Risk-Sensitive Policy Optimization via Predictive CVaR Policy Gradient

Ju-Hyun Kim [1] **Seungki Min** [1]

Abstract

This paper addresses a policy optimization task with the conditional value-at-risk (CVaR) objective. We introduce the *predictive CVaR policy gradient*, a novel approach that seamlessly integrates risk-neutral policy gradient algorithms with minimal modifications. Our method incorporates a reweighting strategy in gradient calculation – individual cost terms are reweighted in proportion to their *predicted* contribution to the objective. These weights can be easily estimated through a separate learning procedure. We provide theoretical and empirical analyses, demonstrating the validity and effectiveness of our proposed method.

1. Introduction

In safety-critical applications, the decision makers concern not only their average performance but also their performance in adverse scenarios. Risk-sensitive reinforcement learning (RL) gives a solution for this issue and has been widely adopted across many domains including autonomous driving (Wen et al., 2020), robotic surgery (Pore et al., 2021), finance (Greenberg et al., 2022), etc.

This paper considers the *conditional value-at-risk* (CVaR; also known as average value-at-risk, or expected shortfall) as an objective, which measures the average loss occurring in the worst q-fraction of scenarios. CVaR has long been a popular choice for risk quantification because of its intuitive interpretation along with its nice mathematical properties as a coherent spectral measure (Rockafellar et al., 2000). Accordingly, a considerable effort has been made recently towards the integration of CVaR objective into RL framework, mainly falling into two categories – value-based approaches (Bäuerle & Ott, 2011; Chow et al., 2015; Stanko & Macek, 2019) and policy-based approaches (Tamar et al., 2015b; Rajeswaran et al., 2016; Tamar et al., 2016; Markowitz et al., 2023).

This paper is lined up with the policy-based approaches, which aim to directly optimize the policy through policy gradient. Most existing CVaR policy gradient algorithms studied in prior work implement the following procedure. Aiming to optimize the CVaR_q objective, the algorithm runs a policy multiple times, and calculates the gradient only with the worst q-fraction of sample trajectories, discarding all the other samples. Despite its simplicity, this behavior leads to a low sample efficiency and therefore slow convergence.

To mitigate this issue, we propose the *predictive CVaR policy gradient* that enables the algorithm to utilize all sample trajectories. The main idea is to reweight the individual cost realizations, where the weight on each cost term is the probability of the current sample trajectory belonging to the worst q-fraction of scenarios, predicted at the moment when the cost was incurred.

To better illustrate this idea, let us consider a situation in which the algorithm tries to aggregate the gradient information from five sample trajectories, visualized in Figure 1(a). With $q = 0.2$, the naïve approach uses the worst trajectory only, as highlighted in Figure 1(b). Our method uses the reweighted version of all five trajectories, where the weights are visualized with opacity in Figure 1(c). This features that the decisions should be evaluated individually according to their *expected* contribution to the objective. The worst trajectory should not be overly emphasized, and likewise, non-worst trajectories should not be completely ignored – for example, if two trajectories share the same sample path up some time t, two partial trajectories up to time t should be identically informative regardless of their final outcomes.

Contributions This paper offers a novel reweighting strategy that significantly enhances the sample efficiency of CVaR policy gradient. These weights, which we call *predictive tail probabilities*, are derived through a series of reformulations of the CVaR objective, exploiting its various mathematical properties (Section 3.2). We show that these predictive tail probabilities enjoy a very limited path-dependence (Proposition 3.3), and therefore building a model to estimate them becomes a trivial supervised learning task that is decoupled from the policy optimization task. Similarly

[1]Department of Industrial and Systems Engineering, KAIST, Daejeon, South Korea. Correspondence to: Seungki Min <skmin@kaist.ac.kr>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

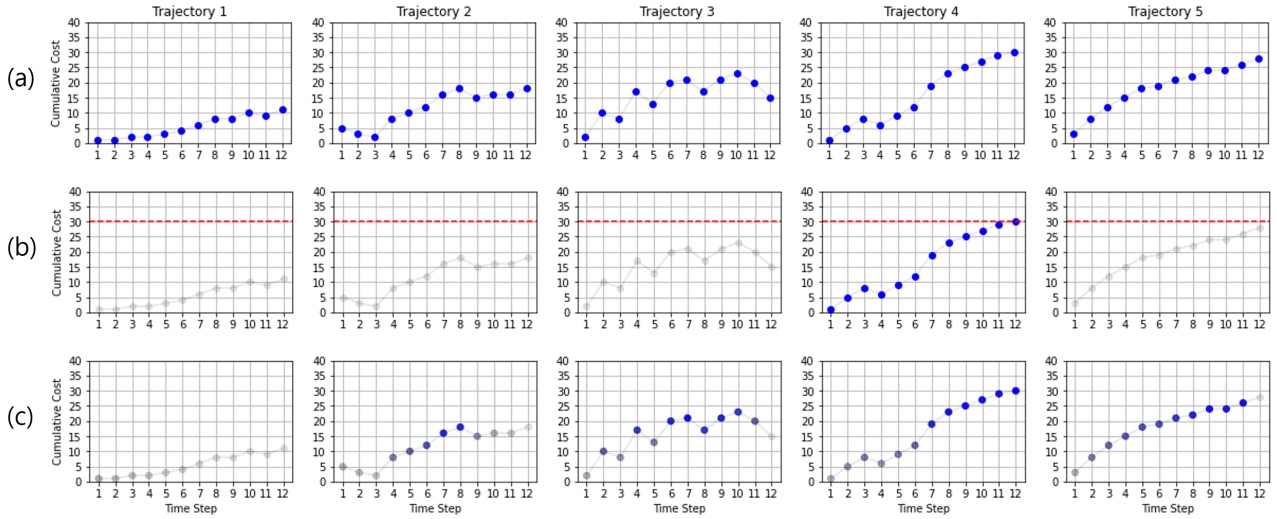

Figure 1. An illustrative example of the proposed reweighting strategy, where five sample trajectories are being processed to compute the gradient ($q = 0.2$). Each plot visualizes the cumulative cost along a trajectory, where the opacity of each point represents the weight applied on each observation: (a) a risk-neutral policy gradient treats all trajectories with equal weights; (b) a naïve CVaR policy gradient only utilizes the worst trajectory, with equal weights across all observations therein; and (c) our predictive CVaR policy gradient assigns different weights on individual observations according to their predictive tail probability.

to actor-critic algorithms, this algorithmic procedure leads to more stable and robust learning outcomes, overcoming the common shortfall of the existing CVaR policy gradient algorithms. Moreover, our method is easily generalized towards the spectral risk measure (Section 3.4).

Related work In the broad landscape of risk-sensitive RL, various performance criteria have been introduced, including exponential utility (Fei et al., 2020; 2021), mean-variance risk functional (Tamar et al., 2012; La & Ghavamzadeh, 2013; Xie et al., 2018), quantile level (Li et al., 2022), and various risk measures such as CVaR (Chow & Ghavamzadeh, 2014; Prashanth, 2014; Chow et al., 2018; Hiraoka et al., 2019; Bastani et al., 2022; Wang et al., 2023), Iterated CVaR (Du et al., 2022; Chen et al., 2023), coherent risk measures (Tamar et al., 2015a; 2016), distortion risk measures (Vijayan & Prashanth, 2021), entropic risk (Borkar & Meyn, 2002; Borkar & Jain, 2014), etc.

Like in the risk-neutral RL literature, the existing CVaR RL algorithms can be categorized as value-based methods and policy-based methods. Value-based methods mainly rely on the Bellman equations formulated on the augmented state space with an extra state variable representing either the running cost (Bäuerle & Ott, 2011; Haskell & Jain, 2015; Miller & Yang, 2017; Bastani et al., 2022) or a notion of remaining risk budget represented as a quantile value (Chow et al., 2015; Pflug & Pichler, 2016; Bonalli et al., 2022; Wang et al., 2023). On the other hand, policy-based methods (policy gradient algorithms) have been studied in Tamar et al. (2015b); Rajeswaran et al. (2016); Tamar et al. (2016); Huang et al. (2021). Although our suggested algorithm would belong to policy-based methods, we leverage the ideas developed in these studies when reformulating the CVaR objective and defining the predictive tail probability process.

Notably, Huang et al. (2021) also introduce reweighting strategies for CVaR policy optimization, sharing the same concern with ours. More specifically, they discuss two schemes – fully path-dependent (multiplicative) importance weights inspired by Tamar et al. (2015b), and state-dependent weights inspired by Liu et al. (2018). However, as they claimed, the former causes 'the magnitude of the gradient to become intractably large in longer horizons' and the latter only applies to the infinite horizon settings. Our reweighting method enjoys the advantage of both schemes in the sense that the predictive tail probabilities are defined to be fully path-dependent while admitting a parsimonious representation so that can be stably estimated.

2. Problem Setup and Preliminaries

We consider a finite-horizon Markov decision process (MDP) specified with a state space $\mathcal{X}$, an action space $\mathcal{A}$, a horizon length T, an initial state $x_1 \in \mathcal{X}$, and a transition kernel p. On each time period $t = 1, \dots, T$, the decision maker (DM) selects an action $A_t \in \mathcal{A}$, pays a cost $C_t \in \mathbb{R}$, and moves onto the next state $X_{t+1} \in \mathcal{X}$. We consider random costs, possibly correlated with the state transition, so that the cost and the next state are drawn from their joint distribution, i.e., $(C_t, X_{t+1}) \sim p(\cdot\,|X_t, A_t)$.

