CAUCHY – Jurnal Matematika Murni dan Aplikasi
Volume 11 (1) (2026), Pages 426-446
p-ISSN: 2086-0382; e-ISSN: 2477-3344

From Risk-Neutral to Risk-Sensitive Reinforcement Learning: Actor–Critic vs REINFORCE with Tail-Based Risk Measures

Aprida Siska Lestia[1,2], Adhitya Ronnie Effendie[1*], Made Tantrawan[1], Muhammad Rafli Azrarsyah[1], Karenditha Gratcia Mananta[1], Naufal Faiq Muyassar[1], Muhammad Faisa Ardra R.[1], and Rafael Bona Kingson Girsang[1]

[1] *Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Yogyakarta, Indonesia*
[2] *Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Lambung Mangkurat, South Kalimantan, Indonesia*

Abstract

This study investigates risk-sensitive reinforcement learning (RL) for portfolio decision-making under empirically heavy-tailed return distributions. We compare two policy-gradient architectures—REINFORCE with baseline (REINFORCE-BL) and batched Advantage Actor–Critic (A2C-B)—and examine how tail-based risk measures modify learning dynamics and robustness. Quantitative diagnostics confirm substantial excess kurtosis and strong rejection of normality in daily NASDAQ returns, motivating the integration of tail-sensitive objectives. Risk sensitivity is introduced at the episodic level through penalties based on Value at Risk (VaR), Conditional Value at Risk (CVaR), and Entropic Value at Risk (EVaR) at the 95% confidence level. Experiments are conducted in a multi-asset portfolio exposure-control environment, with performance evaluated across multiple random seeds using both training dynamics and out-of-sample financial metrics (CAGR, volatility, Sharpe ratio, drawdown, and realized tail risk). Results show that while both architectures perform comparably under the risk-neutral objective, actor–critic learning exhibits greater stability and lower dispersion under coherent tail penalties. In particular, CVaR and EVaR objectives lead to smoother convergence and reduced instability compared to VaR, especially for A2C-B. Statistical tests indicate that performance differences become more pronounced under coherent tail-risk objectives. These findings highlight the interaction between heavy-tailed environments, coherent risk measures, and algorithmic architecture, suggesting that actor–critic methods provide a more robust foundation for risk-sensitive RL in financial settings exposed to extreme events.

Keywords: risk-sensitive reinforcement learning; actor–critic; coherent risk measures; CVaR; EVaR; heavy-tailed returns; portfolio optimization.

1. Introduction

Financial markets are dynamic and non-stationary, so asset selection strategies require methods that are adaptive and able to learn from data flowing over time. Reinforcement Learning (RL) provides an agent–environment interaction-based learning framework: the agent observes the market state, selects an action, and receives a return (reward) that is evaluated quantitatively

*Corresponding author. E-mail: adhityaronnie@ugm.ac.id

Submitted: January 09, 2026 Reviewed: February 06, 2026 Accepted: February 20, 2026
DOI: https://doi.org/10.18860/cauchy.v11i1.40309

[1]. A number of studies show that RL is relevant for understanding market anomalies, investor behavior, as well as trading strategy design and market exposure decision [2, 3, 4].

According to [1], learning in RL occurs through trial-and-error, reward-based feedback, and balancing between exploration and exploitation. Based on how to represent strategy, traditional RL algorithms are grouped into value-based methods (such as multi-armed bandit (MAB), Dynamic Programming, Monte Carlo, and Temporal-Difference (TD) learning), policy-based methods (such as policy gradient), as well as hybrid methods (actor–critic). Classical RL algorithms are generally *risk-neutral*, namely designing optimal policies by maximizing expected reward without accounting for variance or potential extreme losses. Traditional reinforcement learning typically models the expectation of the return rather than its full distribution. Recent work proposes a distributional perspective, emphasizing the importance of modeling the value distribution [5]. This approach is adequate for simple simulation environments, but less suitable for financial and insurance domains that are highly sensitive to risk and *tail events*. In that context, the decision-making goal should not only be "highest possible return", but also consider stability, *drawdown*, and extreme losses that are rare but high-impact. This confirms the need for development of RL approaches that are sensitive to risk, especially for applications in finance and insurance.

