mu_{n+1}, \sigma^2_{n+1})$. Note that $\mu_{n+1}, \sigma_{n+1}, r_{n+1}$ only depend on Y_n and hence are revealed at time n.

We consider a self financing investment strategy $(\pi_n)_{n\geq 0}$ in the market dynamics described earlier, where π_n is the amount of wealth invested in the risky asset at time n. Let the client's wealth at time n to be X_n, and let X_0 to be a pre-defined constant. Then given a investment strategy π, the wealth process follows

$$X_{n+1} = (1 + r)(X_n - \pi_n) + (1 + Z_{n+1})\pi_n = (1 + r)X_n + (Z_{n+1} - r)\pi_n. \tag{3.2}$$

The robo-advisor, via interactions and modelling, chooses the investment strategy on behalf of the client, in order to maximize a mean-variance objective function dependent on the robo-advisor's modelling of client's risk aversion. To specify such objective function, we introduce the client's and the robo-advisor's model of risk aversion in Section 3.1.2 and Section 3.2.

3.1.2 Client model

Client's internal model of risk aversion process

The client has their internal model of risk aversion process, γ^C_n, $n = 0, 1, \dots, T-1$. Client's risk aversion depends on time, idiosyncratic shocks and economy status. More specifically, the client's risk aversion process follows

$$\gamma^C_n = e^{\eta_n}\gamma^{id}_n\gamma^Y_n, \qquad n = 0, 1, \dots, T-1. \tag{3.3}$$

Client's risk aversion process consists of three components:

1. **Effects of age over time.** $\eta_n = -\alpha(T - n)$ captures the effect of age increasing the risk aversion over time and parameter $\alpha \in \mathbb{R}_{\geq 0}$ represents the magnitude of such effect. The larger α is, the more risk averse the client become as time goes by.

2. **Personal idiosyncratic shocks.** γ_n^{id} represents the idiosyncratic shocks to the client's risk aversion independent on time, economy states or investment performance. It is defined as

$$\gamma_n^{id} = \gamma_{n-1}^{id} \cdot e^{\varepsilon_n}, \qquad n = 1, \dots, T-1. \tag{3.4}$$

It is controlled by $(\varepsilon_n)_{n\geq 1}$, an independently and identically distributed random variables that admit the following distribution

$$\varepsilon_n = \begin{cases} \mathcal{N}(0, \sigma_\varepsilon^2), & \text{with probability} p_\varepsilon; \\ 0, & \text{with probability} 1 - p_\varepsilon. \end{cases} \tag{3.5}$$

Here, $p_\varepsilon \in [0, 1]$, $\sigma_\varepsilon \in \mathbb{R}^+$ are personalized parameters of the client. p_ε is the probability that this component changes, in which case ε_n will be drawn from a normal distribution with zero mean and σ_ε^2 variance. With probability $1 - p_\varepsilon$, the idiosyncratic shock component will not change. Usually p_ε is set to be small, such as 0.05, to reflect the practical scenario that a random oscillation to client's risk aversion does not occur with a high probability.

3. **Economy state.** A client's risk aversion also depends on the current economy state, reflected by the component γ_n^Y. γ_n^Y is a deterministic constant determined by (Y_n), which increases as the market's Sharpe ratio increases (Lettau & Ludvigson, 2010). This is because a human client tends to reduce his or her investment in the risky asset if the market Sharpe ratio is high. Capponi et al. (2022) only describes this coefficient to be increasing in the market Sharpe ratio, but does not give a construction of γ_n^Y. To make sure that $\gamma_n^Y \geq 0$, we define γ_n^Y as

$$\gamma_n^Y = \exp(S(Y_n)) = \exp\left(\frac{\mu(Y_n) - r(Y_n)}{\sigma(Y_n)}\right). \tag{3.6}$$

At last, we show that $(\gamma_n^C)_{n\geq 0}$ is well-defined once the initialization γ_0^C is given.

The risk aversion process $(\gamma_n^C)_{n\geq 0}$ is initialized with a given constant γ_0, which is also a personalized parameter of the client (in addition to $\alpha, p_\varepsilon, \sigma_\varepsilon$). Once $\gamma_0^C = \gamma_0$ is determined, as η_0, γ_0^Y are deterministic given the time horizon and initialized economy state, γ_0^{id} is initialized accordingly as $\gamma_0/(e^{\eta_0}\gamma_0^Y)$. For later timestamps, $\eta_n = -\alpha(T-n)$, γ_n^{id} follows Equation (3.4) and γ_n^Y follows Equation (3.6). Hence, client's risk aversion process $(\gamma_n^C)_{n\geq 0}$ is well-defined.

Client's behavioral bias on risk aversion

Other than client's risk aversion process $(\gamma_n^C)_{n\geq 0}$, we also consider behavioral bias on client's risk aversion influenced by market performance (the return rate of the risky asset). Note that client does not always know about market performance unless he or she interacts with the robo-advisor and being inquired to communicate update his or her risk aversion. Denote τ_n to be the timestamp of the most recent interaction at time n. It is trivial to see that time n is an interaction time if and only if $\tau_n = n$, and the previous interaction time (if that exists, i.e. $n > 0$) is at time τ_{n-1}.

At interaction time $n > 0$, where $n = \tau_n$, client's behavioral bias will be applied on the communicated risk aversion

$$\xi_n = \gamma_n^C \gamma_n^Z = \gamma_n^C \exp\left(-\beta\left(\frac{1}{n-\tau_{n-1}} \sum_{k=\tau_{n-1}}^{n-1} (Z_{k+1} - \mu_{k+1})\right)\right). \tag{3.7}$$

In Equation (3.7), γ_Z is defined in a way that depends on the market performance since after the previous interaction time (at time $\tau_{n-1}+1$) up to now (at time n). If the total return of the risky asset $\sum_{k=\tau_{n-1}}^{n-1} Z_{k+1}$ exceeds the expectation $\sum_{k=\tau_{n-1}}^{n-1} \mu_{k+1}$, γ_n^Z will be less than one, indicating a discounting effect on the communicated risk aversion, and vise versa. This reflects the behavioral bias of the client due to the trend-chasing mindset (Oechssler, Roider, & Schmitz, 2009), that is, a client will tend be more risk-tolerant if the performance of the current portfolio exceeds the expectation.

At initial time $n = 0$, even if it is an interaction time (i.e. $\tau_0 = 0$), there will be no behavioral bias on the communicated risk aversion, i.e. $\xi_0 = \gamma_0^C = \gamma_0$, because there is no history of market performance for the client to observe.

Also, at non-interaction times, we define $\xi_n = \xi_{\tau_n}$, the most recent communicated risk aversion.

At last, note that β in Equation (3.7) is another client's personalized parameter that controls the magnitude of the trend-chasing behavioral bias. Larger β indicates that the client will be biased to be more tolerant towards risk if the market performs better than expectation. To summarize, there are in total the following such personalized parameters that control the dynamics of the client's risk aversion process and behavioral bias, as shown in Table 3.1.

3.2 Robo-advisor's model and optimal strategy

In this section, we describe the robo-advisor's model of the client's risk aversion process, its investment objective, and its optimal strategy. Again, we mostly follow the settings of Capponi et

Parameter	Domain	Description
γ_0	$\mathbb{R}_{\geq 0}$	The initial risk aversion of the client.
α	$\mathbb{R}_{\geq 0}$	Measures the effects of time on increasing the client's risk aversion.
β	$\mathbb{R}_{\geq 0}$	Measures how much the market performance history will bias the client's communicated risk aversion at interaction times.
p_ε	$[0, 1]$	The probability of the idiosyncratic shock component does not stay constant (i.e. $\varepsilon_n \neq 0$).
σ_ε	$\mathbb{R}_{\geq 0}$	The standard deviation of ε_n if it is not zero, which occurs with probability p_ε.

Table 3.1: List of client's personalized parameters that control the client's risk aversion and behavioral bias.

al. (2022) and complement where is not described in much detail.

3.2.1 Robo-advisor's model of the client's risk aversion process

First, Capponi et al. (2022) assume that the personalized parameters of the client, i.e., the parameters listed in Table 3.1, are known to the robo-advisor. That is, Capponi et al. (2022) assume that the robo-advisor has complete knowledge on the mechanism following which the client's risk aversion process is generated. However, this is an oversimplified assumption that may not hold realistic in practice. For now, we assume this assumption holds and we will give an algorithm in Section 3.4 for the robo-advisor to estimate these personalized parameters through testing interactions beforehand.

Interaction model

As introduced earlier, the robo-advisor follows a pre-determined fixed interaction schedule that is defined by the sequence $(\tau_n)_{n\geq 0}$ where τ_n is the most recent interaction time before timestamp n (inclusive). Following Capponi et al. (2022), we consider an uniform interaction schedule parameterized by the interaction interval $\phi \in \mathbb{N}_+$, where $0, \phi, 2\phi, \ldots$ are the timestamps where interaction occurs.

At interaction time n, i.e., $\tau_n = n$, the robo-advisor interacts with the client and receives communicated risk aversion ξ_n that is behaviorally biased by the market performance since the previous interaction (if existing) up to date, as defined in Equation (3.7).

Robo-advisor's model of the client's risk aversion process

Through its knowledge on the distribution of the client's risk aversion process, and the communicated (biased) risk aversion values at interaction times, the robo-advisor maintains its own model of client's risk aversion process, denoted as $(\gamma_n^R)_{n\geq 0}$, following

$$\gamma_n^R = e^{\eta_n - \eta_{\tau_n}} \xi_{\tau_n} \frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}. \tag{3.8}$$

Equation (3.8) has the following properties.

1. On interaction time n, the robo-advisor's model of risk aversion γ_n^R is the same as the communicated risk aversion ξ_n as $n = \tau_n$. This makes sense because the robo-advisor will operate on behalf of the client and follows client's direct instructions of the communicated risk aversion.