We denote by H_t the history realized prior to making the

t-th decision: for each $t = 1, \dots, T+1$,

$$H_t := (X_1, A_1, C_1, \dots, X_{t-1}, A_{t-1}, C_{t-1}, X_t),$$

which represents all information available to the DM at the moment of deciding action A_t. We use H_{T+1} to denote a sample trajectory (i.e., a sample path, an episode).

Policy space Let $\Pi^{\mathcal{H}}$ be the set of all non-anticipating policies including the randomized ones, and let $\Pi^{\mathcal{X}}$ be the set of all Markov policies such that select actions based only on the current state. A non-anticipating policy $\pi \in \Pi^{\mathcal{H}}$ is a sequence of mappings, $(\pi_t)_{t=1,\dots,T}$, such that π_t maps each history to an action distribution, i.e., $A_t \sim \pi_t(\cdot|H_t)$. A Markov policy $\pi \in \Pi^{\mathcal{X}}$ consists of mappings from state space instead, i.e., $A_t \sim \pi_t(\cdot|X_t)$. Trivially, $\Pi^{\mathcal{X}} \subset \Pi^{\mathcal{H}}$.

This paper focuses on policy optimization over a certain family of Markov policies,[1] denoted by Π^{Θ}, which are parameterized by a multi-dimensional policy parameter $\theta \in \Theta \subseteq \mathbb{R}^d$. Depending on the context, π_t^θ may specify a state-dependent action distribution (i.e., $A_t \sim \pi_t^\theta(\cdot|X_t)$) or a state-dependent action (i.e., $A_t = \pi_t^\theta(X_t)$). We assume that π_t^θ is a mapping differentiable with respect to θ on every state.

Policy optimization with CVaR objective Given a risk level $q \in (0, 1]$, the CVaR_q value of a random cost C is defined as

$$\text{CVaR}_q[C] := \frac{1}{q}\int_0^q \text{VaR}_u[C]du,$$

where $\text{VaR}_q[C] := \inf\{\eta \in \mathbb{R} | \mathbb{P}(C \le \eta) \ge 1 - q\}$, the worst q-quantile of the cost distribution.

Our goal is to find the optimal policy $\pi^\star \in \Pi^{\Theta}$ that minimizes the CVaR value of the total cost at risk level $q \in (0, 1]$. More formally, we aim to solve

$$\min_{\pi \in \Pi^{\Theta}} \left\{ J_q(\pi) := q \cdot \text{CVaR}_q^\pi \left[\sum_{t=1}^T C_t \right] \right\}, \qquad (*)$$

where the objective function $J_q : \Pi^{\mathcal{H}} \to \mathbb{R}$ is a scaled version of CVaR objective, greatly simplifying some expressions in the later steps. We will also often use $C_{s:t} := \sum_{i=s}^t C_i$ as an abbreviation.

Unlike the risk-neutral setting, the CVaR-optimal non-anticipating policy may not be a Markov policy, i.e., $\min_{\pi \in \Pi^{\mathcal{H}}} J_q(\pi) \le \min_{\pi \in \Pi^{\mathcal{X}}} J_q(\pi)$. Finding the optimal non-anticipating policy is beyond the scope of this paper as we consider policy optimization over a set of Markov policies, $\Pi^{\Theta} \subseteq \Pi^{\mathcal{X}}$.

Later in Section 3.4, we will consider an extension to spectral risk measure (Acerbi, 2002), defined as

$$J_\varphi(\pi) := \int_{q=0}^1 \varphi(q) \cdot \text{VaR}_q^\pi[C_{1:T}] \cdot dq, \qquad (1)$$

for some non-increasing function $\varphi : [0, 1] \to \mathbb{R}_{\ge 0}$. The original CVaR objective $J_q(\pi)$ corresponds to the choice $\varphi(\cdot) = \mathbb{I}\{\cdot < q\}$.

Risk-neutral policy gradient framework We aim to update the policy parameter θ through a stochastic gradient descent procedure. We here sketch the procedure for the risk-neutral policy optimization tasks ($q = 1$). At each iteration, it runs a policy π^θ, obtains a sample trajectory $H_{T+1} = (X_1, A_1, C_1, \dots, X_T, A_T, C_T)$, and updates the policy parameters θ according to

$$\theta \leftarrow \theta - \alpha^\theta \cdot \hat{\nabla}_\theta J(H_{T+1}),$$

where $\hat{\nabla}_\theta J(H_{T+1})$ is some noisy gradient estimate, and $\alpha^\theta \in \mathbb{R}$ is the step size. After an update, a projection or clipping can be applied to ensure that the updated parameter lies within Θ. The gradient estimator $\hat{\nabla}_\theta J(\cdot)$ is desired to be unbiased, i.e., $\mathbb{E}^\pi\left[\hat{\nabla}_\theta J(H_{T+1})\right] = \frac{d}{d\theta} J(\pi^\theta)$.

We here provide some gradient estimators commonly adopted in the risk-neutral policy optimization tasks:

- *Score function trick*: In the task of optimizing a randomized policy ($A_t \sim \pi_t^\theta(\cdot|X_t)$),

$$\hat{\nabla}_\theta J_{q=1}(H_{T+1}) = \sum_{t=1}^T \frac{\partial \log \pi_t^\theta(A_t|X_t)}{\partial \theta} \cdot \sum_{s=t}^T C_s. \qquad (2)$$

- *Direct differentiation*: In the task of optimizing a deterministic policy ($A_t = \pi_t^\theta(X_t)$) in a differentiable environment where the entire sample path is differentiable[2] with respect to θ,

$$\hat{\nabla}_\theta J_{q=1}(H_{T+1}) = \sum_{t=1}^T \frac{\partial \pi_t^\theta(X_t)}{\partial \theta} \cdot \sum_{s=t}^T \frac{dC_s}{dA_t}. \qquad (3)$$

- *Randomized finite difference methods*:

$$\hat{\nabla}_\theta J_{q=1}(H_{T+1}, \tilde{H}_{T+1}) = \frac{\epsilon}{\sigma} \cdot \sum_{t=1}^T (\tilde{C}_t - C_t), \qquad (4)$$

where $\tilde{H}_{T+1}$ is the sample trajectory obtained with perturbed parameter $\tilde{\theta} = \theta + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2 I_d)$.

[1] We would imagine that the decision maker is well aware of what kind of policies will be effective in her own task, and presumably the main factors determining their decisions are well reflected in the state variable so that considering Markov policies is sufficient.

[2] This assumes that the state/action spaces are continuous and the state transition is also differentiable with respect to the state/action variables. Pair trading application in Section 5 is an example of such environment.

3. Algorithm

In this section, we develop an effective procedure to solve (∗), which we call *Predictive CVaR Policy Gradient* (PCVAR).

3.1. Risk-Neutral Reformulation

Following from the variational representation of CVaR measure (Rockafellar et al., 2000), the objective $J_q(\pi)$ in (∗) can be rewritten as

$$J_q(\pi) = \min_{\eta\in\mathbb{R}} \mathbb{E}^\pi \left[q\eta + (C_{1:T} - \eta)^+\right], \tag{5}$$

where $(x)^+ := \max(0, x)$ and $C_{1:T} := \sum_{t=1}^T C_t$. Based on this representation, we can view the CVaR policy optimization problem (∗) as a joint minimization of a risk-neutral objective, i.e., $\min_{\theta\in\Theta,\eta\in\mathbb{R}} \mathbb{E}^{\pi^\theta}\left[q\eta + (C_{1:T} - \eta)^+\right]$.

However, a direct application of risk-neutral policy gradient algorithms could be problematic. The term $(C_{1:T} - \eta)^+$ is not time-separable, which prevents the gradient estimators from relying on partial sum of cost realizations (e.g., (2) and (3) cannot be immediately applied). Also, θ will be updated only when the total cost exceeds η, resulting in a low sample efficiency.

3.2. Predictive Tail Probability Process

Given a policy $\pi \in \Pi^{\mathcal{H}}$ and a threshold $\eta \in \mathbb{R}$, we define a *predictive tail probability process* $Q^{\pi,\eta} = (Q_t^{\pi,\eta})_{t=1,\dots,T}$ as

$$Q_t^{\pi,\eta} := \mathbb{P}^\pi \left(C_{1:T} \geq \eta \,|\, H_{t+1}\right), \tag{6}$$

the likelihood that the current sample path ends up with the total cost exceeding the threshold η.

Rao-Blackwellized time decomposition As running estimates, the process $Q^{\pi,\eta}$ is a Doob martingale satisfying

$$Q_T^{\pi,\eta} = \mathbb{I}\{C_{1:T} \geq \eta\}, \quad Q_t^{\pi,\eta} = \mathbb{E}^\pi\left[Q_T^{\pi,\eta} | H_{t+1}\right],$$
$$Q_0^{\pi,\eta} = \mathbb{P}^\pi(C_{1:T} \geq \eta).$$

Next lemma offers some form of time decomposition utilizing these properties.

Lemma 3.1. *Given a non-anticipating policy $\pi \in \Pi^{\mathcal{H}}$ and a threshold $\eta \in \mathbb{R}$, we have*

$$\mathbb{E}^\pi\left[(C_{1:T} - \eta)^+\right] = \mathbb{E}^\pi\left[\sum_{t=1}^T Q_t^{\pi,\eta} C_t\right] - \eta Q_0^{\pi,\eta}.$$

Proof. By definition, $(C_{1:T} - \eta)^+ = (C_{1:T} - \eta)\cdot Q_T^{\pi,\eta}$. For each $t = 1, \dots, T$, by Tower rule, we have

$$\begin{aligned}\mathbb{E}^\pi\left[Q_T^{\pi,\eta} C_t\right] &= \mathbb{E}^\pi\left[\mathbb{E}\left(Q_T^{\pi,\eta} C_t | H_{t+1}\right)\right] \\ &= \mathbb{E}^\pi\left[\mathbb{E}\left(Q_T^{\pi,\eta} | H_{t+1}\right)\cdot C_t\right] \\ &= \mathbb{E}^\pi\left[Q_t^{\pi,\eta} C_t\right].\end{aligned}$$

Combining this with the fact that $Q_0^{\pi,\eta} = \mathbb{E}^\pi[Q_T^{\pi,\eta}]$ gives the desired result. □

Consequently, we present a risk-neutral and time-separable reformulation of CVaR objective.