On the other hand, risk management literature has developed various risk measures that are richer than just variance or volatility. Value at Risk (VaR) provides a loss quantile limit, while Conditional Value at Risk (CVaR) and Entropic Value at Risk (EVaR) emphasize the tail behavior of loss distribution and have been studied widely in the context of market exposure decision [6, 7, 8]. VaR is widely used in practice but is not coherent because it fails to satisfy subadditivity, while CVaR and EVaR are included in *coherent risk measures* that are more consistent with diversification principles; EVaR even becomes a tight upper bound for VaR and CVaR and possesses *strong monotonicity* properties on a broad class of distributions [7]. Risk-sensitive approaches started to be adopted at the multi-armed bandit level by including risk metrics like variance and CVaR as optimization criteria. Research by [9] proposed a tail risk-based exploration index that prioritizes resilience against large losses, instead of just chasing average returns.

Various bandit models based on mean–variance and tail risk indices were developed to make exploration more conservative towards worst-case scenarios. This initiative paved the way for Markov Decision Process (MDP) based RL algorithms that integrate risk measures into the learning process, both for value-based methods and policy-based methods.

Among policy-gradient approaches, REINFORCE represents a pure likelihood-ratio estimator whose gradient is unbiased but may exhibit high variance, particularly in stochastic environments [10]. In contrast, Actor–Critic incorporates value-function approximation to reduce gradient variance through temporal-difference learning. This variance-reduction mechanism is especially relevant in financial settings with heavy-tailed returns, where extreme observations may destabilize pure policy-gradient updates.

In the realm of risk measures itself, [7] introduced Entropic Value at Risk (EVaR) as a new coherent risk measure that becomes the tightest upper bound for VaR and CVaR. They showed that EVaR-based market exposure control for a multi-asset portfolio can be formulated as a smooth (differentiable) convex program, so it can be solved efficiently with primal–dual interior-point algorithms without direct dependence on sample size. Numerical results show that the EVaR method works more efficiently than the CVaR approach on a large scale, while simultaneously producing portfolios that are more stable and conservative towards extreme risk. These findings provide an important theoretical foundation for risk-based decision making and application of coherent risk measures in stochastic optimization as well as risk-sensitive Reinforcement Learning.

Recent literature shows several ways to combine risk measures with classical RL algorithms. [11] extended the standard policy gradient method to be risk-sensitive by including variability

measures like variance and CVaR into the optimization objective. They developed an actor–critic algorithm that approximates value with CVaR-SGD and GCVaR techniques, so the agent is able to learn policies that avoid losses on the tail of distribution without fully sacrificing average performance. Meanwhile, a policy gradient algorithm based on Entropic Value at Risk (EVaR) was introduced by Neufeld et al., who showed that EVaR is not only coherent theoretically but can also be utilized to encourage exploration towards high-impact returns in an RL environment. In general, those results show that the integration of risk measures like variance, CVaR, and EVaR into RL improves the ability of classical algorithms in managing the *trade-off* between reward exploitation and extreme risk limitation.

Overall, the studies above show that the integration of risk measures, specifically CVaR and EVaR, into the RL framework has the potential to produce policies that are safer and *robust*. However, most research still focuses on relatively simple RL tasks or on general theoretical formulation, not on market exposure control for a multi-asset portfolio with policy-based algorithms like REINFORCE and actor–critic.