2. On any time n, γ_n^R is an *approximation* of the expected value of the client's risk aversion γ_n^C times the most recent behavioral bias $\gamma_{\tau_n}^Z$. This means that the robo-advisor operates based on its belief of the client's biased risk aversion (as the bias component will remain constant unless interacting with the advisor). To verify the claim, notice

$$\begin{aligned}
\gamma_{\tau_n}^Z \mathrm{E}[\gamma_n^C] &= \xi_{\tau_n} \mathrm{E}\left[\frac{\gamma_n^C}{\gamma_{\tau_n}^C}\right] \\
&= \xi_{\tau_n} \mathrm{E}\left[e^{\eta_n - \eta_{\tau_n}} \frac{\gamma_n^{id}}{\gamma_{\tau_n}^{id}} \frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}\right] \\
&= e^{\eta_n - \eta_{\tau_n}} \xi_{\tau_n} \frac{\gamma_n^Y}{\gamma_{\tau_n}^Y} \mathrm{E}\left[\frac{\gamma_n^{id}}{\gamma_{\tau_n}^{id}}\right] \\
&= e^{\eta_n - \eta_{\tau_n}} \xi_{\tau_n} \frac{\gamma_n^Y}{\gamma_{\tau_n}^Y} \mathrm{E}\left[e^{\sum_{i=\tau_n+1}^{n} \varepsilon_i}\right] \\
&\approx e^{\eta_n - \eta_{\tau_n}} \xi_{\tau_n} \frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}
\end{aligned} \tag{3.9}$$

   since ε_n follows Equation (3.5) and is zero with high probability (i.e. $1 - p_\varepsilon$ where p_ε is set to be very small, as per previous discussions).

Note that Equation (3.8) is an approximation of the actual expectation, and the precise value is given by

$$
\begin{aligned}
\gamma_{\tau_n^Z}\mathrm{E}[\gamma_n^C] &= e^{\eta_n-\eta_{\tau_n}}\xi_{\tau_n}\frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}\mathrm{E}\left[e^{\sum_{i=\tau_n+1}^{n}\varepsilon_i}\right] \\
&= e^{\eta_n-\eta_{\tau_n}}\xi_{\tau_n}\frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}\prod_{i=\tau_n+1}^{n}\mathrm{E}\left[e^{\varepsilon_i}\right] \\
&= e^{\eta_n-\eta_{\tau_n}}\xi_{\tau_n}\frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}\prod_{i=\tau_n+1}^{n}\mathrm{E}\left[p_\varepsilon e^X+(1-p_\varepsilon)e^0\right], \qquad \text{where } x\sim\mathcal{N}(0,\sigma_\varepsilon^2) \\
&= e^{\eta_n-\eta_{\tau_n}}\xi_{\tau_n}\frac{\gamma_n^Y}{\gamma_{\tau_n}^Y}(e^{\sigma_\varepsilon^2/2}p_\varepsilon+(1-p_\varepsilon))^{n-\tau_n}.
\end{aligned}
\tag{3.10}
$$

Here, if we would like Equation (3.8) to be exact value instead of approximation, one can change the definition of ε_n in Equation (3.5) to $\mathcal{N}(-\sigma_\varepsilon^2/2,\sigma_\varepsilon^2)$ with probability p_ε and zero with probability $1-p_\varepsilon$. However, such definition will lead to unnecessary complication in Section 3.4, and hence we stick to Equation (3.5).

At last, note that unlike Alsabah et al. (2021), there is no explicit penalty applied to each interaction, there are implicit penalty to too frequent interactions due to the existence of behavioral bias. More frequent interactions will lead to better estimation of the client's risk aversion, but this will also lead to higher behavioral bias as such bias is only applied at interaction times. This trade-off between more accurate estimation and higher bias will be discussed in more detail in Section 4.3.

3.2.2 Robo-advisor's optimization goal

Now that we define the robo-advisor's model of risk aversion, we now define the robo-advisor's optimization goal. We adopt the well-known mean-variaznce objective function (Markowitz, 1952).

To be more specific, with fixed time horizon T, at time n, the robo-advisor maximizes the mean-variance objective function

$$
\mathrm{E}\left[\frac{X_T^\pi-X_n}{X_n}\right]-\frac{\gamma_n^R}{2}\mathrm{Var}\left[\frac{X_T^\pi-X_n}{X_n}\right]. \tag{3.11}
$$

That is, at time n, the robo-advisor aims to maximize the mean-variance objective with its model of client's risk aversion at time n with regard to the expected return at the terminal time T, conditioning on all the information it can obtain at time n.

3.2.3 Robo-advisor's optimal strategy and its evaluation

Note that in Equation (3.11), the optimization objective at time n not only depends on the information available to the robo-advisor so far, but also involves the estimation of future random processes conditioned on current knowledge. Let $\mathcal{D}_n$ be the set of all possible history information up to time n. Then at time n, given the current wealth X_n and the current history d_n, the optimal strategy at time n, π_n^*, is a function of x and d_n, because at time n, that's all the information the robo-advisor knows, and it must make a decision based on such information. In fact, this function is of the form

$$\pi_n^*(x, d_n) = \tilde{\pi}_n^*(d_n) \cdot x \tag{3.12}$$

for all $x \in \mathbb{R}_{\geq 0}$ and all $d_n \in \mathcal{D}_n$, where naturally, $\tilde{\pi}_n^*$ is the proportion of current wealth allocated to the risky asset at time n. Therefore, our goal is to find $\tilde{\pi}_n^*(d)$ for all possible $d \in \mathcal{D}_n$.

Capponi et al. (2022) gives the following result for the robo-advisor's optimal strategy. Since the proof of Theorem 3.2.1 is not of our particular focus (we focus on the numerical algorithm for computation of the optimal strategy), we refer the audience to the original paper by Capponi et al. (2022).

Theorem 3.2.1. *For an optimal solution to (3.11), at time n, with history $d \in \mathcal{D}_n$, the proportion of current wealth allocated to the risky asset follows*

$$\tilde{\pi}_n^*(d) = \frac{1}{\gamma_n^R} \frac{\mathrm{E}[(Z_{n+1} - r_{n+1})\frac{X_T^{\pi^*}}{X_{n+1}}]}{\mathrm{Var}[(Z_{n+1} - r_{n+1})\frac{X_T^{\pi^*}}{X_{n+1}}]} - (1 + r_{n+1}) \frac{\mathrm{Cov}[\frac{X_T^{\pi^*}}{X_{n+1}}, (Z_{n+1} - r_{n+1})\frac{X_T^{\pi^*}}{X_{n+1}}]}{\mathrm{Var}[(Z_{n+1} - r_{n+1})\frac{X_T^{\pi^*}}{X_{n+1}}]}, \tag{3.13}$$

where all the expectation, variance and covariance are taken over the randomness of the future market (from $Y_{n+1}, \ldots$ and $Z_{n+1}, \ldots$) and client's risk aversion process (from $\varepsilon_{n+1} \ldots$), conditioning on the current information $d \in \mathcal{D}_n$.

Game theoretic interpretation of the optimal strategy

Note that in Equation (3.13), the optimal strategy at time n will depend on the future optimal strategy. This can be interpreted in a game-theoretic way. We can treat finding the optimal strategy to maximize (3.11) as a multi-stage multi-player game, where the robo-advisor at time n is seen as the $(n + 1)$-th player. Each play has a reward function to maximize that is the mean-variance objective of the terminal return given current information. At time n, the player can only make decisions based on the current information, given that the future players (i.e., the robo-advisor itself in the future rounds) will act optimally. Therefore, if the strategy at all n satisfies Equation (3.13), it will be a subgame-perfect equilibrium, and hence optimal.

In the spirit of finding subgame-perfect equilibria, we use backwards induction. For the terminal time $n = T - 1$, Equation (3.13) degenerates to a single step single player game, and the optimal strategy $\tilde{\pi}^*_{T-1}$ is given by solving the usual simple mean-variance maximization for any possible full history d_{T-1}. At time $n < T - 1$, assuming that we have solved the optimal strategy of later players given all possible history information. Then the optimal strategy $\tilde{\pi}^*_n$ is found as a solution to Equation (3.11) where all later players follow the optimal strategy solved earlier. Note that when solving for $\tilde{\pi}^*_n$, when calculating the expectation, variance and covariance terms in Equation (3.13), one need to integrate over all possible future scenarios, which can be done because we have solved $\tilde{\pi}^*_m(d)$ for all possible d for all m greater than n. This is analogous to dynamic programming where results of all possible sub-problems are saved for reference to solve the complete problem.

After the optimal strategy as a function of history information is found via backwards induction, the robo-advisor starts engaging with the actual market and the client. At time n, the robo-advisor observes the market information and models client's risk aversion to formulate the history $d_n \in \mathcal{D}_n$, and looks up the earlier found optimal strategy and allocate $\tilde{\pi}^*_n(d_n)$ of the current wealth to the risky asset.

Therefore, the procedure of finding the optimal strategy $\tilde{\pi}^*$ can be described as Algorithm 1.

3.3 Numerical algorithm for computation of optimal strategy

Following Capponi et al. (2022), we describe a numerical algorithm to evaluate the optimal investment strategy in Equation (3.13) via backwards induction. Capponi et al. (2022) did not describe the algorithm in great detail but only the high-level ideas. In this section, we close such gap and formally describe the algorithm and its implementation.

We first point out two issues with Algorithm 1 that render the procedure unrealistic to be run in practice.

1. Although it is possible to solve the terminal strategy $\tilde{\pi}^*_{T-1}$, it is not trivial to solve for $\tilde{\pi}^*_n$ for smaller n in Line 2 of Algorithm 1 as this involves expectation, variance and covariance over the randomness of future market and client's risk aversion process.

2. For each n, one will have to discretize $\mathcal{D}_n$, the space of history up to time n, which is not practical as it is very high dimensional data. The information history available to the

Algorithm 1 Overview of the procedure of the robo-advising algorithm.
The algorithm consists of two parts: (i) backwards induction to find the optimal strategy as a function of history information; (ii) run the robo-advisor to use the found strategy for portfolio selection as the robo-advisor observes information as time goes by.

1. For all possible $d \in \mathcal{D}_{T-1}$, solve $\tilde{\pi}^*_{T-1}(d)$ by solving Equation (3.13).
2. For all $n = T-2, \ldots, 0$:
   - For all possible $d \in \mathcal{D}_n$, solve $\tilde{\pi}^*_n(d)$ by solving Equation (3.13), this is possible because for any $m > n$, and for any $d' \in \mathcal{D}_m$, we have already solved $\tilde{\pi}^*_m(d')$. Hence, one can evaluate the expectation, variance and covariance of future dynamics conditioned on the current information d and the optimality of future actions.
3. This marks the termination of backwards induction, the robo-advisor stores the optimal strategy $\tilde{\pi}^*_n(d)$ for all $d \in \mathcal{D}_n$ for all n.
4. For $n = 0, 1, \ldots, T-1$:
   - The robo-advisor obtains information $d_n \in \mathcal{D}_n$ via observing the economy, the market, and possible interactions with the client.
   - The robo-advisor allocates wealth of the amount $\tilde{\pi}^*_n(d_n)X_n$ to the risky asset, which is the optimal strategy.

robo-advisor up to time n include:

- Economy states $(Y_0, Y_1, \ldots, Y_n) \in \{1, 2\}^n$;
- Risky asset returns $(Z_1, Z_2, \ldots, Z_n) \in \mathbb{R}^n$;
- Communicated risk aversions from the client $(\xi_0, \xi_1, \ldots, \xi_n) \in \mathbb{R}^n_{\geq 0}$.