Proposition 3.2. *Given a non-anticipating policy $\pi \in \Pi^{\mathcal{H}}$, let $\eta^\pi := \mathit{VaR}_q^\pi[C_{1:T}]$, the optimal solution to minimization in (5). If $C_{1:T}$ has no probability mass at η^π,*

$$J_q(\pi) = \mathbb{E}^\pi\left[\sum_{t=1}^T Q_t^{\pi,\eta^\pi} C_t\right]. \tag{7}$$

If $C_{1:T}$ has a probability mass at η^π,

$$\left|J_q(\pi) - \mathbb{E}\left[\sum_{t=1}^T Q_t^{\pi,\eta^\pi} C_t\right]\right| \leq |\eta^\pi| \cdot \mathbb{P}\left(C_{1:T} = \eta^\pi\right).$$

Proof. By Lemma 3.1, we have

$$J_q(\pi) = \eta^\pi \cdot (q - Q_0^{\pi,\eta^\pi}) + \mathbb{E}^\pi\left[\sum_{t=1}^T Q_t^{\pi,\eta^\pi} C_t\right].$$

We further deduce that $|q - Q_0^{\pi,\eta^\pi}| \leq \mathbb{P}^\pi\left(C_{1:T} = \eta^\pi\right)$ by utilizing the following fact (Rockafellar & Uryasev, 2002, Proposition 5):

$$-\mathbb{P}^\pi(C_{1:T} = \eta^\pi) \leq q - \mathbb{P}^\pi(C_{1:T} \geq \eta^\pi) \leq 0.$$

□

Predictive tail probability estimation In general, the predictive tail probability $Q_t^{\pi,\eta}$ should be considered as a function of history H_{t+1}. If we restrict our attention to Markov policies $\Pi^{\mathcal{X}}$, the process $Q^{\pi,\eta}$ exhibits a very limited dependence on the history, which is useful to build a model to approximate it.

Proposition 3.3. *Fix a Markov policy $\pi \in \Pi^{\mathcal{X}}$. There exists a sequence of functions $(f_t^\pi)_{t=1,\dots,T}$ with $f_t^\pi : \mathcal{X}\times\mathbb{R} \to [0,1]$ satisfying*

$$Q_t^{\pi,\eta} = f_t^\pi(X_{t+1}, C_{1:t} - \eta),$$

almost surely. These functions are invariant in η.

Proof. The claim immediately follows from that

$$\begin{aligned}Q_t^{\pi,\eta} &= \mathbb{P}^\pi\left(C_{1:T} > \eta | H_{t+1}\right) \\ &= \mathbb{P}^\pi\left(C_{t+1:T} > \eta - C_{1:t} | H_{t+1}\right) \\ &\overset{(a)}{=} \mathbb{P}^\pi\left(C_{t+1:T} > \eta - C_{1:t} | H_{t+1}, C_{1:t} - \eta\right) \\ &\overset{(b)}{=} \mathbb{P}^\pi\left(C_{t+1:T} > \eta - C_{1:t} | X_{t+1}, C_{1:t} - \eta\right),\end{aligned}$$

where step (a) uses the fact that history H_{t+1} includes all cost realizations up to time t, $(C_1, \dots, C_t)$, and step (b) uses the Markov property. □

Exploiting Proposition 3.3, we will consider a series of functions $(f_t^\phi)_{t=1,\dots,T}$, parameterized by $\phi \in \Phi$, as a model to approximate the predictive tail probabilities, i.e.,

$$Q_t^{\pi,\eta} \approx f_t^\phi(X_{t+1}, C_{1:t} - \eta).$$

For example, one can consider a logistic model such as $f_t^\phi(x,c) = \left(1+\exp(\phi_t^\top(x,c))\right)^{-1}$ or a parametric distribution model such as $f_t^\phi(x,c) = \mathbb{P}(\mathcal{N}(\mu_t^\phi(x), \sigma_t^\phi(x)) \le c)$. Instead of investigating a particular functional form, we mildly assume that f_t^ϕ's are differentiable with respect to ϕ. Finding ϕ can be thought as a typical supervised learning task, which is to predict whether the total cost exceeds the threshold η using information X_{t+1} and $C_{1:t} - \eta$. See also Equation (11).

3.3. Predictive CVaR Policy Gradient Algorithm

Based on the findings presented in Section 3.2, we decompose the CVaR policy optimization ($*$) into *three* optimization problems – an outer optimization for the policy parameter θ, and two inner optimizations for the threshold η and the predictive tail probability model parameter ϕ. Namely, we solve

$$\min_{\theta\in\Theta} \left\{ J(\theta,\eta,\phi) \,\middle|\, \begin{array}{l} \eta \in \arg\min_{\eta'\in\mathbb{R}} L(\theta,\eta'), \\ \phi \in \arg\min_{\phi'\in\Phi} M(\theta,\eta,\phi') \end{array} \right\}, \quad (8)$$

where

$$J(\theta,\eta,\phi) := \mathbb{E}\left[\sum_{t=1}^T \hat{Q}_t C_t\right], \quad (9)$$

$$L(\theta,\eta) := \mathbb{E}\left[q\eta + (C_{1:T} - \eta)^+\right], \quad (10)$$

$$M(\theta,\eta,\phi) := \mathbb{E}\left[\sum_{t=1}^T \left(\mathbb{I}\{C_{1:T} \ge \eta\} - \hat{Q}_t\right)^2\right], \quad (11)$$

and $(\hat{Q}_t)_{t=1,\dots,T}$ is the approximate predictive tail probability process generated according to

$$\hat{Q}_t = f_t^\phi(X_{t+1}, C_{1:t} - \eta).$$

Each optimization problem is justified as follows.

1. *Policy optimization (θ optimization)*, which aims to optimize the policy parameter θ via the objective $J(\theta,\eta,\phi) = \mathbb{E}[\sum_{t=1}^T \hat{Q}_t C_t]$. Proposition 3.2 justifies this objective.

2. *VaR$_q$ value estimation (η optimization)*, which aims to estimate the VaR$_q$ value of the total cost distribution under the policy π^θ, via the objective $L(\theta,\eta) = \mathbb{E}[q\eta + (C_{1:T} - \eta)^+]$. The variational representation of CVaR measure (5) justifies this objective.

3. *Predictive tail probability estimation (ϕ optimization)*, which aims to learn a model f^ϕ to approximate the process $Q^{\pi^\theta,\eta}$, via the objective $M(\theta,\eta,\phi)$. The objective Proposition 3.3 justifies this objective.

The optimization problem (8) is a leader-follower type of optimization problem in which θ should be optimized in a consideration of η and ϕ values being re-optimized in response to the changes in θ value. To apply the gradient descent procedure, we first substitute the inner optimizations with their first order optimality conditions, i.e.,

$$\min_{\theta\in\Theta} \left\{ J(\theta,\eta,\phi) \,\middle|\, \frac{\partial}{\partial\eta} L(\theta,\eta) = 0, \frac{\partial}{\partial\phi} M(\theta,\eta,\phi) = 0 \right\},$$

and then apply the Lagrangian relaxation so that we optimize θ using the following objective:

$$J(\theta,\eta,\phi) + \lambda_L \frac{\partial}{\partial\eta} L(\theta,\eta) + \lambda_M^\top \frac{\partial}{\partial\phi} M(\theta,\eta,\phi), \quad (12)$$

where $\lambda_L \in \mathbb{R}$ and $\lambda_M \in \mathbb{R}^{\dim(\phi)}$ are Lagrangian multipliers.

Our suggested algorithm, PCVaR, updates the three variables, θ, η, and ϕ, in parallel via stochastic gradient descent procedure with the objectives (12), (10), and (11), respectively. Algorithm 1 below sketches the overall workflow.

Algorithm 1 Predictive CVaR Policy Gradient

1: Initialize θ, η, ϕ.
2: **for** *episode* $k = 1, 2, 3, \dots$ **do**
3: Run π^θ and obtain a sample trajectory $H_{T+1} = (X_1, A_1, C_1, \dots, X_T, A_T, C_T)$.
4: Compute approximate predictive tail probabilities:

$$\hat{Q}_t \leftarrow f_t^\phi(X_{t+1}, C_{1:t} - \eta), \quad \forall t = 1, \dots, T.$$

5: Re-weight cost realizations with predictive tail probabilities:

$$H_{T+1}^Q \leftarrow (X_1, A_1, \hat{Q}_1 C_1, \dots, X_T, A_T, \hat{Q}_T C_T).$$

6: Compute $\hat{\nabla}_\theta J(H_{T+1}^Q)$ using H_{T+1}^Q as in the risk-neutral policy gradient (e.g., (16)–(18))
7: Update θ, η, and ϕ using (15), (13), and (14), respectively.
8: **end for**

Initialization To obtain reasonable initial parameter values, the result of risk-neutral policy optimization can be utilized (but not restricted to). After running a conventional policy gradient algorithm with the objective $\mathbb{E}[C_{1:T}]$, the resulting policy parameter can be used as an initial θ value, and then by minimizing (10) and (11) with sample trajectories collected during this procedure, η and ϕ values can be properly initialized.