Building upon this background, the present study investigates risk-sensitive reinforcement learning in the context of stock market exposure control for a multi-asset portfolio. Unlike classical portfolio weight optimization, the problem is formulated as a discrete market exposure decision process, allowing a controlled comparison between risk-neutral and risk-sensitive learning objectives. The novelty of this work lies not in proposing new risk measures, but in providing a systematic empirical evaluation of how coherent tail-based measures—CVaR and EVaR—affect learning stability and algorithmic robustness when integrated into policy-gradient architectures. Specifically, REINFORCE with baseline and batched Advantage Actor–Critic are compared under risk-neutral and risk-sensitive specifications using heavy-tailed NASDAQ return data. This design enables an explicit analysis of the trade-off between total return and stability under extreme market conditions, and highlights the role of algorithmic architecture in the effective integration of coherent risk measures into reinforcement learning.

2. Methods

This section presents the methodological framework used in this study. We first describe the environment setup and reward construction, followed by the formulation of risk-sensitive objectives (VaR, CVaR, and EVaR). We then detail the integration of these objectives into the REINFORCE and A2C-B algorithms, and finally outline the experimental protocol.

2.1. Policy Gradient & Actor–Critic

The approach used in this research belongs to the family of policy-based reinforcement learning. In this framework, the policy $\pi_\theta(a \mid s)$ is parameterized directly by the parameter vector θ, and the learning objective is to adjust θ in a direction that increases the expected return $J(\theta)$. This formulation represents the standard risk-neutral policy-gradient objective, which later serves as a baseline before introducing risk-sensitive modifications.

In general, the gradient of the objective function can be written as

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta} \left[\nabla_\theta \log \pi_\theta(a \mid s)\, Q^\pi(s,a) \right],$$

where $Q^\pi(s,a)$ denotes the action-value function under policy π. In Monte Carlo policy-gradient methods such as REINFORCE, this action-value function is approximated using the realized discounted return obtained from sampled trajectories.

The REINFORCE algorithm is the simplest form of policy-gradient method. In this algorithm, the agent generates one episode $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$ by following the policy π_θ. For each time step t, the discounted return is computed as

$$G_t = \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k,$$

and the policy parameters are updated according to

$$\theta \leftarrow \theta + \alpha\, \gamma^t (G_t - b)\, \nabla_\theta \log \pi_\theta(A_t \mid S_t),$$

where b denotes a baseline used to reduce the variance of the gradient estimator. In this study, the baseline is chosen as the average episodic return.

While REINFORCE provides an unbiased estimate of the policy gradient and is conceptually simple, its gradient estimates are known to suffer from high variance, particularly in stochastic environments with noisy or heavy-tailed returns. This limitation motivates the use of actor–critic methods, which incorporate value-function approximation to reduce gradient variance and improve learning stability.

To address the high variance of pure policy-gradient methods, this study employs the Actor–Critic approach, which combines two components: an actor and a critic. The actor updates the policy parameters θ based on the gradient of the action log-probability, while the critic approximates the state-value function $V(s; w)$ and provides feedback through the temporal-difference (TD) error. This framework allows variance reduction by incorporating value-function information into the policy update, leading to more stable learning dynamics.

In particular, this study adopts an episodic batched Advantage Actor–Critic scheme (A2C-B). At each time step t, the critic computes the TD error

$$\delta_t = r_t + \gamma V(s_{t+1}; w) - V(s_t; w),$$

which serves as an estimate of the advantage function. The actor then updates the policy parameters according to

$$\theta \leftarrow \theta + \alpha\, \delta_t \nabla_\theta \log \pi_\theta(a_t \mid s_t),$$

while the critic updates the value-function parameters as

$$w \leftarrow w + \beta\, \delta_t \nabla_w V(s_t; w).$$

Parameter updates are performed using batched trajectories collected over complete episodes, which improves gradient stability compared to step-wise updates. By using TD-based advantage estimation, the Actor–Critic framework mitigates gradient variance arising from noisy and heavy-tailed reward signals. This property makes A2C-B particularly suitable for risk-sensitive reinforcement learning in stochastic financial environments, where extreme returns can otherwise destabilize learning.