Therefore, for any n, to enumerate all possible $d \in \mathcal{D}_n$, one need to enumerate at least

$$O\left(\frac{|\mathcal{Y}|^n}{(\Delta Z)^n(\Delta\xi)^n}\right), \tag{3.14}$$

tuples of $((Y_0, Y_1, \ldots, Y_n), (Z_1, Z_2, \ldots, Z_n), (\xi_0, \xi_1, \ldots, \xi_n))$, where ΔZ is the step size to discretize a single risky asset return and $\Delta\xi$ is the step size to discretize a single risk aversion. This leads to exponentially large complexity only to discretize a single step and is not possible to compute in reality.

Capponi et al. (2022) addresses the first issue by decomposing the solution in Equation (3.13), while they oversimplify the process to solve the second issue. We will close such gap by proving that there exists a tuple of low-dimensional data whose discretization is enough to find optimal strategy of all possible scenarios.

3.3.1 Decomposition of the optimal strategy for computational convenience

Following Capponi et al. (2022), one can decompose Equation (3.13) into the following. For any n and any state $d_n \in D_n$, we have

$$\tilde{\pi}_n^*(d_n) = \frac{1}{\gamma_n^R} \frac{\mu_n^{az}(d_n) - (1 + r_{n+1})\gamma_n^R(\mu_n^{bz}(d_n) - \mu_n^a(d_n)\mu_n^{az}(d_n))}{\mu_n^{bz^2}(d_n) - (\mu_n^{az}(d_n))^2}, \tag{3.15}$$

where

$$\begin{aligned} \mu_n^a(d_n) &= \mathrm{E}[a_{n+1}(d_{n+1})] \\ \mu_n^{az}(d_n) &= \mathrm{E}[a_{n+1}(d_{n+1}) \cdot (Z_{n+1} - r_{n+1})] \\ \mu_n^{bz}(d_n) &= \mathrm{E}[b_{n+1}(d_{n+1}) \cdot (Z_{n+1} - r_{n+1})] \\ \mu_n^{bz^2}(d_n) &= \mathrm{E}[b_{n+1}(d_{n+1}) \cdot (Z_{n+1} - r_{n+1})^2] \end{aligned} \tag{3.16}$$

where for $n = 0, 1, \ldots, T-1$ and any $d_n \in \mathcal{D}_n$,

$$\begin{aligned} a_n(d_n) &= \mathrm{E}[(1 + r_{n+1} + (Z_{n+1} - r_{n+1})\tilde{\pi}_n^*(d_n))a_{n+1}(d_{n+1})] \\ b_n(d_n) &= \mathrm{E}[(1 + r_{n+1} + (Z_{n+1} - r_{n+1})\tilde{\pi}_n^*(d_n))^2 b_{n+1}(d_{n+1})] \end{aligned} \tag{3.17}$$

and for $n = T$, $a_T = b_T \equiv 1$. Here, all the expectations are taken over the randomness of future state d_{n+1}, which consists of the randomness of market (originating from future economy state Y_{n+1} and market return Z_{n+1}) and risk aversion process (originating from ε_{n+1}), all conditioned on the current state d_n.

Carefully inspecting Equations (3.15), (3.16) and (3.17), one would easily find the following order of evaluation:

$$\begin{aligned} & a_T, b_T && \Rightarrow \mu_{T-1}^a, \mu_{T-1}^{az}, \mu_{T-1}^{bz}, \mu_{T-1}^{bz^2} && \Rightarrow \tilde{\pi}_{T-1}^* \\ \Rightarrow \quad & a_{T-1}, b_{T-1} && \Rightarrow \mu_{T-2}^a, \mu_{T-2}^{az}, \mu_{T-2}^{bz}, \mu_{T-2}^{bz^2} && \Rightarrow \tilde{\pi}_{T-2}^* \\ \Rightarrow \quad & a_{T-2}, b_{T-2} && \Rightarrow \ldots \end{aligned} \tag{3.18}$$

To be more specific, for all $n = T-1, \ldots, 0$, and for any possible state $d_n \in \mathcal{D}_n$, one can first calculate the values of $\mu_n^a(d_n), \mu_n^{az}(d_n), \mu_n^{bz}(d_n), \mu_n^{bz^2}(d_n)$ via Equation (3.16) given that $a_{n+1}(d_{n+1}), b_{n+1}(d_{n+1})$ have been solved for all possible state d_{n+1}. This is because in Equation 3.16, the expectations are evaluated by integrating over all possible future states d_{n+1} conditioned on current state d_n. Then, since the current state d_n is known, the robo-advisor model the risk aversion as γ_n^R and hence the optimal strategy at time n given state d_n, $\tilde{\pi}_n^*(d_n)$, can be valuated by Equation (3.15). At last, one can calculate $a_n(d_n), b_n(d_n)$ via Equation (3.17) as it only involves expectation over the future states at time $n+1$, which can be calculated given d_n. One can keep doing this until we find all the $\pi_n^*(d)$ for all n and d.

3.3.2 Discretization of low-dimensional data

Although the decomposition of the optimal strategy presented in Equation (3.15) leads to an algorithm (described in Equation 3.18) to evaluate $\tilde{\pi}_n^*(d)$ at any n given any state $d \in \mathcal{D}_n$, it is still an impractical algorithm because of the high dimensionality of D_n and its complexity to be discretized. As shown in earlier sections, each state d_n at time n consists of three vectors of length n and to discretize $\mathcal{D}_n$ to enumerate all states d_n admits exponential complexity that is not feasible in reality. Capponi et al. (2022) briefly addresses this issue by making a claim that there exists a low-dimensional tuple that uniquely determines all the terms in Equations (3.15), (3.16) and (3.17), and that discretizing such low-dimensional data is sufficient for finding $\tilde{\pi}_n^*(d)$ for all possible state $d \in \mathcal{D}_n$. However, they did not elaborate on such discretization nor did they show how the terms in Equations (3.15), (3.16) and (3.17) can be represented using these low-dimensional tuples. In this section, we formally describe a low-dimensional tuple of five variables, such that all terms in Equations (3.15), (3.16) and (3.17) will only depend on this tuple.

Theorem 3.3.1. *For time n, let $\tilde{d}_n$ be a tuple of the following five values:*

- Y_{τ_n}*: the economy state at the last interaction time;*
- Y_n*: the current economy state;*
- $\sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1})$*: the sum of excess market returns between the two most recent interaction times;*
- $\sum_{k=\tau_n}^{n-1}(Z_{k+1}-\mu_{k+1})$*: the sum of excess market returns since the most recent interaction time;*
- ξ_n*: the most recent communicated risk aversion.* [1]

[1]Note that the most recent interaction should be at time τ_n. If n is interaction time, then $n=\tau_n$ and $\xi_{\tau_n}=\xi_n$. And if n is not interaction time, then by definition $\xi_n=\xi_{\tau_n}$. Therefore, ξ_n is indeed the most recent communicated risk aversion.

Let $\tilde{\mathcal{D}}_n = \mathcal{Y} \times \mathcal{Y} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_{\geq 0}$ to be the space of all possible $\tilde{d}_n$. Then all terms appearing in Equations (3.15)*,* (3.16) *and* (3.17) *can be determined by the values in $\tilde{d}_n$. In other words, $\tilde{\pi}^*_n, \mu^a_n, \mu^{az}_n, \mu^{bz}_n, \mu^{bz^2}_n, a_n, b_n, \gamma^R_n$ are not only functions of the full history d_n, but also functions of the low-dimensional state $\tilde{d}_n$.*

Proof. To proof that all the terms mentioned in the Equations (3.15), (3.16) and (3.17) only depend on the five variables in $\tilde{d}_n$, we follow the order of execution in Equation (3.18) to give a closed-form representation of each term w.r.t. $\tilde{d}_n$ respectively.

First, we examine $\mu^a_n(d_n), \mu^{az}_n(d_n), \mu^{bz}_n(d_n), \mu^{bz^2}_n(d_n)$ from Equation (3.16), and show how to evaluate their values given only $\tilde{d}_n$. We take μ^{az}_n as an example, and other functions are highly similar.

Given $\tilde{d}_n \in \tilde{\mathcal{D}}_n$,

$$\begin{aligned}\mu^{az}_n(\tilde{d}) &= \mu^{az}_n(Y_{\tau_n}, Y_n, \sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1}), \sum_{k=\tau_n}^{n-1}(Z_{k+1}-\mu_{k+1}), \xi_n) \\ &= \mathrm{E}[a_{n+1}(\tilde{d}_{n+1})(Z_{n+1}-r_{n+1})].\end{aligned}$$

Therefore, the key is to integrate over all the possible $\tilde{d}_{n+1}$ conditioned on $\tilde{d}_n$.

1. If $n+1 \neq \tau_{n+1}$, i.e., $n+1$ is not an interaction time. Then, the next state $\tilde{d}_{n+1}$ only depends on the next economy state Y_{n+1} and the market return at time $n+1$, Z_{n+1}, which follows distribution $\mathcal{N}(\mu_{n+1}, \sigma^2_{n+1})$ where $\mu_{n+1}, \sigma^2_{n+1}$ are known because they are determined by current economy state Y_n, which is given in $\tilde{d}_n$. Hence, to calculate the expectation, we need to integrate over Y_{n+1} and Z_{n+1}, i.e.,

$$\mu^{az}_n(\tilde{d}_n) = \sum_{Y_{n+1}\in\mathcal{Y}} P_{Y_n,Y_{n+1}} \int_{z\in\mathbb{R}} a_{n+1}(\tilde{d}_{n+1})(z-r_{n+1}) f_{Z_{n+1}}(z)\mathrm{d}z, \tag{3.19}$$

   where $f_{Z_{n+1}}(\cdot)$ is the probability density function of Z_{n+1}, i.e., the probability density function of $\mathcal{N}(\mu_{n+1}, \sigma^2_{n+1})$.