η update Applying the stochastic gradient descent (SGD) with respect to the objective $L(\theta, \eta) := \mathbb{E}\left[q\eta + (C_{1:T} - \eta)^+\right]$, PCVAR updates η according to

$$\eta \leftarrow \eta - \alpha^\eta \cdot \hat{\nabla}_\eta L, \quad \hat{\nabla}_\eta L := q - \mathbb{I}\{C_{1:T} \geq \eta\}, \quad (13)$$

where α^η is the step size that may change over iterations. Roughly speaking, the threshold η gets updated to satisfy $\mathbb{E}[\hat{\nabla}_\eta L] = 0$, stochastically converging to $\text{VaR}_q^{\pi^\theta}[C_{1:T}]$. See Proposition 4.2 for the formal convergence analysis.

ϕ update Applying the SGD with respect to the objective $M(\theta, \eta, \phi)$ defined in (11), the algorithm updates ϕ according to

$$\phi \leftarrow \phi - \alpha^\phi \cdot \hat{\nabla}_\phi M, \quad (14)$$

$$\hat{\nabla}_\phi M := \frac{\partial}{\partial \phi} \sum_{t=1}^{T} \left(\mathbb{I}\{C_{1:T} \geq \eta\} - f_t^\phi(X_{t+1}, C_{1:t} - \eta)\right)^2,$$

where α^ϕ is the step size. A stylized convergence analysis is provided in Proposition 4.1.

θ update One salient feature of PCVAR is that conventional risk-neutral policy gradient algorithms can be used seamlessly to update θ. Namely, applying the SGD with respect to the objective (12), it updates θ according to

$$\theta \leftarrow \theta - \alpha^\theta \cdot \left(\hat{\nabla}_\theta J + \lambda_L \hat{\nabla}^2_{\theta\eta} L + \lambda_M \hat{\nabla}^2_{\theta\phi} M\right), \quad (15)$$

where $\hat{\nabla}_\theta J$, $\hat{\nabla}^2_{\theta\eta} L$, and $\hat{\nabla}^2_{\theta\phi} M$ are noisy estimates of $\frac{\partial}{\partial\theta} J(\theta, \eta, \phi)$, $\frac{\partial^2}{\partial\theta\partial\eta} L(\theta, \eta)$, and $\frac{\partial^2}{\partial\theta\partial\phi} M(\theta, \eta, \phi)$, respectively.

The estimate $\hat{\nabla}_\theta J$ can be computed using usual gradient estimators adopted for risk-neutral policy optimization tasks, simply by replacing the actual cost realizations $(C_1, \ldots, C_T)$ with their re-weighted version $(\hat{Q}_1 C_1, \ldots, \hat{Q}_T C_T)$. Specifically, one can adopt the estimate based on score function trick (cf. (2)),

$$\hat{\nabla}_\theta J(H^Q_{T+1}) = \sum_{t=1}^{T} \frac{\partial \log \pi_t^\theta(A_t|X_t)}{\partial\theta} \cdot \sum_{s=t}^{T} \hat{Q}_s C_s, \quad (16)$$

the one based on direct differentiation (cf. (3)),

$$\hat{\nabla}_\theta J(H^Q_{T+1}) = \sum_{t=1}^{T} \frac{\partial \pi_t^\theta(X_t)}{\partial\theta} \cdot \sum_{s=t}^{T} \frac{d(\hat{Q}_s C_s)}{dA_t}, \quad (17)$$

or the one based on randomized FDM (cf. (4)),

$$\hat{\nabla}_\theta J(H^Q_{T+1}, \tilde{H}^Q_{T+1}) = \frac{\epsilon}{\sigma} \cdot \sum_{t=1}^{T} (\tilde{\hat{Q}}_t \tilde{C}_t - \hat{Q}_t C_t). \quad (18)$$

The other estimates $\hat{\nabla}^2_{\theta\eta} L$ and $\hat{\nabla}^2_{\theta\phi} M$ can also be computed similarly. As a concrete example, score function trick suggests $\hat{\nabla}^2_{\theta\eta} L = S \times \hat{\nabla}_\eta L$, and $\hat{\nabla}^2_{\theta\phi} M = S \times \hat{\nabla}_\phi M$, with $S := \sum_{t=1}^{T} \partial \log \pi_t^\theta(A_t|X_t)/\partial\theta$.

The Lagrangian multipliers, λ_L and λ_M, can be fixed to proper constants or optimized interatively as in the saddle point optimization.

Sample efficiency A direct application of CVaR policy gradient to the objective (5) corresponds to the case where $\hat{Q}_1 = \ldots = \hat{Q}_T = \mathbb{I}\{C_{1:T} \geq \eta\}$. It discards the sample trajectories with total cost below η, so that it utilizes only a q-fraction of the samples. In contrast, PCVAR utilizes every single sample trajectory by reweighting the cost terms, proportionally to their predicted contribution to the objective. This procedure enhances the sample efficiency as it aggregates gradient feedback across a larger number of samples and removes unnecessary noises in the policy evaluation. See also Lemma 4.4 and Proposition 4.5 for some formal analysis.

3.4. Extension to Spectral Risk Measure

Linear combination of CVaR objectives As a simpler extension, let us consider a situation where the objective is given by a linear combination of CVaR objectives,

$$J_w(\pi) := \sum_{i=1}^{n} w_i \cdot J_{q_i}(\pi),$$

for given risk levels $(q_1, \ldots, q_n) \in [0, 1]^n$ and their weights $(w_1, \ldots, w_n) \in \mathbb{R}^n_+$. For each risk level q_i, the corresponding VaR_{q_i} value can be defined as $\eta_i := \min_{\eta \in \mathbb{R}} \mathbb{E}^\pi [q_i \cdot \eta + (C_{1:T} - \eta)^+]$, and its update can be done according to (13) without any modification.

The predictive tail probabilities can also be defined and approximated analogously to the simple CVaR case, i.e.,

$$\begin{aligned} Q_t^{\pi,\eta} &:= \mathbb{E}^\pi \left[\sum_i w_i \mathbb{I}\{C_{1:T} \geq \eta_i\} \,\middle|\, H_{t+1} \right] \\ &= \sum_i w_i \mathbb{P}^\pi \left(C_{1:T} \geq \eta_i \middle| H_{t+1} \right) \\ &\approx \sum_i w_i f_t^\phi(X_{t+1}, C_{1:t} - \eta_i). \end{aligned}$$

As implied in Proposition 3.3, the function f_t^ϕ does not need to depend on η_i so that we do not need to build a separate model for each q_i. All procedures implemented Algorithm 1 remain effective except that $\hat{Q}_t \leftarrow \sum_i w_i f_t^\phi(X_{t+1}, C_{1:t} - \eta_i)$ will be used instead.

Spectral risk measure Now we consider a situation where the objective is given by a spectral risk measure.

Lemma 3.4. *Consider a spectral risk measure (1) induced by a non-increasing function $\varphi : [0,1] \to \mathbb{R}_{\geq 0}$. If $C_{1:T}$ is bounded,*

$$J_\varphi(\pi) = \varphi(1) J_{q=1}(\pi) - \int_{q=0}^{1} J_q(\pi) d\varphi(q).$$

Proof. Fix π and let $h(q) := J_q(\pi)$. Observe that

$$\frac{dh(q)}{dq} = \frac{d}{dq}\left(\int_0^q \text{VaR}_u^\pi[C_{1:T}]du\right) = \text{VaR}_q^\pi[C_{1:T}].$$

Therefore,

$$\begin{aligned} J_\varphi(\pi) &:= \int_0^1 \varphi(q)\text{VaR}_q^\pi[C_{1:T}]dq \\ &= \int_0^1 \varphi(q) dh(q) \\ &\overset{(a)}{=} \varphi(1)h(1) - \varphi(0)h(0) - \int_0^1 h(q) d\varphi(q) \\ &\overset{(b)}{=} \varphi(1)h(1) - \int_0^1 h(q) d\varphi(q), \end{aligned}$$

where step (a) uses integration by parts, and step (b) uses the fact that $h(0) = J_0(\pi) = 0$ if $C_{1:T}$ is bounded. □

Lemma 3.4 shows that this objective can be understood as a linear combination of CVaR objectives. More specifically, with $q_i := i/n$ for $i = 0, \ldots, n$, one can approximate

$$J_\varphi(\pi) \approx \sum_{i=0}^{n} w_i \cdot J_{q_i}(\pi),$$

where $w_i := \varphi(q_i) - \varphi(q_{i+1})$ for $i = 0, \ldots, n-1$, and $w_n := \varphi(q_{n-1})$. This reformulation allows us to apply PCVaR algorithm.

4. Theoretical Analysis

In this section, we provide a few theoretical results showing the correctness and effectiveness of PCVaR algorithm. All proofs are provided in Appendix A.

We begin by showing the *consistency* of the estimators related to parameters ϕ and η.