The Actor–Critic framework is theoretically motivated by its variance-reduction property through value-function approximation, which has been extensively studied in the reinforcement learning literature [12], [13]. In this study, REINFORCE is employed as a baseline policy-gradient method to represent learning without explicit variance correction, while Actor–Critic is selected as a structured alternative for improving learning stability in stochastic environments.

2.2. Risk Measures

Risk measure is defined as a functional that maps loss random variable X to real number $\rho(X)$, which is interpreted as the minimum capital amount required so that a financial position is considered eligible or acceptable [8]. Formally, risk measure is a mapping $\rho : \mathcal{X} \to \mathbb{R}$, with $\mathcal{X}$ the set of loss random variables relevant in the model.

In this study, the reinforcement learning (RL) agent is trained on episodic cumulative *returns*. To maintain a consistent sign convention with the risk-measure literature, we define the corresponding episodic *loss* as

$$X(\tau) := -R(\tau),$$

where $R(\tau)$ denotes the cumulative (discounted) return of an episode/trajectory τ. All risk measures in this paper are applied to the loss variable $X(\tau)$.

In [7], risk measures are used among others to determine capital requirements and solvency levels of insurance as well as banking companies, help the process of pricing financial products and reinsurance contracts, compare portfolios under uncertainty conditions, as well as support market exposure control for a multi-asset portfolio and risk control in financial and actuarial contexts. In other words, the choice of risk measure plays a direct role towards how risk is measured, reported, and managed.

[6] proposed four axioms that must be satisfied so that a risk measure is called coherent. A functional ρ is called *coherent risk measure* if for every loss random variable X, Y and every constant $c \in \mathbb{R}$ holds: (i) *Translation invariance*: $\rho(X + c) = \rho(X) + c$, which means addition of deterministic component only shifts the magnitude of risk by the value of that constant; (ii) *Subadditivity*: $\rho(X + Y) \leq \rho(X) + \rho(Y)$, which reflects that diversification must not increase total risk; (iii) *Monotonicity*: if $X \leq Y$ (almost surely), then $\rho(X) \geq \rho(Y)$, so a portfolio with worse results in every state will have larger risk; (iv) *Positive homogeneity*: $\rho(\lambda X) = \lambda\rho(X)$ for every $\lambda \geq 0$, which indicates that doubling position will double also the risk level.

In this research considered three risk measures that are widely used, namely Value at Risk (VaR), Conditional Value at Risk (CVaR), and Entropic Value at Risk (EVaR). Value at Risk at confidence level $\alpha \in (0, 1)$ is defined as loss distribution quantile,

$$\mathrm{VaR}_\alpha(X) = \inf\{x \in \mathbb{R} : \mathbb{P}(X \leq x) \geq \alpha\}.$$

VaR is widely used in financial industry regulation because its concept is simple, however it does not satisfy subadditivity so it is not a coherent risk measure [8].

CVaR which is also known as *expected shortfall* (ES), is the expected value of loss on the α worst tail of loss distribution,

$$\mathrm{CVaR}_\alpha(X) = \mathbb{E}[X \mid X \geq \mathrm{VaR}_\alpha(X)].$$

CVaR has been proven to satisfy the four coherence axioms and therefore is a prime example of coherent risk measure. This measure is more sensitive to tail distribution changes compared to VaR and is used widely in market exposure control for a multi-asset portfolio as well as *risk-sensitive reinforcement learning* [14].

EVaR is a coherent, exponential-moment-based risk measure defined by

$$\mathrm{EVaR}_\alpha(X) = \inf_{\lambda > 0} \frac{1}{\lambda}\Big(\log \mathbb{E}[e^{\lambda X}] - \log(1 - \alpha)\Big),$$

where $\lambda > 0$ is an auxiliary optimization variable (distinct from the policy parameters). EVaR provides an upper bound for VaR and CVaR and admits a smooth objective formulation [7].