   Given Y_{n+1}, Z_{n+1}, the next state $\tilde{d}_{n+1}$ is determined as follows:

   - The economy state at the last interaction time is Y_{τ_n} because the most recent interaction is still at τ_n if $n+1$ is not.
   - The current (at time $n+1$) economy state is Y_{n+1}, which is given.
   - The cumulative excess market returns between the two most recent interaction times is still $\sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-u_{k+1})$ because the two most recent interactions are still at $\tau_n - \phi$

and τ_n.

- The cumulative excess market returns since the most recent interaction time ($\tau_{n+1} = \tau_n$) u to date $(n+1)$ is $\sum_{k=\tau_n}^{n-1}(Z_{k+1} - \mu_{k+1}) + Z_{n+1} - \mu_{n+1}$.
- The most recent communicated risk aversion keeps ξ_n as there's no new interaction.

Therefore, if $n+1$ is not an interaction time, we have

$$\tilde{d}_{n+1} = (Y_{\tau_n}, Y_{n+1}, \sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1} - u_{k+1}), \sum_{k=\tau_n}^{n-1}(Z_{k+1} - \mu_{k+1}) + Z_{n+1} - \mu_{n+1}, \xi_n), \tag{3.20}$$

which completes Equation (3.19).

2. If $n+1$ is indeed an interaction time, then the next state $\tilde{d}_{n+1}$ depends not only on Y_{n+1}, Z_{n+1}, but also on the client's risk aversion process, due to the ϕ i.i.d. random variables $\varepsilon_{\tau_n+1}, \ldots, \varepsilon_{n+1}$. In fact, $\tilde{d}_{n+1}$ will depend only on the sum $\varepsilon_{\tau_n+1} + \ldots + \varepsilon_{n+1}$. Therefore, to calculate the expectation, one needs to integrate over all three variables, i.e.,

$$\mu_n^{az}(\tilde{d}_n) = \sum_{Y_{n+1}\in\mathcal{Y}} P_{Y_n,Y_{n+1}} \int_{z\in\mathbb{R}} \int_{\varepsilon\in\mathbb{R}} a_{n+1}(\tilde{d}_{n+1})(z - r_{n+1}) f_{Z_{n+1}}(z) f_{\varepsilon}^{(\phi)}(\varepsilon) \mathrm{d}z\mathrm{d}\varepsilon, \tag{3.21}$$

where $f_{\varepsilon}^{(\phi)}$ is the probability density function of random variable $\varepsilon_{\tau_n+1} + \ldots + \varepsilon_{n+1}$, which can be found via ϕ-fold convolution of ε_n whose distribution is defined in Equation (3.5).

Given $Y_{n+1}, Z_{n+1}, \varepsilon = \varepsilon_{\tau_n+1} + \ldots + \varepsilon_{n+1}$, the next state $\tilde{d}$ is determined as follows:

- The economy state at the last interaction time is Y_{n+1} because the most recent interaction is $n+1$ as $n+1$ is an interaction time.
- The current (at time $n+1$) economy state is Y_{n+1}, which is given.
- The cumulative excess market returns between the two most recent interaction times is

$$\sum_{k=\tau_n}^{n-1}(Z_{k+1} - u_{k+1}) + Z_{n+1} - \mu_{n+1}$$

  because the two most recent interactions are at $\tau_n = n+1-\phi$ and $n+1$.
- The cumulative excess market returns since the most recent interaction time $(n+1)$ u to date $(n+1)$ is 0.

- At last, for the most recent communicated risk aversion ξ_{n+1}, we have

$$\xi_{n+1} = \gamma_{n+1}^{C} \exp\left(-\frac{\beta}{\phi}\left(\sum_{k=\tau_n}^{n-1}(Z_{k+1}-\mu_{k+1}) + Z_{n+1} - \mu_{n+1}\right)\right)$$

where

$$\gamma_{n+1}^{C} = \gamma_{\tau_n}^{C} \exp(\eta_{n+1} - \eta_{\tau_n}) \frac{\gamma_{n+1}^{id}}{\gamma_{\tau_n}^{id}} \frac{\gamma_{n+1}^{Y}}{\gamma_{\tau_n}^{Y}} = \gamma_{\tau_n}^{C} \exp(\alpha\phi + \varepsilon) \frac{\exp S(Y_{n+1})}{\exp S(Y_{\tau_n})}$$

where

$$\gamma_{\tau_n}^{C} = \xi_{\tau_n} \exp\left(\frac{\beta}{\phi} \sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1})\right) = \xi_n \exp\left(\frac{\beta}{\phi} \sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1})\right).$$

Assembling the above, we have

$$\xi_{n+1} = \xi_n e^{\left(\alpha\phi+\varepsilon-\frac{\beta}{\phi}\left(\sum_{k=\tau_n}^{n-1}(Z_{k+1}-\mu_{k+1})+Z_{n+1}-\mu_{n+1}-\sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1})\right)+S(Y_{n+1})-S(Y_{\tau_n})\right)}. \tag{3.22}$$

Note that all terms appearing in Equation (3.22) can be calculated as they are either given by $\tilde{d}_n$ or provided by $X_{n+1}, Y_{n+1}, \varepsilon$.

Therefore, given current state $\tilde{d}_n$ and $Y_{n+1}, Z_{n+1}, \varepsilon$, the next state follows

$$\tilde{d}_{n+1} = \begin{pmatrix} Y_{n+1}, \\ Y_{n+1}, \\ \sum_{k=\tau_n}^{n-1}(Z_{k+1}-u_{k+1}) + Z_{n+1} - \mu_{n+1}, \\ 0, \\ \xi_n e^{\left(\alpha\phi+\varepsilon-\frac{\beta}{\phi}\left(\sum_{k=\tau_n}^{n-1}(Z_{k+1}-\mu_{k+1})+Z_{n+1}-\mu_{n+1}-\sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1})\right)+S(Y_{n+1})-S(Y_{\tau_n})\right)} \end{pmatrix} \tag{3.23}$$

Substituting Equation (3.23) as $\tilde{d}_{n+1}$ back to Equation (3.21), one can calculate $\mu_n^{az}(\tilde{d}_n)$ via integration over $Y_{n+1}, Z_{n+1}, \varepsilon$.

Hence, we've shown that μ_n^{az} can be determined only by $\tilde{d}_n$ (instead of the high-dimensional d_n). One can calculate $\mu_n^{a}, \mu_n^{bz}, \mu_n^{bz^2}$ similarly given any tuple $\tilde{d}_n$. Next, as instructed by the execution order in Equation (3.18), we examine the evaluation of $\tilde{\pi}_n^*$ via Equation (3.15). Note that all terms but γ_n^R has been determined, then it suffices to show that one can evaluate γ_n^R given $\tilde{d}_n$.

This is possible because as per its definition (Equation (3.8))

$$\gamma_n^R = \exp(\eta_n - \eta_{\tau_n})\xi_{\tau_n}\frac{\gamma_n^Y}{\gamma_{\tau_n}^Y} = \exp(\eta_n - \eta_{\tau_n})\xi_n\frac{\exp(S(Y_n))}{\exp(S(Y_{\tau_n}))}, \tag{3.24}$$

where all terms in Equation (3.24) can be determined from $\tilde{d}_n$. Therefore, $\tilde{\pi}_n^*$ can also be calculated via Equation (3.15).

At last, as directed by the execution order in Equation (3.18), we next examine variables a_n, b_n and verify that they indeed are functions of $\tilde{d}_n$. This is very similar to what we have done with μ_n^{az}.

Take $a_n(\tilde{d}_n)$ for example. If $n+1$ is not an interaction time, according to its definition (3.17), we have

$$a_n(d_n) = \sum_{Y_{n+1}\in\mathcal{Y}}\int_{z\in\mathbb{R}}(1 + r_{n+1} + (z - r_{n+1})\tilde{\pi}_n^*(\tilde{d}_n))a_{n+1}(\tilde{d}_{n+1})f_{Z_{n+1}}(z)\mathrm{d}z$$

where $\tilde{d}_{n+1}$ follows Equation (3.20). If $n+1$ is an interaction time, then we have

$$a_n(d_n) = \sum_{Y_{n+1}\in\mathcal{Y}}\int_{\varepsilon\in\mathbb{R}}\int_{z\in\mathbb{R}}(1 + r_{n+1} + (z - r_{n+1})\tilde{\pi}_n^*(\tilde{d}_n))a_{n+1}(\tilde{d}_{n+1})f_{Z_{n+1}}(z)f_{\varepsilon}^{(\phi)}(\varepsilon)\mathrm{d}z\mathrm{d}\varepsilon$$

where $\tilde{d}_{n+1}$ follows Equation (3.23). $b_n(d_n)$ can be evaluated in the same way.

Hence, we have shown that all the terms required to calculate the optimal strategy are functions of $\tilde{d}_n$, thus have completed the proof.

□

Corollary 3.3.2. *One only need to discretize the space of $\tilde{\mathcal{D}}_n$ instead of $\mathcal{D}_n$ to find the optimal strategy given any possible state. If the step size of discretizing cumulative returns to be ΔZ and the step size of discretizing risk aversion to be $\Delta\xi$, the number of grid points to be discretized in total for any n will be*

$$O\left(\frac{|\mathcal{Y}|^2}{(\Delta Z)^2\Delta\xi}\right) \tag{3.25}$$

which does not increase in n and hence is a significant improvement of the complexity that increases exponentially with n in Equation (3.14).

Combining the decomposition in Equation (3.15) and the low-dimensional discretization in Theorem 3.3.1, the backwards induction algorithm is now complete and practical to execute, which is described in Algorithm 2.

Algorithm 2 Numerical evaluation of optimal strategy via backwards induction and discretization.
This is a direct result of Equation (3.15) and Theorem 3.3.1.

- For $n = T-1, T-2, \dots, 0$:
  - For the $O\left(\frac{|\mathcal{Y}|^2}{(\Delta Z)^2 \Delta\xi}\right)$ many tuples of possible $\tilde{d}_n$:
    * Calculate values $\mu_n^a(\tilde{d}_n), \mu_n^{az}(\tilde{d}_n), \mu_n^{bz}(\tilde{d}_n), \mu_n^{bz^2}(\tilde{d}_n), \pi_n^*(\tilde{d}_n), a_n(\tilde{d}_n), b_n(\tilde{d}_n)$ as shown in the proof of Theorem 3.3.1 via integration over the future economy state Y_{n+1}, market return Z_{n+1} and risk aversion idiosyncratic shock random variable ε_{n+1}.