Proposition 4.1. *Suppose $|\mathcal{X}| < \infty, |\mathcal{A}| < \infty$, and random costs C_t's are supported on a finite set so that $C_{t:T}$ also takes values in a finite set $\mathcal{Y} \subset \mathbb{R}$ with $|\mathcal{Y}| < \infty$. Consider a tabular parameterization of predictive tail probability model, i.e., $f_t^\phi(x,y) = \phi_{t,x,y}$ with $\phi \in \mathbb{R}^{T\times|\mathcal{X}|\times|\mathcal{Y}|}$.*

Given a Markov policy $\pi \in \Pi^{\mathcal{X}}$ and a sequence of η values $(\eta^{(k)})_{k\in\mathbb{N}} \in \mathcal{Y}^\infty$, suppose that the update rule (14) is adopted:

$$\phi^{(k+1)} \leftarrow \phi^{(k)} - \alpha_k^\phi \cdot$$

$$\frac{\partial}{\partial \phi} \sum_{t=1}^{T} \Big(\mathbb{I}\{C_{1:T}^{(k)} \geq \eta^{(k)}\} - \overbrace{f_t^\phi(X_{t+1}^{(k)}, C_{1:t}^{(k)} - \eta^{(k)})}^{=:\hat{Q}_t^{(k)}}\Big)^2,$$

where $H_{t+1}^{(k)}$ is the k-th sample trajectory obtained with π. If $\eta^{(k)} \to \eta^\star$ almost surely and the step size sequence satisfies $\sum_k \alpha_k^\phi = \infty$ and $\sum_k (\alpha_k^\phi)^2 < \infty$, we have

$$\max_t |\hat{Q}_t^{(k)} - Q_t^{\pi,\eta^\star}| \to 0,$$

almost surely as $k \to \infty$.

Proposition 4.2. *Given a non-anticipating policy $\pi \in \Pi^{\mathcal{H}}$, suppose that the total cost distribution has no probability mass at $\eta^\pi := \text{VaR}_q^\pi[C_{1:T}]$. With the policy π fixed, if the update rule (13) is adopted with a step size sequence satisfying $\sum_k \alpha_k^\eta = \infty$ and $\sum_k (\alpha_k^\eta)^2 < \infty$, we have*

$$\eta^{(k)} \to \eta^\pi,$$

almost surely as $k \to \infty$, where $\eta^{(k)}$ is the value of η parameter at the k^{th} iteration.

Ignoring a subtle inconsistency in their technical conditions,[3] above propositions show that ϕ and η will jointly converge to their target values (i.e., the optimal solutions to (11) and (10)) for fixed θ. As long as the policy parameter θ changes slowly over the iterations, the estimated process $\hat{Q}$ will keep serving as a good approximation of the ideal predictive tail probability process Q^{π^θ,η^π}. Exploiting this implication, we investigate the gradient estimators in the policy optimization procedure assuming $\hat{Q} = Q^{\pi^\theta,\eta^\pi}$.

We next show the *unbiasedness* of the gradient estimators of reformulated objectives.

Theorem 4.3. *Given a policy $\pi^\theta \in \Pi^\Theta$, a threshold $\eta \in \mathbb{R}$, and a prediction model parameter $\phi \in \Phi$, define*

$$\hat{\nabla}_\theta J_t^{(1)} := \frac{\partial \log \pi_t^\theta(A_t|X_t)}{\partial \theta} \cdot \sum_{s=1}^{T} \hat{Q}_s C_s,$$

$$\hat{\nabla}_\theta J_t^{(2)} := \frac{\partial \log \pi_t^\theta(A_t|X_t)}{\partial \theta} \cdot \sum_{s=t}^{T} \hat{Q}_s C_s,$$

where $(\hat{Q}_t)_{t=1,\ldots,T}$ is the approximate predictive tail probability process generated according to

$$\hat{Q}_t = f_t^\phi(X_{t+1}, C_{1:t} - \eta).$$

[3] Proposition 4.1 assumes that the total cost is a discrete random variable, which is inconsistent to Proposition 4.2's assumption that the total cost has no probability mass at η^π.

Then, for all $i \in \{1, 2\}$,

$$\frac{\partial}{\partial\theta} J(\theta, \eta, \phi) = \mathbb{E}\left[\sum_{t=1}^{T} \hat{\nabla}_\theta J_t^{(i)}\right].$$

Note that $\hat{\nabla}_\theta J_t^{(2)}$ corresponds to the score-based gradient estimator suggested in (16). We remark that this result is not so trivial. For example, $\hat{\nabla}_\theta J_t^{(2)}$ could also fail if the history H_t did not include X_t. While not formally stated here, the unbiasedness of the gradient estimator (17) with $\hat{Q} = Q^{\pi^\theta, \eta^\pi}$ immediately follows from the chain rule.

We next show that the reweighting with predictive tail probabilities is helpful to achieve the *variance reduction* in the gradient estimation.

Lemma 4.4. *Given* $\pi \in \Pi^{\mathcal{H}}$ *and* $\eta \in \mathbb{R}$, *for any* t

$$\mathit{Var}(Q_t^{\pi,\eta} C_t) \le \mathit{Var}(Q_T^{\pi,\eta} C_t).$$

As an immediate consequence of Rao-Blackwellization, above lemma shows that our reformulated objective and our suggested gradient estimators involve the cost terms that are less noisy (e.g., $q\eta + (C_{1:T} - \eta)^+$ vs. $\sum_{t=1}^T Q_t C_t$). Although variance reductions in the gradient estimation are not theoretically guaranteed due to a possibly complicated temporal-dependency among these cost terms, the actual reductions are observed in numerical experiments.

Proposition 4.5. *Consider the setting of Theorem 4.3. If the random costs are bounded and non-negative, for any* t,

$$\mathbb{V}\left[\hat{\nabla}_\theta J_t^{(2)}\right] \le \mathbb{V}\left[\hat{\nabla}_\theta J_t^{(1)}\right],$$

where $\mathbb{V}[X] := \mathit{trace}(\mathit{Cov}[X])$ *for a random vector* X.

Particularly for the score-based gradient estimators, above proposition shows that an additional variance reduction can be made by time decomposition. In comparison with $\hat{\nabla}_\theta J_t^{(1)}$, the estimator $\hat{\nabla}_\theta J_t^{(2)}$ involves a fewer number of terms.

5. Numerical Experiments

We conduct two numerical experiments to evaluate our suggested algorithm (PCVaR) in a comparison with the other two competing algorithms – GCVaR (Tamar et al., 2015b), and a naïve version of PCVaR (NCVaR) that does not employ the predictive tail probabilities. Details about the NCVaR are described in Appendix B.1.

5.1. Continuous Blackjack Game

Setting We consider a continuous version of Blackjack game. At each time step, the agent decides whether to continue or stop receiving a random number uniformly distributed on $[0, 4]$. The agent earns a reward equal to the drawn random number, but if their cumulative sum exceeds 21, the game ends and the agent receives a large penalty instead. Formally, $\mathcal{X} := \mathbb{R}_+$, $\mathcal{A} := \{\text{cont}, \text{stop}\}$, $X_{t+1} = X_t + U_t + 21 \cdot \mathbb{I}\{A_t = \text{stop}\}$ where $U_t \sim \text{Unif}[0, 4]$, and $R_t = U_t \cdot \mathbb{I}\{A_t = \text{cont}, X_{t+1} < 21\} + Z \cdot \mathbb{I}\{A_t = \text{cont}, X_{t+1} \ge 21\}$ where $Z \sim \mathcal{N}(-30, 1)$.

Table 1. Variance of gradient estimates used by the three algorithms in order to update the policy parameter θ, in the continuous Blackjack experiment. The gradients are evaluated at $\theta = 16.83$ (the risk-neutral solution; the first row), and at $\theta = 14.2$ (the CVaR-optimal solution; the second row).

Evaluation point	PCVaR	NCVaR	GCVaR
$\theta = \arg\max_\theta \mathbb{E}[R_{1:T}]$	33.66	97.59	96.92
$\theta = \arg\max_\theta \text{CVaR}_q[R_{1:T}]$	6.71	24.01	24.67

We concern the target risk level $q = 0.1$, and consider a randomized policy that involves a soft threshold $\theta \in [0, 21]$ such that $\pi^\theta(A_t = 1 | X_t) = \sigma(0.5(\theta - X_t))$ where $\sigma(z) := 1/(1+e^{-z})$ is a sigmoid function. Given $q = 0.1$, the CVaR value is optimized at $\theta \approx 14.2$.

We initialize the policy parameter θ to be either $\theta = 16.38$ (a risk-neutral optimal solution) or $\theta = 13.5$, and run the three algorithms with a learning rate $\alpha^\theta = 0.005$ and a batch size $B = 16$. To apply our suggested PCVaR algorithm, we introduce a prediction model with 12-dim parameters, $\phi = (\phi^1, \phi^2) \in \mathbb{R}^6 \times \mathbb{R}^6$, such that $f^\phi(x, c) = \mathbb{I}\{c < 0\} \cdot B_5^{\phi^1}(x/21) + \mathbb{I}\{c \ge 0\} \cdot B_5^{\phi^2}(x/21)$, where $B_5^\phi(\cdot)$ is a Bernstein polynomial of degree 5 with coefficients ϕ, and use constant Lagrangian multipliers, $\lambda_L = \lambda_M = 0.3$. Further details are given in Appendix B.2.

Results Figure 2 shows the trajectories of θ value, highlighting that PCVaR learns the CVaR-optimal solution correctly and faster than the baselines. Note that the same learning rate and the batch size are used across the three algorithms. Table 1 shows the variance of gradient estimates adopted by the three algorithms, evaluated at either $\theta = 16.83$ or $\theta = 14.2$. These results demonstrate that the enhanced sample efficiency is attributable to the reduction in the variation estimation.