Throughout the experiments, we fix the confidence level at $\alpha = 0.95$ for all risk specifications. In the next subsection, these risk measures are integrated into the RL training objective by evaluating VaR_α, CVaR_α, or EVaR_α on the empirical distribution of episodic losses $\{X(\tau)\}$ collected at the end of each episode. The resulting risk estimate replaces the standard cumulative return in the policy update step for REINFORCE-BL and A2C-B. For EVaR, the auxiliary parameter $\lambda > 0$ is optimized numerically at each training iteration to obtain the tightest exponential risk bound for the current empirical loss distribution.

2.3. Market Environment & Experimental Setup

This study formulates the stock market exposure control problem for a multi-asset portfolio as a Markov Decision Process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$. At each discrete time step t, the agent observes a state S_t consisting of market information, including selected technical indicators and recent historical price features. Based on this information, the agent selects an action A_t representing a discrete market exposure decision for the portfolio, rather than continuous portfolio weight optimization.

After executing action A_t, the agent receives a scalar reward R_t that reflects the portfolio performance over the corresponding period. In this study, rewards are constructed from scaled changes in total portfolio value rather than raw logarithmic returns. Let V_t denote the total portfolio value at time t. The per-step reward is defined as

$$R_t = \text{clip}\big(\kappa(V_{t+1} - V_t), -1, 1\big),$$

where κ is a fixed reward-scaling constant used to stabilize training, and the clipping operation ensures bounded reward magnitudes. Cumulative episodic rewards therefore correspond to the undiscounted sum $\sum_{t=0}^{T} R_t$ under this scaling scheme.

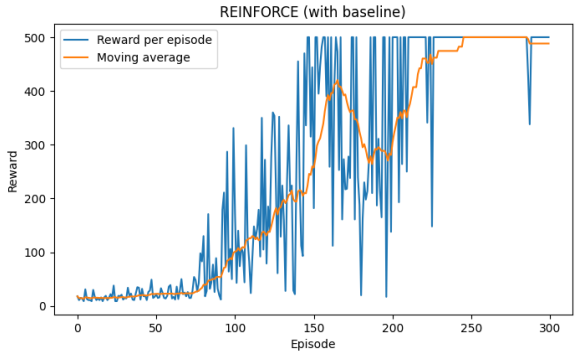
In the standard risk-neutral setting, the objective is to find an optimal policy π^* that maximizes the expected discounted return

$$\mathbb{E}\left[\sum_{t=0}^{T} \gamma^t R_t\right].$$

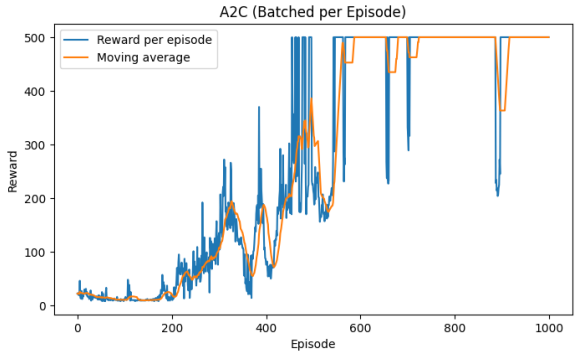
Two model-free reinforcement learning algorithms are considered: REINFORCE with baseline (REINFORCE-BL) and Advantage Actor–Critic with episodic batching (A2C-B). These algorithms provide complementary learning architectures that allow us to examine the effect of variance reduction and stability when extending the objective to risk-sensitive formulations.

Before applying the proposed learning framework to financial market data, the correctness of the algorithmic implementation is first validated using the CartPole environment from OpenAI Gym. This environment serves as a standard benchmark for verifying policy-gradient and actor–critic implementations.

In the CartPole task, the agent observes a four-dimensional continuous state vector (cart position, cart velocity, pole angle, and pole angular velocity) and selects discrete actions to maintain pole balance. Both REINFORCE-BL and A2C-B are implemented using feedforward neural networks with a single hidden layer of 128 units and ReLU activation. Parameter updates are performed using the Adam optimizer, with gradient clipping applied to prevent gradient explosion and a small entropy bonus added to encourage exploration.