3.3.3 Approximation for faster computation

Although Theorem 3.3.1 has significantly reduced the computational complexity to evaluate the optimal strategy via backwards induction, we further optimize for faster computation. Note that for each n and for each $\tilde{d}_n$, we need to calculate six integrals (or even double integration if $n+1$ is an interaction time) for $\mu_n^a, \mu_n^{az}, \mu_n^{bz}, \mu_n^{bz^2}, a_n, b_n$. We can avoid most integrals by approximating X_{n+1} as its mean value, μ_{n+1}. In this way, we replace Equation (3.19) with

$$\mu_n^{az}(\tilde{d}_n) = \sum_{Y_{n+1}\in\mathcal{Y}} P_{Y_n,Y_{n+1}} a_{n+1}(\tilde{d}_{n+1}|_{Z_{n+1}=\mu_{n+1}})(\mu_{n+1} - r_{n+1}) \tag{3.26}$$

where $\tilde{d}_{n+1}|_{Z_{n+1}=\mu_{n+1}}$ is Equation (3.20) with Z_{n+1} substituted with μ_{n+1}. Also, we replace Equation (3.21) with

$$\mu_n^{az}(\tilde{d}_n) = \sum_{Y_{n+1}\in\mathcal{Y}} P_{Y_n,Y_{n+1}} \int_{\varepsilon\in\mathbb{R}} a_{n+1}(\tilde{d}_{n+1}|_{Z_{n+1}=\mu_{n+1}})(\mu_{n+1} - r_{n+1}) f_\varepsilon^{(\phi)}(\varepsilon)\mathrm{d}\varepsilon \tag{3.27}$$

$\tilde{d}_{n+1}|_{Z_{n+1}=\mu_{n+1}}$ is Equation (3.23) with Z_{n+1} substituted with μ_{n+1}.

Note that Capponi et al. (2022) did not mention that such approximation is a little bit more complicated for terms where Z_{n+1} is not linear, such as $\mu_n^{bz^2}$ and b_n, since the expectation of square terms cannot be simply approximated via substitution. To be more specific, we have

$$\mu_n^{bz^2}(\tilde{d}_n) = \mathrm{E}[b_{n+1}(d_{n+1}|_{Z_{n+1}=\mu_{n+1}}) \cdot ((\mu_{n+1} - r_{n+1})^2 + \sigma_{n+1}^2)] \tag{3.28}$$

$$b_n(\tilde{d}_n) = \mathrm{E}[\{\tilde{\pi}_n^*(\tilde{d}_n)^2\sigma_{n+1}^2 + [\tilde{\pi}_n^*(\tilde{d}_n)(\mu_{n+1} - r_{n+1}) + 1 + r_{n+1}]^2\} b_{n+1}(d_{n+1}|_{Z_{n+1}=\mu_{n+1}})], \tag{3.29}$$

where the expectation can be calculate via integration over Y_{n+1} and ε (for interaction times) as usual. Equation (3.28) is due to

$$\mathrm{E}[(Z_{n+1} - r_{n+1})^2] = \mathrm{E}[Z_{n+1} - r_{n+1}]^2 + \mathrm{Var}[Z_{n+1} - r_{n+1}] = (\mu_{n+1} - r_{n+1})^2 + \sigma_{n+1}^2,$$

while similarly, Equation (3.29) is due to

$$\begin{aligned}
&\mathrm{E}[(1 + r_{n+1} + (Z_{n+1} - r_{n+1})\tilde{\pi}_n^*(\tilde{d}_n))^2] \\
=&\mathrm{E}[1 + r_{n+1} + (Z_{n+1} - r_{n+1})\tilde{\pi}_n^*(\tilde{d}_n)]^2 + \mathrm{Var}[1 + r_{n+1} + (Z_{n+1} - r_{n+1})\tilde{\pi}_n^*(\tilde{d}_n)] \\
=&(1 + r_{n+1} + (\mu_{n+1} - r_{n+1})\tilde{\pi}_n^*(\tilde{d}_n))^2 + \pi_n^*(\tilde{d}_n)^2\sigma_{n+1}^2.
\end{aligned}$$

By applying approximation to integrate over Z_{n+1}, we completely get rid of integration if $n+1$ is not an interaction time, and we only have single integration over ε when $n+1$ is an interaction time.

3.4 Estimation of client's personalized parameters

Capponi et al. (2022) assume that the robo-advisor knows the client's parameters (as shown in Table 3.1) that define the client's risk aversion process (i.e. full knowledge of the distribution of the stochastic process). However, such knowledge is not realistic to have in practice. We describe a method to estimate client's personalized parameters that defines the client's risk aversion process, i.e. γ_0, the client's initial risk aversion, α, the time discount factor for client's risk aversion; β, the behavioral bias factor describing the client's trend-chasing bias; p_ε and σ_ε, the parameters describing idiosyncratic shocks to the client's risk aversion.

The general idea is to create a testing environment with complete and perfect information so that the client can make informed decisions of portfolio selection. If we assume that the client act optimally, one can learn the client's personalized parameters by observing the client's actions. This is an *inverse reinforcement learning* problem (Ng & Russell, 2000) where we are to observe the agent's interactions with the environment and infer the agent's internal states and parameters given that the agent will make optimal choices, similar to the setup of Alsabah et al. (2021). To guarantee that the parameters estimated in the testing environment apply to the robo-advising process described earlier, the market and client model in the testing environment follows the models defined in Section 3.1, with some simplification. There exists a risk-free asset with known return rate r and a risky asset whose return Z is drawn from $\mathcal{N}(\mu, \sigma^2)$. At time 0, the client has initial wealth of value X_0. At each time n, the client chooses π_n, the amount of wealth invested in the risky asset, and hence the client's wealth at time $n+1$, X_{n+1}, will be $(1+r)X_n + (Z_{n+1} - r)\pi_n$. We assume that the client acts optimally and maximizes the mean-variance objective variance with risk aversion

$$\gamma_n^T = \gamma_n^C \gamma_n^Z = e^{\eta_n} \gamma_n^{\mathrm{id}} \gamma_n^Y \gamma_n^Z.$$

Note that we consider the behavioral bias parameter γ_n^Z as the client's will display trend-chasing mindset as he or she optimizes the portfolio over time.

We let the robo-advisor observe the client's decisions for N rounds, i.e., $n = 0, 1, \ldots, N-1$, and the robo-advisor estimates client's parameters as described in the following subsections. For the convenience of some calculations later, we let N to be an odd integer.

3.4.1 Observing biased risk aversion via interactions

Assume that the client will act optimally in term of maximizing the mean-variance objective variance, one can uniquely identify the risk aversion parameter by observing client's portfolio selection.

Theorem 3.4.1. *At time n, if the client chooses π_n^* such that π_n^* maximizes the mean-variance objective with risk aversion γ_n^T*

$$\mathrm{E}\left[\frac{X_{n+1}(\pi_n) - X_n}{X_n}\right] - \frac{\gamma_n^T}{2}\mathrm{Var}\left[\frac{X_{n+1}(\pi_n) - X_n}{X_n}\right],$$

then the robo-advisor can infer the risk aversion used in such decision making, i.e., γ_n^T, by observing the optimal action π_n^*. More specifically,

$$\gamma_n^T = \frac{(\mu - r)X_n}{\sigma^2 \pi_n^*}.$$

Proof. The client's wealth X_{n+1} of time $n+1$ given the strategy π_n of time $n+1$ is

$$(1+r)X_n + (Z_{n+1} - r)\pi_n. \tag{3.30}$$

Therefore, we have

$$\mathrm{E}[X_{n+1}] = (1+r)X_n + (\mu - r)\pi_n,$$
$$\mathrm{Var}[X_{n+1}] = \pi_n^2 \sigma^2.$$

Substituting back, we have the mean-variance objective function given by

$$\mathrm{E}\left[\frac{X_{n+1}-X_n}{X_n}\right]-\frac{\gamma_n^T}{2}\mathrm{Var}\left[\frac{X_{n+1}-X_n}{X_n}\right]=\frac{rX_n+(\mu-r)\pi_n}{X_n}-\frac{\gamma_n^T}{2}\frac{\pi_n^2\sigma^2}{X_n^2}.$$

Differentiating w.r.t. π_n, the objective function takes its maximum if and only if $\pi_n = \frac{X_n(\mu-r)}{\gamma_n^T\sigma^2}$. Therefore, if π_n^* is the optimal strategy to maximize the mean-variance objective function, the risk aversion γ_n^T used in the objective function is uniquely identifiable, given by $\frac{(\mu-r)X_n}{\pi_n^*\sigma^2}$. □

3.4.2 Estimating client's personalized parameters via the observed biased risk aversion

Theorem 3.4.1 shows that observing the client's optimal actions is equivalent to observing the client's risk aversion parameter used in the decision making process. From this point, we assume that the robo-advisor observes γ_n^T, $n = 0, 1, \ldots, N-1$, and aims to estimate the parameters that determine this stochastic process, i.e., $\gamma_0, \alpha, \beta, p_\varepsilon, \sigma_\varepsilon$. We will repeatedly apply *method of moments* (Bowman & Shenton, 2004) to estimate these parameters.

Theorem 3.4.2. *Let the number of testing rounds, N, be an odd positive integer. Given complete information on market dynamics and the client's risk aversion process during the testing rounds, γ_n^T, $n = 0, 1, \ldots, N-1$, let $Q_n = \ln(\gamma_n^T/\gamma_{n-1}^T)$ and $\Delta Z_n = Z_n - Z_{n-1}$ for $n = 1, 2, \ldots, N-1$.* [2] *The personalized parameters $\gamma_0, \alpha, \beta, \sigma_\varepsilon, p_\varepsilon$ can be estimated as*

$$\begin{aligned}
\hat{\gamma}_0 &= \gamma_0^T,\\
\hat{\beta} &= \frac{Q_1+Q_3+\ldots Q_{N-2}-Q_2-Q_4-\ldots-Q_{N-1}}{\Delta Z_2+\Delta Z_4+\ldots+\Delta Z_{N-1}-\Delta Z_1-\Delta Z_3-\ldots-\Delta Z_{N-2}},\\
\hat{\alpha} &= \frac{Q_1+Q_2+\ldots Q_{N-1}+\hat{\beta}(Z_{N-1}-Z_0)}{N-1},\\
\hat{\sigma}_\varepsilon &= \sqrt{\frac{\hat{\mu}_4}{3\hat{\mu}_2}},\\
\hat{p}_\varepsilon &= \frac{\hat{\mu}_2}{\hat{\sigma}_\varepsilon^2},
\end{aligned}$$

[2] Z_0 has not been defined. For the scope of this subsection, we set Z_0 to be $\mu(Y_0)$, the mean value for Z_1.

where

$$\hat{\mu}_2 = \frac{1}{N-1}\sum_{n=1}^{N-1}(Q_n + \hat{\beta}\Delta Z_n - \hat{\alpha})^2$$
$$\hat{\mu}_4 = \frac{1}{N-1}\sum_{n=1}^{N-1}(Q_n + \hat{\beta}\Delta Z_n - \hat{\alpha})^4.$$

Proof. This theorem immediately follows the following lemmas.