5.2. Pair Trading

Setting We evaluate PCVaR algorithm using a real-world dataset. Following Han et al. (2023), we consider the intraday pair trading of two stocks and the use of Tiingo dataset[4],

[4]Obtained through Tiingo End-Of-Day API: https://api.tiingo.com/documentation/iex

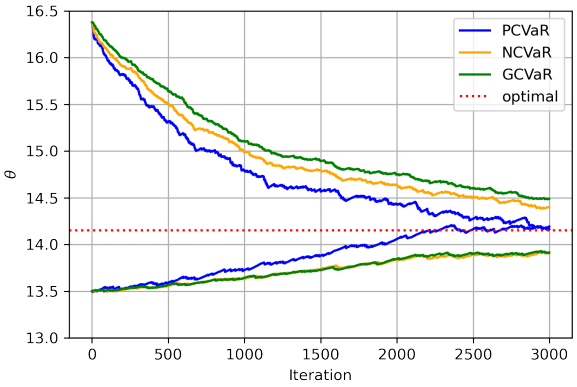

Figure 2. Trajectories of the policy parameter θ over the course of policy gradient procedure in the continuous Blackjack experiment. The dashed horizontal line indicates its CVaR-optimal solution, and its initial value is set to be either $\theta = 16.38$ (above) or $\theta = 13.5$ (below).

which consists of 990 days of observations (from Jan 2, 2015 to Dec 18, 2018).

We consider an intraday trading strategy that makes decisions every ten minutes ($T = 39$). In each period t, it determines the trading position $A_t \in [-1, 1]$ and earns a reward R_t that is given by

$$R_t = A_t \cdot r_{t+1} - 0.05 \cdot |A_t - A_{t-1}|,$$

where r_t indicates the difference in 10-min return of two stocks measured in percentage. The first term captures the trading profit/loss and the second term captures the transaction cost. We consider a certain family of trading strategies that determine the trading position according to $A_t = \pi_t^\theta(X_t) = \tanh\left(\theta_1^\top X_t\right) \cdot \sigma\left(\theta_2^\top X_t\right)$, where the state variable is given as $X_t := (1, r_{t-5}, \ldots, r_t, \sum_{s=1}^t R_s, A_{t-1})$. Our objective is $J(\pi) := \mathbb{E}\left[\sum_{t=1}^T R_t\right] + 0.1 \cdot \text{CVaR}_q\left[\sum_{t=1}^T R_t\right]$ with $q = 0.2$. As a linear combination of CVaR objectives, the decomposition established in Section 3.4 is adopted.

A practical deployment of policy optimization techniques is considered: we use the first 330 days of data for the initial training of the trading strategy, and evaluate the strategy during the remaining days, while periodically re-optimizing it every other days using the prior ten days of data. Policy gradient algorithms are used for the initial and periodic trading strategy optimization.

In the application of PCVaR, the predictive tail probabilities are estimated using a model $f_t^\phi(x, c) = 1 - \frac{1}{2}\tanh(\phi_{1,x}^\top x + \phi_{1,c} c) \cdot \sigma(\phi_{2,x}^\top x + \phi_{2,c} c)$, with the choice of $\lambda_L = \lambda_M = 0$. The learning rate and the batch size are tuned for each algorithm separately. Further details are given in Appendix B.3.

Results Figure 3 reports the cumulative return (%) achieved by pair-trading strategy under the maintenance of four different policy gradient algorithms. Our method outperforms all other baseline models. In particular, our method exhibits a stable performance throughout the testing period, highlighting that our PCVaR learns a risk-sensitive policy effectively with a few number of samples.

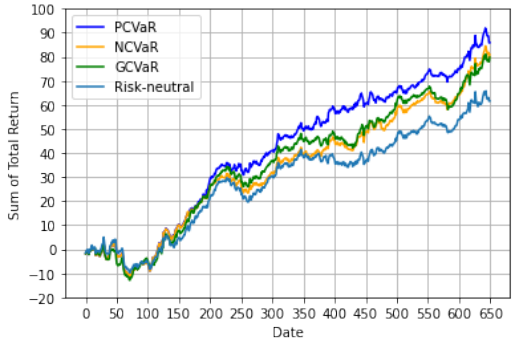

Figure 3. Cumulative return (%) achieved by pair-trading strategy being re-optimized periodically using different policy gradient algorithms.

6. Conclusion and Future Work

We have proposed the predictive CVaR policy gradient that employs the *predictive tail probabilities* to accelerate the risk-sensitive reinforcement learning. While our theoretical analyses and numerical experiments demonstrate the validity and effectiveness of this method, some questions remain unanswered. First, we have restricted our attention to Markov policies, which leads to nice temporal properties of predictive tail probability process. Given that the risk-neutral and time-separable reformulation of CVaR objective, (7), is valid for all non-anticipating policies, it will be worth investigating whether the idea of predictive tail probability can be leveraged for policy optimization over a broader class of policies or even for the value-based methods. Second, our theoretical analysis does not guarantee the convergence rate nor the global optimality. Papini et al. (2018); Xu et al. (2019; 2020) provide a guideline toward the convergence analysis powered by variance reduction techniques, and Bhandari & Russo (2024) provides a guideline toward the global convergence analysis.

Acknowledgements

This research was supported by the National Research Foundation of Korea (NRF-2022R1C1C1013402).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Acerbi, C. Spectral measures of risk: A coherent representation of subjective risk aversion. *Journal of Banking & Finance*, 26(7):1505–1518, 2002.

Bastani, O., Ma, J. Y., Shen, E., and Xu, W. Regret bounds for risk-sensitive reinforcement learning. *Advances in Neural Information Processing Systems*, 35: 36259–36269, 2022.

Bäuerle, N. and Ott, J. Markov decision processes with average-value-at-risk criteria. *Mathematical Methods of Operations Research*, 74:361–379, 2011.

Bhandari, J. and Russo, D. Global optimality guarantees for policy gradient methods. *Operations Research*, 2024.

Bonalli, R., Pavone, M., Chapman, M. P., Smith, K. M., Yang, I., and Tomlin, C. J. Risk-sensitive safety analysis using conditional value-at-risk. *IEEE Transactions on Automatic Control*, 2022.

Borkar, V. and Jain, R. Risk-constrained markov decision processes. *IEEE Transactions on Automatic Control*, 59 (9):2574–2579, 2014.

Borkar, V. S. and Meyn, S. P. Risk-sensitive optimal control for markov decision processes with monotone cost. *Mathematics of Operations Research*, 27(1):192–209, 2002.

Chen, Y., Du, Y., Hu, P., Wang, S., Wu, D., and Huang, L. Provably efficient iterated cvar reinforcement learning with function approximation and human feedback. In *The Twelfth International Conference on Learning Representations*, 2023.

Chow, Y. and Ghavamzadeh, M. Algorithms for cvar optimization in mdps. *Advances in neural information processing systems*, 27, 2014.

Chow, Y., Tamar, A., Mannor, S., and Pavone, M. Risk-sensitive and robust decision-making: a cvar optimization approach. *Advances in neural information processing systems*, 28, 2015.

Chow, Y., Ghavamzadeh, M., Janson, L., and Pavone, M. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18 (167):1–51, 2018.

Du, Y., Wang, S., and Huang, L. Provably efficient risk-sensitive reinforcement learning: Iterated cvar and worst path. *arXiv preprint arXiv:2206.02678*, 2022.

Durrett, R. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

Fei, Y., Yang, Z., Chen, Y., Wang, Z., and Xie, Q. Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret. *Advances in Neural Information Processing Systems*, 33:22384–22395, 2020.

Fei, Y., Yang, Z., Chen, Y., and Wang, Z. Exponential bellman equation and improved regret bounds for risk-sensitive reinforcement learning. *Advances in Neural Information Processing Systems*, 34:20436–20446, 2021.

Greenberg, I., Chow, Y., Ghavamzadeh, M., and Mannor, S. Efficient risk-averse reinforcement learning. *Advances in Neural Information Processing Systems*, 35:32639–32652, 2022.

Han, W., Huang, J., Xie, Q., Zhang, B., Lai, Y., and Peng, M. Mastering pair trading with risk-aware recurrent reinforcement learning. *arXiv preprint arXiv:2304.00364*, 2023.

Haskell, W. B. and Jain, R. A convex analytic approach to risk-aware markov decision processes. *SIAM Journal on Control and Optimization*, 53(3):1569–1598, 2015.

Hiraoka, T., Imagawa, T., Mori, T., Onishi, T., and Tsuruoka, Y. Learning robust options by conditional value at risk optimization. *Advances in Neural Information Processing Systems*, 32, 2019.

Huang, A., Leqi, L., Lipton, Z. C., and Azizzadenesheli, K. On the convergence and optimality of policy gradient for markov coherent risk. *arXiv preprint arXiv:2103.02827*, 2021.

La, P. and Ghavamzadeh, M. Actor-critic algorithms for risk-sensitive mdps. *Advances in neural information processing systems*, 26, 2013.

Li, X., Zhong, H., and Brandeau, M. L. Quantile markov decision processes. *Operations research*, 70(3):1428–1447, 2022.

Liu, Q., Li, L., Tang, Z., and Zhou, D. Breaking the curse of horizon: Infinite-horizon off-policy estimation. *Advances in neural information processing systems*, 31, 2018.

Markowitz, J., Gardner, R. W., Llorens, A., Arora, R., and Wang, I.-J. A risk-sensitive approach to policy optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 15019–15027, 2023.

Miller, C. W. and Yang, I. Optimal control of conditional value-at-risk in continuous time. *SIAM Journal on Control and Optimization*, 55(2):856–884, 2017.

Papini, M., Binaghi, D., Canonaco, G., Pirotta, M., and Restelli, M. Stochastic variance-reduced policy gradient. In *International conference on machine learning*, pp. 4026–4035. PMLR, 2018.

Pflug, G. C. and Pichler, A. Time-consistent decisions and temporal decomposition of coherent risk functionals. *Mathematics of Operations Research*, 41(2):682–699, 2016.

Pore, A., Corsi, D., Marchesini, E., Dall'Alba, D., Casals, A., Farinelli, A., and Fiorini, P. Safe reinforcement learning using formal verification for tissue retraction in autonomous robotic-assisted surgery. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4025–4031. IEEE, 2021.