(a) REINFORCE learning results on CartPole environment.



(b) Actor-critic learning results on CartPole environment

Fig. 1: Learning curves of REINFORCE-BL and A2C-B on the CartPole environment, used to validate the correctness and stability of the algorithmic implementations.

Learning results in the CartPole environment as presented in Fig. 1 show that REINFORCE-BL reaches reward close to maximum relatively quickly, while A2C-B gives more stable updates with low fluctuation. This finding becomes the basis for initial *hyperparameter* configuration selection for the stock portfolio case study.

The dataset is divided into training and evaluation periods using a chronological split to avoid look-ahead bias. All experiments are conducted on the training period, while performance metrics are reported on the evaluation period. To improve reproducibility and reduce randomness

effects, each experiment is repeated over multiple random seeds, and reported results correspond to averaged performance across runs.

2.4. Risk-Sensitive Objective Integration

To provide a baseline tail-based objective, a VaR-sensitive formulation is also considered. Using the episodic loss variable $X(\tau) = -R(\tau)$, the VaR-based objective is defined as

$$J_{\mathrm{VaR}}(\theta) = -\mathrm{VaR}_\alpha(X_\theta),$$

where $\mathrm{VaR}_\alpha(X_\theta)$ denotes the empirical α-quantile of the distribution of episodic losses induced by policy π_θ. Although VaR does not satisfy subadditivity and is therefore not coherent, it is included for comparative purposes against coherent risk measures such as CVaR and EVaR.

To overcome limitations of the risk-neutral approach, experimental design is expanded by integrating coherent risk measures into the objective function. First, CVaR is used at confidence level α, which is formulated following [15]. Cumulative reward along one trajectory τ is denoted by $R(\tau)$, and the objective function is changed to

$$J_{\mathrm{CVaR}}(\theta) = -\mathrm{CVaR}_\alpha(X_\theta), \qquad X_\theta = -R_\theta.$$

Thus, the learning process focuses on reducing losses in the tail of the episodic loss distribution. In practice, CVaR_α is estimated as the empirical average of the worst $(1-\alpha)$ fraction of episodic losses within each training batch. The resulting tail estimate replaces the standard cumulative return in the policy update, so that parameter updates are primarily driven by trajectories associated with extreme losses. In the actor–critic version, the value function is adjusted to approximate this tail-based objective consistently with the CVaR definition.

Besides CVaR, this research also integrates EVaR as an entropy-based coherent risk measure. EVaR at confidence level α is formulated with parameter $\lambda > 0$ as

$$J_{\mathrm{EVaR}}(\theta) = -\inf_{\lambda>0} \frac{1}{\lambda}\left(\log \mathbb{E}_\tau[e^{\lambda X(\tau)}] - \log(1-\alpha)\right).$$

In this approach, policy parameter θ and risk parameter λ are optimized simultaneously to obtain the tightest risk bound. This formulation is smooth (*differentiable*) and computationally efficient so it is suitable to be integrated into policy gradient as well as actor–critic algorithms.

Having defined the three risk-sensitive objectives, we now describe how these episodic tail-based penalties are operationally integrated into the policy-gradient and actor–critic training procedures. Since the tail-risk objective is defined at the episodic level, while actor–critic learning operates through step-wise temporal-difference updates, an explicit linkage between the two levels is required. We therefore describe how the episodic penalty is incorporated into the TD-based actor–critic update.

For the actor–critic implementation (A2C-B), we incorporate the episodic tail-risk penalty through a risk-adjusted per-step reward so that both the actor update and the critic target are modified consistently. At training iteration k, we first estimate an episodic tail-risk term $\widehat{\rho}_\alpha^{(k)} \in \{\widehat{\mathrm{VaR}}_\alpha, \widehat{\mathrm{CVaR}}_\alpha, \widehat{\mathrm{EVaR}}_\alpha\}$ from the empirical distribution of episodic losses $X(\tau^{(i)}) = -R(\tau^{(i)})$ within the batch. We then distribute this episodic penalty uniformly across the episode of length T by defining

$$\tilde{r}_t^{(i)} \;=\; r_t^{(i)} \;-\; \frac{\widehat{\rho}_\alpha^{(k)}}{T}.$$