Lemma 3.4.3. *Let the number of testing rounds, N, be an odd positive integer. Given complete information on market and client's risk aversion process γ_n^T, $n = 0, 1, \ldots, N-1$, define $Q_n, \Delta Z_n$ as in Theorem 3.4.2. The personalized parameters γ_0, α, β can be estimated as*

$$\begin{aligned}
\hat{\gamma}_0 &= \gamma_0^T \\
\hat{\beta} &= \frac{Q_1 + Q_3 + \ldots Q_{N-2} - Q_2 - Q_4 - \ldots - Q_{N-1}}{\Delta Z_2 + \Delta Z_4 + \ldots + \Delta Z_{N-1} - \Delta Z_1 - \Delta Z_3 - \ldots - \Delta Z_{N-2}} \\
\hat{\alpha} &= \frac{Q_1 + Q_2 + \ldots Q_{N-1} + \hat{\beta}(Z_{N-1} - Z_0)}{N-1}.
\end{aligned}$$

Moreover, conditioning on the event that $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{N-1}$ are all zeros[3]*, the estimates are accurate.*

Proof. We first express γ_n^T as a process dependent on these parameters:

$$\gamma_0^T = \gamma_0, \tag{3.31}$$
$$\frac{\gamma_{n+1}^T}{\gamma_n^T} = e^{\eta_{n+1}-\eta_n}\frac{\gamma_{n+1}^{id}}{\gamma_n^{id}}\frac{\gamma_{n+1}^Y}{\gamma_n^Y}\frac{\gamma_{n+1}^Z}{\gamma_n^Z} = e^{\alpha}e^{\varepsilon_{n+1}}e^{-\beta(Z_{n+1}-Z_n)}, \quad n = 0, 1, \ldots, N-2. \tag{3.32}$$

Note that (3.32) is correct for $n = 0$ because we manually set Z_0 to be the mean of Z_1, and that $\gamma_0^Z = 1$ because there's no investment history to bias the client.

Therefore, the robo-advisor can have an accurate estimation of γ_0 by only observing the client's action at time 0 (since at initialization, there's no bias at all and the communicated risk aversion is indeed the actual risk aversion). For $n = 1, 2, \ldots, N-1$, $Q_n = \ln(\gamma_n^T/\gamma_{n-1}^T)$ and $\Delta Z_n = Z_n - Z_{n-1}$ are both observable. The robo-advisor aims to estimate α, β from

$$\alpha + \varepsilon_n - \beta\Delta Z_n = Q_n, \quad n = 1, 2, \ldots, N-1$$

We first estimate β, then α, and finally ε (in the next lemma), by method of moments. First,

[3]This is a high probability event if setting N to be small.

for β, note that

$$\begin{aligned}
\varepsilon_2 - \varepsilon_1 &= \beta(\Delta Z_2 - \Delta Z_1) + Q_2 - Q_1, \\
\varepsilon_4 - \varepsilon_3 &= \beta(\Delta Z_4 - \Delta Z_3) + Q_4 - Q_3, \\
&\dots \\
\varepsilon_{N-1} - \varepsilon_{N-2} &= \beta(\Delta Z_{N-1} - \Delta Z_{N-2}) + Q_{N-1} - Q_{N-2}
\end{aligned} \tag{3.33}$$

where the LHS of each equation in (3.33) is an i.i.d. sample of a random variable with zero mean (due to the distribution of ε_n). Hence, one can estimate

$$\hat{\beta} = \frac{Q_1 + Q_3 + \dots Q_{N-2} - Q_2 - Q_4 - \dots - Q_{N-1}}{\Delta Z_2 + \Delta Z_4 + \dots + \Delta Z_{N-1} - \Delta Z_1 - \Delta Z_3 - \dots - \Delta Z_{N-2}}$$

by letting the sum of the RHS formulas in (3.33) zero and solving $\hat{\beta}$.

Then, to estimate α, note that

$$\begin{aligned}
\varepsilon_1 &= Q_1 + \beta\Delta Z_1 - \alpha \\
\varepsilon_2 &= Q_2 + \beta\Delta Z_2 - \alpha \\
&\dots \\
\varepsilon_{N-1} &= Q_{N-1} + \beta\Delta Z_{N-1} - \alpha
\end{aligned} \tag{3.34}$$

where the LHS of each equation in (3.34) an i.i.d. sample of a random variable with zero mean (due to the distribution of ε_n). Hence one can estimate

$$\hat{\alpha} = \frac{Q_1 + Q_2 + \dots Q_{N-1} + \hat{\beta}(Z_{N-1} - Z_0)}{N-1}.$$

by setting the sum of the RHS formulas in (3.34) zero and solving $\hat{\alpha}$.

At last, if all ε_n are zeros, then the LHS of each equation in (3.33) and (3.34) are precisely zeros instead of i.i.d. samples of a random variable with zero mean, rendering the solved $\hat{\alpha}$ and $\hat{\beta}$ precise. □

Lemma 3.4.4. *Following the notations introduced in Theorem 3.4.2 and Lemma 3.4.3, given complete information on market dynamics and client's risk aversion process during the testing period* γ_n^T*,* $n = 0, 1, \dots, N-1$*, and the accurate values of* α *and* β*. The personalized parameters* σ_ε *and* p_ε *can be estimated as*

$$\hat{\sigma}_\varepsilon = \sqrt{\frac{\hat{\mu}_4}{3\hat{\mu}_2}}, \quad \hat{p}_\varepsilon = \frac{\hat{\mu}_2}{\hat{\sigma}_\varepsilon^2},$$

where

$$\begin{aligned}\hat{\mu}_2 &= \frac{1}{N-1}\sum_{n=1}^{N-1}(Q_n + \beta\Delta Z_n - \alpha)^2 \\ \hat{\mu}_4 &= \frac{1}{N-1}\sum_{n=1}^{N-1}(Q_n + \beta\Delta Z_n - \alpha)^4\end{aligned} \tag{3.35}$$

are sample means of ε_n of order 2 and order 4.

Proof. Given $Q_n, \beta, \Delta Z_n, \alpha$, (3.34) provides $N-1$ i.i.d. samples of ε_n which admits the following distribution

$$\varepsilon_n = \begin{cases} \mathcal{N}(0, \sigma_\varepsilon^2), & \text{with probability } p_\varepsilon \\ 0, & \text{with probability } 1 - p_\varepsilon \end{cases}.$$

To apply the method of moments for estimation of parameters p_ε and σ_ε, we need to evaluate the expectation values of ε_n^2 and ε^4. This is because for odd orders, the expectation will always be zero, which is not a function of the parameters p_ε and σ_ε.

To calculate the expectation values, we first express ε_n as XY where X and Y are independent random variables, and $X \sim \text{Bernoulli}(p_\varepsilon)$ and $Y \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. Then we have

$$\begin{aligned}\mathrm{E}[\varepsilon_n^2] = \mathrm{E}[X^2Y^2] = \mathrm{E}[X^2]\mathrm{E}[Y^2] = \mathrm{E}[X](\mathrm{Var}[Y] + \mathrm{E}[Y]^2) = p_\varepsilon\sigma_\varepsilon^2 \\ \mathrm{E}[\varepsilon_n^4] = \mathrm{E}[X^4Y^4] = \mathrm{E}[X^4]\mathrm{E}[Y^4] = 3p_\varepsilon\sigma_\varepsilon^4.\end{aligned}$$

At last, solving $\hat{\mu}_2 = p_\varepsilon\sigma_\varepsilon^2$ and $\hat{\mu}_4 = 3p_\varepsilon\sigma_\varepsilon^4$, we have estimations

$$\hat{\sigma}_\varepsilon = \sqrt{\frac{\hat{\mu}_4}{3\hat{\mu}_2}}, \quad \hat{p}_\varepsilon = \frac{\hat{\mu}_2}{\hat{\sigma}_\varepsilon^2},$$

where $\hat{\mu}_2$ is the sample mean of ε_n^2 and $\hat{\mu}_4$ is the sample mean of ε_n^4, as defined in (3.35). □

Combining Lemma 3.4.3 and Lemma 3.4.4, we have completed the proof of Theorem 3.4.2.

□

Chapter 4

Experiments and discussions

4.1 Experimental setup

4.1.1 Configurations for market and client models

For empirical evaluation, we consider the following market with two economy states $\mathcal{Y} = \{1, 2\}$. 1 is a bad economy and 2 is a good one. Based on Chauvet and Hamilton (2006), we set the transition matrix for the economy states is given by

$$P = \begin{bmatrix} 0.95 & 0.05 \\ 0.1 & 0.9 \end{bmatrix}$$

indicating that there's a high probability that the economy state stays the same but there's also positive probability to escape. The initial economy is set to be the bad one, i.e. $y_0 = 1$.

In the bad economy, the risk-free asset has zero return rate, while the risky asset has return drawn from $\mathcal{N}(0.01, 0.05^2)$. In the good economy, the risk-free asset has a return rate of 0.01 and the risky asset has return drawn from $\mathcal{N}(0.03, 0.1^2)$. Using the notations introduced in Section 3.1.1, we have

$$r(y) = \begin{cases} 0.00, & y = 1 \\ 0.01, & y = 2 \end{cases}, \qquad \mu(y) = \begin{cases} 0.01, & y = 1 \\ 0.03, & y = 2 \end{cases}, \qquad \sigma(y) = \begin{cases} 0.05, & y = 1 \\ 0.1, & y = 2 \end{cases}.$$

Unless otherwise specified, we set the time horizon T to be 12 and the interaction interval to be 3, indicating a one-year robo-advising product with quarterly interactions.

As for the client, unless otherwise specified, we set the personalized parameters as follows

$$\gamma_0 = 4, \quad \alpha = 0.01, \quad \beta = 4, \quad p_\varepsilon = 0.05, \quad \sigma_\varepsilon = 0.64.$$

Given the configuration for the market, client and the robo-advisor models, we run backwards induction as a one-time pre-processing step to find the optimal strategy at all timestamps given all possible states $\tilde{d}$. Then, for all the experiments we conduct, we simulate the market and run the robo-advising algorithm for an extensive number of trials and report the mean values for statistical significance.