Prashanth, L. Policy gradients for cvar-constrained mdps. In *International Conference on Algorithmic Learning Theory*, pp. 155–169. Springer, 2014.

Rajeswaran, A., Ghotra, S., Ravindran, B., and Levine, S. Epopt: Learning robust neural network policies using model ensembles. *arXiv preprint arXiv:1610.01283*, 2016.

Rockafellar, R. T. and Uryasev, S. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 26(7):1443–1471, 2002.

Rockafellar, R. T., Uryasev, S., et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

Stanko, S. and Macek, K. Risk-averse distributional reinforcement learning: A cvar optimization approach. In *IJCCI*, pp. 412–423, 2019.

Tamar, A., Di Castro, D., and Mannor, S. Policy gradients with variance related risk criteria. In *Proceedings of the twenty-ninth international conference on machine learning*, pp. 387–396, 2012.

Tamar, A., Chow, Y., Ghavamzadeh, M., and Mannor, S. Policy gradient for coherent risk measures. *Advances in neural information processing systems*, 28, 2015a.

Tamar, A., Glassner, Y., and Mannor, S. Optimizing the cvar via sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015b.

Tamar, A., Chow, Y., Ghavamzadeh, M., and Mannor, S. Sequential decision making with coherent risk. *IEEE transactions on automatic control*, 62(7):3323–3338, 2016.

Vijayan, N. and Prashanth, L. Likelihood ratio-based policy gradient methods for distorted risk measures: A non-asymptotic analysis. *ArXiv, abs/2107.04422*, 2021.

Wang, K., Kallus, N., and Sun, W. Near-minimax-optimal risk-sensitive reinforcement learning with cvar. In *International Conference on Machine Learning*, pp. 35864–35907. PMLR, 2023.

Wen, L., Duan, J., Li, S. E., Xu, S., and Peng, H. Safe reinforcement learning for autonomous vehicles through parallel constrained policy optimization. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–7. IEEE, 2020.

Xie, T., Liu, B., Xu, Y., Ghavamzadeh, M., Chow, Y., Lyu, D., and Yoon, D. A block coordinate ascent algorithm for mean-variance optimization. *Advances in Neural Information Processing Systems*, 31, 2018.

Xu, P., Gao, F., and Gu, Q. Sample efficient policy gradient methods with recursive variance reduction. *arXiv preprint arXiv:1909.08610*, 2019.

Xu, P., Gao, F., and Gu, Q. An improved convergence analysis of stochastic variance-reduced policy gradient. In *Uncertainty in Artificial Intelligence*, pp. 541–551. PMLR, 2020.

A. Proofs

A.1. Proof of Proposition 4.1

Lemma A.1. *(L^p Convergence Theorem (Durrett, 2019)) If $M_n, n \geq 1$, be a martingale with $\sup \mathbb{E}|M_n|^p < \infty$ where $p > 1$, then $M_n \to M$ almost surely and in L^2.*

Lemma A.2. *Suppose the same condition with Proposition 4.1. Let define $\mathcal{X}'_+ := \{x' = (t, x, c) \mid P(X_{t+1} = x, C^{\pi}_{0:t} - \eta = c) > 0\}$ and $Z^{x'}_k := \mathbb{I}\{x' \text{ visit in } H^{(k)}_{T+1}\}$. For all $x' \in \mathcal{X}'_+$, if $\sum_k \alpha_k = \infty$ and $\sum_k \alpha_k^2 < \infty$, then*

$$\sum_k \alpha_k Z^{x'}_k = \infty \quad almost\ surely.$$

Proof. Let be $\mathbb{E}[Z^{x'}_k] = p > 0$ and define $M_n := \sum_{k=1}^n \alpha_k (Z^{x'}_k - p)$. As we consider fixed $\pi \in \Pi^{\mathcal{X}}$, $\mathbb{E}[\alpha_k(Z^{x'}_k - p)] = 0$ for all k. M_n is a martingale.

Note that for all $n \geq 1$,

$$\begin{aligned}
\mathbb{E}[M_n^2] &= \text{Var}(M_n) - \mathbb{E}[M_n]^2 \\
&= \text{Var}(M_n) \\
&\overset{(a)}{=} \sum_k a_k^2 \text{Var}(Z^{x'}_k - p) \\
&= \sum_k a_k^2 p(1-p) < \infty,
\end{aligned}$$

where step (a) uses independence of each trajectory. Then, by Lemma A.1, $M_n \to M_\infty$ almost surely. However, $\sum_{k=1}^n \alpha_k p = \infty$. Thus, $\sum_k \alpha_k Z^{x'}_k = \infty \ almost\ surely.$ □

Proof of Proposition 4.1 Note that $\eta^{(k)} \in \mathcal{Y}$ with $|\mathcal{Y}| < \infty$ and $\eta^{(k)} \to \eta^* almost\ surely$ imply $\exists N\ s.t\ \eta^k = \eta^*, k > N$. Obviously, the step size sequence satisfies $\sum_{k=N} \alpha^{\phi}_k = \infty$ and $\sum_{k=N} (\alpha^{\phi}_k)^2 < \infty$. Thus, WLOG, it suffices to show the convergence of predictive tail probability estimates for fixed η^*.

The step size sequences of all state x' with positive visit probability satisfy Robbins-Monro condition (sum of sequence is infinite, but square sum of sequence is finite). By Lemma A.2, the update rule (14) a converges optimal value almost surely. □

A.2. Proof of Theorem 4.3

Let us introduce i.i.d. random disturbances $W_1, \ldots, W_T$ to describe the randomness in the random cost realizations, i.e., there exists a function $C(\cdot)$ such that the random costs are determined as $C_t = c(X_t, A_t, W_t)$. The trajectory H_{T+1} is drawn from density distribution $p(H_{T+1}|\theta)$ described as

$$p(H_{T+1}|\theta) = \prod_{t=1}^{T} \pi_\theta(A_t|X_t) p(X_{t+1}|X_t, A_t) p(W_t)$$

Applying the score function trick gives

$$\frac{\partial}{\partial \theta} J(\theta, \eta, \phi) = \frac{\partial}{\partial \theta} \mathbb{E}\left[\sum_{t=1}^{T} \hat{Q}_t C_t\right] = \mathbb{E}\left[\frac{\partial}{\partial \theta} \log p(H_{T+1}|\theta) \times \sum_{t=1}^{T} \hat{Q}_t C_t\right].$$

Since the state transition dynamics and the random disturbance distribution do not depend on the policy, we have

$$\begin{aligned}\frac{\partial}{\partial\theta}\log p(H_{T+1}|\theta) &= \frac{\partial}{\partial\theta}\left(\sum_{t=1}^{T}\log\pi_\theta(A_t|X_t)+\sum_{t=1}^{T}\log p(X_{t+1}|X_t,A_t)+\sum_{t=1}^{T}\log p(W_t)\right)\\ &= \frac{\partial}{\partial\theta}\left(\sum_{t=1}^{T}\log\pi_\theta(A_t|X_t)\right)\\ &= \sum_{t=1}^{T}\frac{\partial\log\pi_\theta(A_t|X_t)}{\partial\theta}.\end{aligned}$$

Therefore,

$$\frac{\partial}{\partial\theta}J(\theta,\eta,\phi)=\mathbb{E}\left[\left(\sum_{t=1}^{T}\frac{\partial\log\pi_\theta(A_t|X_t)}{\partial\theta}\right)\times\left(\sum_{t=1}^{T}\hat{Q}_tC_t\right)\right]=\mathbb{E}\left[\sum_{t=1}^{T}\hat{\nabla}_\theta J_t^{(1)}\right].$$

Also note that, for any $s<t$, since $\hat{Q}_s$ and C_s are measurable with respect to H_t, we have

$$\begin{aligned}\mathbb{E}\left[\mathbb{E}\left[\frac{\partial\log\pi_t^\theta(A_t|X_t)}{\partial\theta}\cdot\hat{Q}_sC_s\middle|H_t\right]\right] &= \mathbb{E}\left[\mathbb{E}\left[\frac{\partial\log\pi_t^\theta(A_t|X_t)}{\partial\theta}\cdot\hat{Q}_sC_s\middle|H_t\right]\right]\\ &= \mathbb{E}\left[\mathbb{E}\left[\frac{\partial\log\pi_t^\theta(A_t|X_t)}{\partial\theta}\middle|H_t\right]\cdot\hat{Q}_sC_s\right]\\ &= \mathbb{E}\left[0\cdot\hat{Q}_sC_s\right]\\ &= 0.\end{aligned}$$

Therefore,

$$\frac{\partial}{\partial\theta}J(\theta,\eta,\phi)=\mathbb{E}\left[\sum_{t=1}^{T}\left(\frac{\partial\log\pi_\theta(A_t|X_t)}{\partial\theta}\times\sum_{s=t}^{T}\hat{Q}_sC_s\right)\right]=\mathbb{E}\left[\sum_{t=1}^{T}\hat{\nabla}_\theta J_t^{(2)}\right].$$

□

A.3. Other Proofs

Proof of Proposition 4.2 The assumption that total cost distribution has no probability mass at $\eta^{\star,\pi}:=\mathrm{VaR}_q^\pi[C_{1:T}]$ implies that there is a unique solution for (5), $\eta^{\star,\pi}$ (Rockafellar & Uryasev, 2002). If step size sequence of η satisfies Robbins-Monro condition, the update rule (13) a converges optimal value almost surely. □

Proof of Lemma 4.4 Let $Q_T:=Q_T^{\pi,\eta}$ and $Q_t:=Q_t^{\pi,\eta}$.