To maintain compatibility with step-wise TD learning, the episodic penalty is distributed uniformly across the horizon. Accordingly, the TD-error used by A2C-B becomes

$$\tilde{\delta}_t^{(i)} \;=\; \tilde{r}_t^{(i)} + \gamma V(s_{t+1}^{(i)}; w) - V(s_t^{(i)}; w),$$

so the actor update uses $\tilde{\delta}_t^{(i)}$ as a risk-adjusted advantage estimate, while the critic is trained by minimizing the corresponding TD loss. This makes the A2C-B risk integration explicit and consistent with the episodic risk objective. For REINFORCE-BL, the episodic tail-risk estimate directly replaces the standard cumulative return in the policy-gradient update.

Final performance evaluation is done by comparing four agent scenarios: (i) *risk-neutral* (standard REINFORCE-BL and A2C-B), (ii) *risk-sensitive* with VaR-based objective, (iii) *risk-sensitive* with CVaR-based objective, and (iv) *risk-sensitive* with EVaR-based objective. For completeness, a VaR-based objective is also considered by penalizing the α-quantile of episodic losses, although VaR is not coherent and is mainly included as a baseline for comparison.

Analysis focuses on the trade-off between total return and stability under extreme market movements, with Maximum Drawdown (MDD) reported as an ex-post risk indicator. The hypothesis that integration of coherent risk measures (CVaR and EVaR) produces policies that are more *robust* towards extreme market conditions compared to risk-neutral or VaR approaches.

To test the performance of policy gradient (REINFORCE) and actor-critic algorithms in real market environment, this research uses daily closing price data from 10 multinational company stocks traded on NASDAQ exchange. Selected companies are issuers with large capitalization and high liquidity level, like FAANG group (Meta, Apple, Amazon, Netflix, Google), Microsoft, Tesla, NVIDIA, and several other blue-chip stocks. Selection of those assets reflects a market that is quite representative so it is suitable as multi-asset market setting in RL experiment. Data used covers at least 5 years of consecutive trading days.Daily price data is changed into daily return so it forms a matrix of size $T \times 10$. At every time t, environment *state* consists of normalized daily return vector of 10 stocks, added with two scalar components representing cash proportion and held portfolio position intensity (average number of units per stock) so obtained state vector of dimension $10 + 2$. RL agent does not optimize each stock individually, but manages *equal-weight* portfolio over those ten stocks.

At every time step (trading day), the agent chooses one of three discrete actions: (i) *hold*, maintaining the existing portfolio position; (ii) *buy*, namely entering fully into the market by allocating wealth evenly across all 10 stocks (equal-weight portfolio); or (iii) *sell*, namely liquidating the entire stock position and moving wealth to cash. The reward is defined as a scaled change in total portfolio value (stock + cash) between time t and $t + 1$, after accounting for a fixed transaction cost of 20 basis points when a position change occurs. Specifically,

$$R_t = \text{clip}\big(\kappa(V_{t+1} - V_t), -1, 1\big),$$

where V_t denotes the total portfolio value at time t, and $\kappa = 10^{-3}$ is a fixed scaling constant used to stabilize training. Every episode is defined as one trading year (horizon of 252 trading days) with discount factor $\gamma = 0.99$. The model is trained using a simple neural network architecture with one hidden layer of 128 units and ReLU activation. Policy parameter optimization is performed using the Adam algorithm, with a small entropy bonus coefficient (e.g., 0.01) to maintain exploration. This configuration is applied for both REINFORCE with baseline and the actor–critic scheme.