4.1.2 Implementation details

We implement our robo-advising algorithm in Python[1] and SciPy (Virtanen et al., 2020) and run all experiments on Linux 20.04 LTS in a consumer-level machine equipped with Ryzen 9 5900HS and 32 GB of RAM. Our implementation follows Algorithm 2 with approximation as described in Section 3.3.3. A single run of the backwards induction algorithm to find values at all grid points of $\tilde{\mathcal{D}}_n$ took less than three hours. Although the algorithm has been described in great detail in Section 3.3, we specify the following important implementation details.

Discretization configuration

As suggested by Theorem 3.3.1, for each n, we need to evaluate the values of $\mu_n^a, \mu_n^{az}, \mu_n^{bz}, \mu_n^{bz^2}, \tilde{\pi}_n^*$, a_n, b_n given all possible states $\tilde{d}_n \in \tilde{\mathcal{D}}_n$. Therefore, we discretize the space $\tilde{\mathcal{D}}_n$:

1. For economy states at current time and the most recent interaction time, which is discrete, we naturally enumerate all the possible values in $\mathcal{Y} = 1, 2$.

2. For cumulative excess returns, we consider a range of $[-0.5, 0.5]$ and discretize the range with a grid size of 0.1.

3. For risk aversion values, we consider a range of $(0, 10]$ and discretize the range with a grid size of 0.5.

This is a reasonable discretization grid because in our experiments, as the values almost never fall out of the specified ranges.

[1]https://www.python.org/

In backwards induction, for all n, we fill in the values of $\mu_n^a, \mu_n^{az}, \mu_n^{bz}, \mu_n^{bz^2}, \tilde{\pi}_n^*, a_n, b_n$ at all grid points in $\tilde{\mathcal{D}}_n$. For querying data at states which are not a grid point, we use linear interpolation (Meijering, 2002) to construct reasonable new data points from values at neighbor grid points.

Approximate integration over ε

Note that as pointed in the proof of Theorem 3.3.1, one need to integrate over the randomness of $\varepsilon = \varepsilon_{\tau_n+1} + \ldots + \varepsilon_{n+1}$ whose distribution is a ϕ-fold convolution of ε_n whose distribution is defined in Equation (3.5). Note that this is not trivial to integrate because the probability density function for each ε_n is not continuous, i.e.

$$f_{\varepsilon_n}(x) = \begin{cases} \frac{p_\varepsilon}{\sigma_\varepsilon}\varphi\left(\frac{x}{\sigma_\varepsilon}\right), & x \neq 0 \\ (1 - p_\varepsilon) + \frac{p_\varepsilon}{\sigma_\varepsilon\sqrt{2\pi}}, & x = 0 \end{cases}$$

where $\varphi(\cdot)$ is the probability density function for the standard Gaussian $\mathcal{N}(0, 1)$.

We take an approximation approach by first evenly taking many values from the range of ε_n and treat ε_n as a discrete random variable. We calculate the probability mass function for this approximate discrete random variable, and apply Fast Fourier Transform convolution (Brigham, 1988) to find the probability mass function for $\varepsilon = \varepsilon_{\tau_n+1} + \ldots + \varepsilon_{n+1}$, and using the PMF to calculate the expectation over the randomness of (now discrete) ε. Thus, we transform a difficult integration operation into a weighted sum. To be more specific, we evenly take 65 points from $[-2, 2]$ to form the discrete random variables. We take odd number of points to make sure to include the probability of the outlier point 0.

4.2 Optimal investment strategy

Experimental configuration follows Section 4.1. We report in Figure 4.1 the optimal proportion of wealth allocated to the risky asset, $\tilde{\pi}_n^d$, as a function of client's risk aversion, γ_n^C, six months after the start of the investment process, i.e., $n = 6$. Each scatter point in the figure represents a trial of market simulation and robo-advisor's response. Green points represents $Y_n = 1$, indicating a bad economy at the reported timestamp and red points represents $Y_n = 2$, indicating a good one. We also plot (in solid lines) the optimal strategy at $n = 6$ as a one-stage mean-variance optimization without behavioral bias, which, as shown in the proof of Theorem 3.4.1, is given by

$$\tilde{\pi}_n^C(\gamma_n^C) := \frac{\mu(Y_n) - r(Y_n)}{\sigma^2(Y_n)\gamma_n^C}.$$

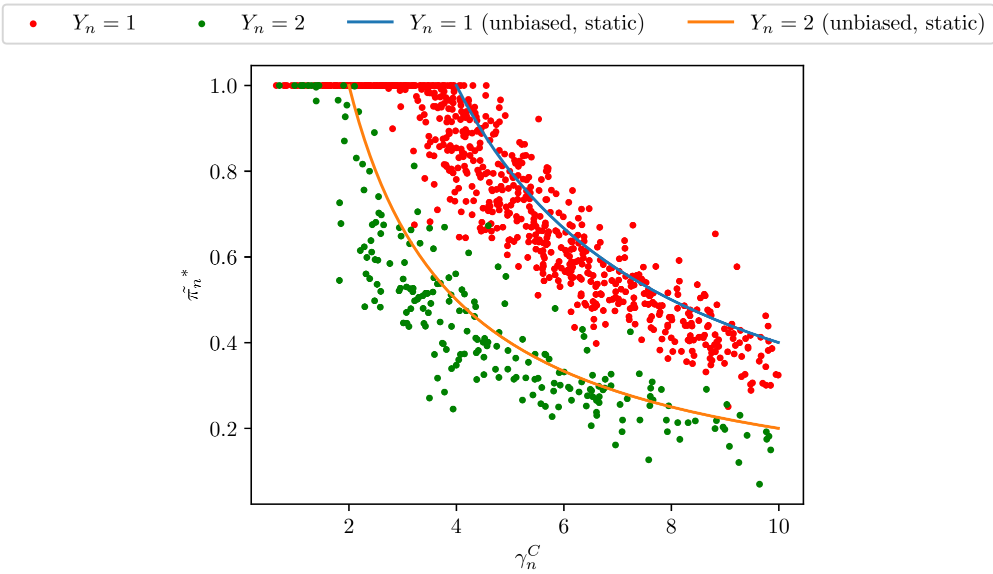

Figure 4.1: Optimal investment strategy as a function of client's risk aversion.

Hence, $\tilde{\pi}_n^C$ is greater when $Y_n = 1$ than $Y_n = 2$ (i.e., the solid blue line is above the solid orange line) because $\tilde{\pi}_n^C = 4/\gamma_n^C$ when $Y_n = 1$ while $\tilde{\pi}_n^C = 2/\gamma_n^C$ when $Y_n = 2$.

First, we note that there are more scatter points with $Y_n = 1$ than with $Y_n = 2$. This is because the simulation starts at $Y_0 = 1$ and according to the transition matrix P, there's a much higher probability of staying in the same economy state than escaping. Also, we note that even though with same γ_n^C and economy state Y_n, there are points with different optimal strategies, this is because the optimal strategy $\tilde{\pi}_n^*$ given by the robo-advisor also depends on the previous market performance and the client's behavioral bias. In each case of Y_n, Figure 4.1 shows clear trends of $\tilde{\pi}_n^*$ inversely proportional to γ_n^C, which agrees with the common sense that higher risk aversion leads to less proportion of wealth allocated to the asset. Figure 4.1 also shows that all $\tilde{\pi}_n^*$ values are concentrate around their respective $\tilde{\pi}_n^C$ values with the same Y_n, which makes sense because the robo-advisor's optimal strategies are approximately behaviourally biased client's optimal strategies. This also verifies the correctness of our implemented algorithm.

At last, for both cases of Y_n, we notice that $\tilde{\pi}_n^*$ is more likely to be smaller than $\tilde{\pi}_n^C$ rather than larger (for example, most of the red scatter points around the blue solid line are below the line rather than above), indicating that the behavioral bias is more likely to increase client's risk aversion, i.e., $\gamma_n^Z > 1$. This is because

$$E[\gamma_n^Z] = E\left[\exp\left(-\frac{\beta}{\phi}\sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1}-\mu_{k+1})\right)\right]$$

where each term $Z_{k+1} - \mu_{k+1}$ follows a normal distribution with zero mean and positive variance. Hence, $\beta/\phi \cdot \sum_{k=\tau_n-\phi}^{\tau_n-1}(Z_{k+1} - \mu_{k+1}))$ follows a normal distribution of zero mean and positive variance, denoted as σ_z^2. Therefore, we have $\mathrm{E}[\gamma_n^Z] = \exp(\sigma_z^2/2) > 1$, thus magnifying risk aversion and leading to lower proportion of asset allocated to the risky asset on average.

Figure 4.1 is consistent with the findings of Capponi et al. (2022), but we also include discussions on the effects of economy states by distinguishing $Y_n = 1$ and $Y_n = 2$ scenarios.

4.3 Robo-advisor's personalization measure

As shown in Section 4.2, the divergence between the robo-advisor's responding optimal strategy, $\tilde{\pi}_n^*$, and the client's single-stage greedy optimal strategy with unbiased risk aversion γ_n^C, $\tilde{\pi}_n^C$, is caused by the client's behavioral bias. Naturally, one would like to examine the magnitude of such divergence as a measure of the robo-advisor's personalization measure. Since the divergence between $\tilde{\pi}_n^*$ and $\tilde{\pi}_n^C$ is originated from the difference in γ_n^C and γ_n^R, Capponi et al. (2022) propose the following personalization measure as a function of interaction interval ϕ and behavioral bias parameter β

$$\mathcal{R}(\phi, \beta) := \mathrm{E}\left[\frac{1}{T}\sum_{n=0}^{T-1}\left|\frac{\gamma_n^C}{\gamma_n^R} - 1\right|\right]. \tag{4.1}$$

Before experiments, we have a few hypothesis on the properties of $\mathcal{R}$:

1. Since the personalization measure $\mathcal{R}$ measures the difference between γ_n^C and γ_n^R, that is, it measures $|\gamma_n^Z|$, the magnitude of the behavioral bias, we expect the lower level of personalization (i.e., higher $\mathcal{R}$ value) to decrease with an increase in β, which indicates stronger behavioral bias.

2. The robo-advisor estimates γ_n^C via interactions with the client, hence, we may expect higher level of personalization (i.e., smaller $\mathcal{R}$ value) with smaller ϕ and more frequent interactions. However, interactions are also the reason that gives rise to the existence of behavioral bias, and too frequent interactions will cause a surge in bias γ_n^Z and hence results in lower personalization level.