$$\begin{aligned}\mathrm{Var}(Q_TC_t) &\overset{(a)}{=} \mathbb{E}\big[\mathrm{Var}(Q_TC_t)|H_{t+1}\big]+\mathrm{Var}\big(\mathbb{E}[Q_TC_t|H_{t+1}]\big)\\ &\geq \mathrm{Var}\big(\mathbb{E}[Q_TC_t|H_{t+1}]\big)\\ &\overset{(b)}{=} \mathrm{Var}\big(C_t\cdot\mathbb{E}[Q_T|H_{t+1}]\big)\\ &\overset{(c)}{=} \mathrm{Var}\big(C_tQ_t\big),\end{aligned}$$

where step (a) uses the law of total variance, step (b) uses the fact that H_{t+1} includes C_t, and step (c) uses the definition of Q_t. □

Proof of Proposition 4.5 Note that, for a vector $A=(A_1,\ldots,A_m)^\top$, $\mathbb{V}[A]=\mathrm{trace}(\mathrm{Cov}[A])=\mathrm{trace}(\mathbb{E}\left[A-\mathbb{E}[A]\right]\mathbb{E}\left[A-\mathbb{E}[A]\right]^\top)=\sum_{i=1}^m\left(\mathbb{E}[A_i^2]-\mathbb{E}[A_i]^2\right)$. Then,

$$
\begin{aligned}
&\mathbb{V}\left[\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta}\cdot\sum_{s=1}^{T}\hat{Q}_s C_s\right]-\mathbb{V}\left[\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta}\cdot\sum_{s=t}^{T}\hat{Q}_s C_s\right]\\
&=\sum_{i=1}^{m}\left(\mathbb{E}\left[\left(\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}\cdot\sum_{s=1}^{T}\hat{Q}_s C_s\right)^2\right]-\mathbb{E}\left[\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}\cdot\sum_{s=1}^{T}\hat{Q}_s C_s\right]^2\right)\\
&-\sum_{i=1}^{m}\left(\mathbb{E}\left[\left(\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}\cdot\sum_{s=t}^{T}\hat{Q}_s C_s\right)^2\right]-\mathbb{E}\left[\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}\cdot\sum_{s=t}^{T}\hat{Q}_s C_s\right]^2\right)\\
&\overset{(a)}{=}\sum_{i=1}^{m}\left(\mathbb{E}\left[\left(\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}\cdot\sum_{s=1}^{T}\hat{Q}_s C_s\right)^2\right]-\mathbb{E}\left[\left(\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}\cdot\sum_{s=t}^{T}\hat{Q}_s C_s\right)^2\right]\right)\\
&=\sum_{i=1}^{m}\left(\mathbb{E}\left[\frac{\partial \log \pi_\theta(A_t|X_t)}{\partial \theta_i}^2\cdot\left(\left(\sum_{s=1}^{T}\hat{Q}_s C_s\right)^2-\left(\sum_{s=t}^{T}\hat{Q}_s C_s\right)^2\right)\right]\right)\\
&\overset{(b)}{\geq} 0,
\end{aligned}
$$

where step (a) uses Theorem 4.3 and step (b) uses the non-negativity condition assumed on the cost. □

B. Numerical Experiment Details

B.1. A Naïve CVaR (NCVaR) Policy Gradient Algorithm

We simply denote by NCVaR a naïve version of PCVaR that does not include the predictive tail probabilities. Without introducing the prediction model parameter ϕ, it solves

$$
\min_{\theta\in\Theta}\left\{J(\theta,\eta):=\mathbb{E}\left[\mathbb{I}\{C_{1:T}\geq\eta\}C_{1:T}\right]\;\middle|\;\eta\in\arg\min_{\eta'\in\mathbb{R}}L(\theta,\eta')\right\},
$$

where $L(\theta,\eta):=\mathbb{E}\left[q\eta+(C_{1:T}-\eta)^+\right]$ shares the same definition. As done for PCVaR, θ optimization is done by applying the SGD with respect to the objective $J(\theta,\eta)+\lambda_L\frac{\partial}{\partial\eta}L(\theta,\eta)$, and η optimization is done by applying the SGD with respect to the objective $L(\theta,\eta)$. The gradient estimate $\hat{\nabla}_\theta J$ can be computed as, for example,

$$
\hat{\nabla}_\theta J(H_{T+1})=\left(\sum_{t=1}^{T}\frac{\partial \log \pi_t^\theta(A_t|X_t)}{\partial\theta}\right)\cdot\mathbb{I}\{C_{1:T}\geq\eta\}C_{1:T},
$$

which does not enjoy time-decomposition unlike PCVaR. Below Algorithm 2 sketches the implementation of NCVaR algorithm.

Algorithm 2 Naïve CVaR Policy Gradient

1: Initialize θ, η.
2: **for** *episode* $k=1,2,3,\ldots$ **do**
3: Run π^θ and obtain a sample trajectory $H_{T+1}=(X_1,A_1,C_1,\ldots,X_T,A_T,C_T)$.
4: Update η through (13).
5: Update θ through

$$
\theta\leftarrow\theta-\alpha^\theta\cdot\left(\hat{\nabla}_\theta J+\lambda_L\hat{\nabla}^2_{\theta\eta}L\right).
$$

6: **end for**

B.2. Continuous Blackjack Game Details

Bernstein polynomials As illustrated in Section 5.1, for PCVAR, we introduce a prediction model $f^{\phi}(x, c) = \mathbb{I}\{c < 0\} \cdot B_5^{\phi^1}(x/21) + \mathbb{I}\{c \geq 0\} \cdot B_5^{\phi^2}(x/21)$. Here, each $B_5^{\phi^i} : [0, 1] \to \mathbb{R}$ is the Bernstein polynomial defined as

$$B_5^{\phi^i}(x) = \sum_{k=0}^{5} \phi_k^i \times \binom{5}{k} x^k (1-x)^{5-k}.$$

Since f^{ϕ} is linear in ϕ, ϕ optimization ($\min_{\phi} M(\theta, \eta, \phi)$) is just a simple linear regression task.

Hyperparameters/initialization We use the following configurations in the simulations.

- θ learning rate (all): $\alpha^{\theta} = 0.005$.
- η learning rate (NCVaR & PCVaR only): $\alpha^{\eta} = 0.1$.
- ϕ learning rate (PCVaR only): $\alpha^{\phi} = 0.01$.
- θ initialization (all): (a) $\theta = 16.38$ which the risk-neutral optimal solution, and (b) $\theta = 13.5$ which is an arbitrary number smaller than the CVaR-optimal solution.
- η initialization (NCVaR & PCVaR only): (a) $\eta = 10.96$ which is the estimated VaR_q value for $\theta = 16.38$, and (b) $\eta = 8.96$ which is the estimated VaR_q value for $\theta = 13.5$.
- ϕ initialization (PCVaR only): (a) $\phi^1 = (0.097, 0.128, -0.059, 0.597, -1.454, 4.091)^{\top}, \phi^2 = (15.253, -7.874, 3.751, -1.459, 0.470, 0.071)^{\top}$ which are the fitted prediction model parameters for $\theta = 16.38$, and (b) $\phi^1 = (0.101, 0.082, 0.025, 0.773, -4.375, 19.975)^{\top}$, $\phi^2 = (15.253, -7.874, 3.751, -1.459, 0.470, 0.071)^{\top}$ which are the fitted prediction model parameters for $\theta = 13.5$.

B.3. Pair Trading Details

For initialization, we use the first 330 days (Data 1) for the initial training of the trading strategy and evaluate the strategy during the remaining days (Data 2). We periodically re-optimize the strategy every other days using the prior ten days of data (Figure 4).

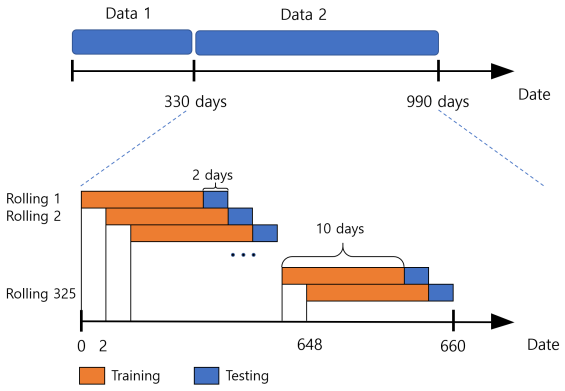

Figure 4. Data split and algorithm evaluation scheme in the pair trading example. Dataset 1 is utilized for initialization, and dataset 2 is utilized for evaluation.

Followings are the detailed configurations of individual algorithms. Regarding the choice of learning rate and batch size, we attempt multiple learning rate values, $\{0.00005, 0.0001, 0.0005, 0.001, 0.005, 0.01\}$, along with multiple batch size values, $\{1, 8, 16, 32\}$, and the best set of configuration is adopted for each algorithm.

- PCVaR

  - Learning rates: $\alpha^\theta = 0.0005$, $\alpha^\eta = 0.0005$, $\alpha^\phi = 0.0005$.
  - Initialization: θ, η, and ϕ are optimized/fitted to Data 1 via PCVaR algorithm.
  - Batch size: $B = 1$.

- NCVaR
  - Learning rates: $\alpha^\theta = 0.001$, $\alpha^\eta = 0.001$.
  - Initialization: same as PCVaR configuration, but utilize θ and η only.
  - Batch size: $B = 1$.

- GCVaR
  - Learning rate: $\alpha^\theta = 0.001$.
  - Initialization: same as PCVaR configuration, but utilize θ only.
  - Batch size: $B = 1$.

- Risk-neutral policy gradient
  - Learning rates: $\alpha^\theta = 0.0001$.
  - Initialization: θ is optimized to Data 1 via the risk-neutral policy gradient algorithm.
  - Batch size: $B = 1$.