To confirm the hypothesis, we run the robo-advising algorithm and report its personalization measure $\mathcal{R}(\phi, \beta)$ in Figure 4.2, with varying $\phi \in \{1, 2, 3, \ldots, 8\}$ and $\beta \in \{0, 2, 4\}$ values. Note that to better demonstrate the effects of ϕ, we run for an extended period time with a time horizon of 3 years, i.e., $T = 36$. Otherwise, for $T = 12$ as in previous experiments, $\phi = 6, 7, 8$ will all lead to only two interactions.

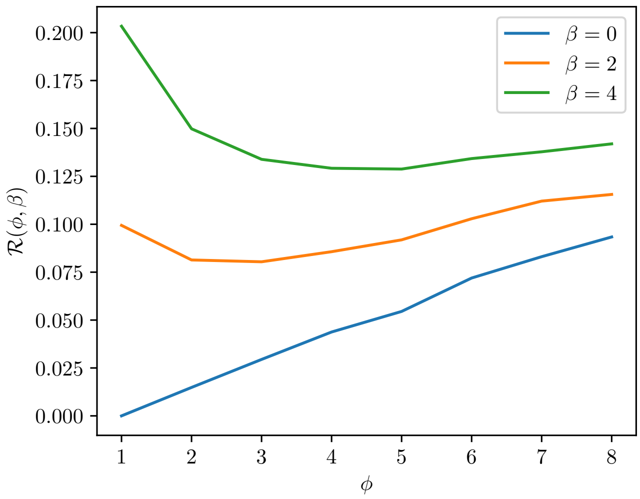

Figure 4.2: Robo-advisor's personalization measure $\mathcal{R}(\phi, \beta)$ as a function of ϕ and β on a time horizon of $T = 36$.

Effects of β. As reported in Figure 4.2, one can clearly note that given the same interaction schedule, robo-advisors with higher β values will have higher $\mathcal{R}$, resulting in weaker personalization. This is because higher β values means that the client is more behaviorally biased and hence the γ_n^R is more biased away from γ_n^C. When $\beta = 0$, $\gamma_n^Z \equiv 0$ and there is no behavioral bias. In this case, the only reason that causes γ_n^R to be different from γ_n^C is the estimation error of the robo-advisor when modelling the client's risk aversion process. This component is the same regardless of the value of β, and for larger β there will be stronger behavioral bias, hence, personalization measure $\mathcal{R}$ increases (i.e., weaker personalization) with β.

Effects of ϕ. We discuss the effects of ϕ when $\beta = 0$ first. In this case, there's no behavioral bias included in γ_n^R and at all n, the robo-advisor maintains its estimation of γ_n^C as γ_n^R and gets corrected on interaction times as the communicated risk aversion has no behavioral bias. Therefore, if $\beta = 0$ and $\phi = 1$, i.e., at all times the robo-advisor estimates γ_n^C as γ_n^R directly from the client's input of unbiased risk, the personalization measure would be zero and the robo-advisor has achieved perfect personalization. When ϕ increases, there are less frequent interactions and the robo-advisor is corrected with client's input of γ_n^C less less. Hence, $\mathcal{R}$ increases as ϕ increases when $\beta = 0$m as shown in Figure 4.2. When $\beta > 0$, we also need to take behavioral bias into consideration. More frequent interactions will let the robo-advisor to adjust its estimation more often, but since behavioral bias is only applied at interaction times, more frequent interactions will also increase the behavioral bias. Therefore, there's a natural trade-off between better estimation of γ_n^C and smaller γ_n^Z when the robo-advisor tries to model $\gamma_n^R = \mathrm{E}[\gamma_n^C]\gamma_N^Z$. As shown in Figure 4.2, for both cases

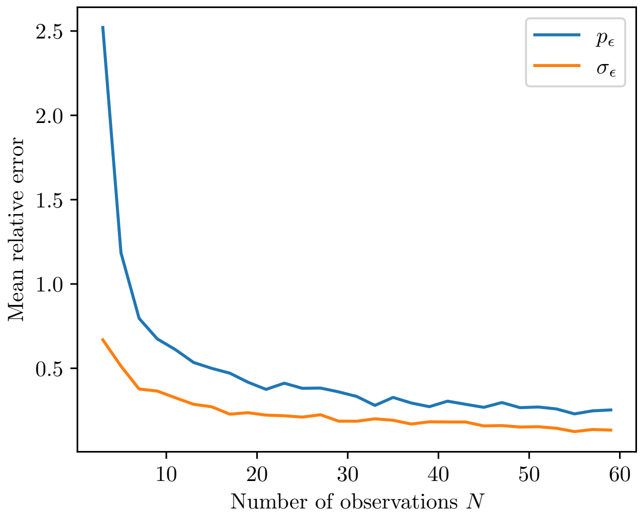

Figure 4.3: Mean relative error of estimations of parameters p_ε and σ_ε.

of $\beta = 2, 4$, we note a decrease in $\mathcal{R}$ when ϕ is small and then an increase in $\mathcal{R}$ when ϕ is larger. The first phase is due to the decrease in behavioral bias caused by less frequent interactions, while the second phase is due to worse estimations of γ_n^R when the robo-advisor do not have sufficient interactions to correct itself. Figure 4.2 shows that although our setting does not adopt an explicit penalty for each interaction as in (Alsabah et al., 2021), there's still an implicit penalty caused of interactions by behavioral bias that only occurs at interaction times. One need to first simulate the robo-advisor algorithm to find out the optimal interaction schedule before allowing it to deploy in the reality. For β, denote the best interaction interval at β is ϕ_beta, then we have is $\phi_0 = 1$, $\phi_2 = 3$ and $\phi_4 = 5$.

4.4 Estimation of client's personalized parameters

At last, we design and conduct experiments to verify the algorithm to estimate the client's personalized parameters as describe in Theorem 3.4.2. In this section, we let the sandbox investment environment to be one with a single economy state and $r = 0.05, \sigma = 0.2, \mu = 0.1$ and fix the ground truth parameters to be $\gamma_0 = 5, \alpha = 0.01, \beta = 4, \sigma_\varepsilon = 1, p_\varepsilon = 0.5$. We experiment with varying number of observations $N \in [3, 61]$.

As pointed out in Lemma 3.4.3, one can obtain accurate estimations of γ_0, α, β by setting N to be very small. Hence, we assume the estimator already has the accurate estimations of these variables and focus on the precision of the estimations of p_ε and σ_ε. We report the mean relative error (MRE) of p_ε and σ_ε estimations w.r.t. to number of observations N in Figure 4.3, which is given by

$$\mathrm{MRE}(p_\varepsilon) = \frac{1}{M}\sum_{i=1}^{M}\left|\frac{\hat{p}_\varepsilon^{(i)} - p_\varepsilon}{p_\varepsilon}\right|$$
$$\mathrm{MRE}(\sigma_\varepsilon) = \frac{1}{M}\sum_{i=1}^{M}\left|\frac{\hat{\sigma}_\varepsilon^{(i)} - \sigma_\varepsilon}{\sigma_\varepsilon}\right|$$

where $M = 100$ is the number of trials for any N and $\hat{p}_\varepsilon^{(i)}$ is the estimation in the i-th trial and $\hat{\sigma}_\varepsilon^{(i)}$ is similarly defined.

It can be seen from Figure 4.3 that the MRE of both estimations converges to zero when $N \to \infty$, validating our method of moments for parameter estimation. One can also notice that the MRE of p_ε is consistently higher than σ_ε. This is because the ground truth value of p_ε is smaller than σ_ε and hence a small absolute error in the estimation will lead to a stronger disturbance to the mean relative error. However, Figure 4.3 also exposes some limitations of our estimation method. Although our method can give accurate estimations when $N \to \infty$, it takes too many observations to reach usable estimations of the parameters, which is not practical in real-world application. This is also a weakness of method of moments for parameter estimation. Novel and more advanced estimation method can be proposed to address this issue in the future.

Chapter 5

Conclusion

References

Alsabah, H., Capponi, A., Ruiz Lacedelli, O., & Stern, M. (2021). Robo-advising: Learning investors' risk preferences via portfolio choices. *Journal of Financial Econometrics*, *19*(2), 369–392.

Beketov, M., Lehmann, K., & Wittke, M. (2018). Robo advisors: quantitative methods inside the robots. *Journal of Asset Management*, *19*(6), 363–370.

Bowman, K. O., & Shenton, L. (2004). Estimation: Method of moments. *Encyclopedia of statistical sciences*, *3*.

Brigham, E. O. (1988). *The fast Fourier transform and its applications*. Prentice-Hall, Inc.

Capponi, A., Olafsson, S., & Zariphopoulou, T. (2022). Personalized robo-advising: Enhancing investment through client interaction. *Management Science*, *68*(4), 2485–2512.

Chauvet, M., & Hamilton, J. D. (2006). Dating business cycle turning points. *Contributions to Economic Analysis*, *276*, 1–54.

Lettau, M., & Ludvigson, S. C. (2010). Measuring and modeling variation in the risk-return trade-off. *Handbook of financial econometrics: Tools and techniques*, 617–690.

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, *7*(1), 77–91. Retrieved 2023-03-23, from `http://www.jstor.org/stable/2975974`

Meijering, E. (2002). A chronology of interpolation: from ancient astronomy to modern signal and image processing. *Proceedings of the IEEE*, *90*(3), 319–342.

Ng, A. Y., & Russell, S. (2000). Algorithms for inverse reinforcement learning. In *ICML* (Vol. 1, p. 2).

Oechssler, J., Roider, A., & Schmitz, P. W. (2009). Cognitive abilities and behavioral biases. *Journal of Economic Behavior & Organization*, *72*(1), 147–152.

Statista. (2023). *Robo-advisors - worldwide.* Retrieved 2023-03-08, from `https://www.statista.com/outlook/dmo/fintech/digital-investment/robo-advisors/worldwide?currency=usd`

Tertilt, M., & Scholz, P. (2018). To advise, or not to advise—how robo-advisors evaluate the risk preferences of private investors. *The Journal of Wealth Management*, *21*(2), 70–84.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., ... SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272. doi: 10.1038/s41592-019-0686-2

Wang, H., & Yu, S. (2021). Robo-advising: Enhancing investment with inverse optimization and deep reinforcement learning. In *2021 20th ieee international conference on machine learning and applications (icmla)* (pp. 365–372).

Wang, H., & Zhou, X. Y. (2020). Continuous-time mean–variance portfolio selection: A reinforcement learning framework. *Mathematical Finance*, *30*(4), 1273–1308.