Sensitive parameters like *learning rate* and discount factor are varied to analyze result stability. Experimental results are presented in the form of portfolio *learning curve* and *equity curve* graphs, as well as financial metric tables like total return, annual volatility, *maximum drawdown*, and Sharpe ratio. Analysis is focused on comparison between REINFORCE and actor-critic, both under *risk-neutral* as well as *risk-sensitive* specifications (VaR, CVaR, and EVaR), as well as against simple passive strategy made as baseline.

To enhance methodological clarity, Algorithm 1 summarizes the operational training procedure, detailing how tail-based risk measures are integrated into the policy-gradient updates without modifying the theoretical framework.

Algorithm 1 Risk-Sensitive Policy Gradient Training (REINFORCE-BL / A2C-B)

Require: Confidence level $\alpha \in (0,1)$; batch size N; discount factor γ; stepsizes η_θ (and η_w); objective type RN/VaR/$CVaR$/$EVaR$
Ensure: Trained policy parameters θ
Initialize policy parameters θ
if Actor–Critic **then**
  Initialize value-function parameters w
end if
if objective $= EVaR$ **then**
  Initialize $\lambda > 0$ {auxiliary risk parameter}
end if
for $k = 1, 2, \ldots, K$ **do**
  {training iterations}
  Collect episodic trajectories $\{\tau_i\}_{i=1}^N$ using policy π_θ
  for $i = 1, 2, \ldots, N$ **do**
    Compute cumulative discounted return $R(\tau_i) = \sum_{t=0}^{T-1} \gamma^t r_t^{(i)}$
    Define episodic loss $X(\tau_i) \leftarrow -R(\tau_i)$
  end for
  if objective $= VaR$ **then**
    $\widehat{\mathrm{VaR}}_\alpha \leftarrow \mathrm{Quantile}_\alpha(\{X(\tau_i)\}_{i=1}^N)$
    $J(\theta) \leftarrow -\widehat{\mathrm{VaR}}_\alpha$
  else if objective $= CVaR$ **then**
    $\widehat{\mathrm{VaR}}_\alpha \leftarrow \mathrm{Quantile}_\alpha(\{X(\tau_i)\}_{i=1}^N)$
    $\mathcal{T} \leftarrow \{i : X(\tau_i) \geq \widehat{\mathrm{VaR}}_\alpha\}$
    $\widehat{\mathrm{CVaR}}_\alpha \leftarrow \frac{1}{|\mathcal{T}|} \sum_{i \in \mathcal{T}} X(\tau_i)$
    $J(\theta) \leftarrow -\widehat{\mathrm{CVaR}}_\alpha$
  else if objective $= EVaR$ **then**
    $\lambda^\star \leftarrow \arg\min_{\lambda > 0} \frac{1}{\lambda} \Big(\log(\frac{1}{N} \sum_{i=1}^N e^{\lambda X(\tau_i)}) - \log(1-\alpha) \Big)$
    $\widehat{\mathrm{EVaR}}_\alpha \leftarrow \frac{1}{\lambda^\star} \Big(\log(\frac{1}{N} \sum_{i=1}^N e^{\lambda^\star X(\tau_i)}) - \log(1-\alpha) \Big)$
    $J(\theta) \leftarrow -\widehat{\mathrm{EVaR}}_\alpha$
  else
    $J(\theta) \leftarrow \frac{1}{N} \sum_{i=1}^N R(\tau_i)$ {risk-neutral}
  end if
  if REINFORCE-BL **then**
    $b \leftarrow \frac{1}{N} \sum_{i=1}^N R(\tau_i)$ {baseline}
    Update θ using episodic policy gradient with baseline b
  else if A2C-B **then**
    $\tilde{r}_t \leftarrow r_t$
    if objective $\in \{\mathrm{VaR}, \mathrm{CVaR}, \mathrm{EVaR}\}$ **then**
      Estimate $\hat{\rho}_\alpha$ from batch losses
      $\tilde{r}_t \leftarrow r_t - \hat{\rho}_\alpha / T$
    end if
    $\tilde{\delta}_t = \tilde{r}_t + \gamma V(s_{t+1}; w) - V(s_t; w)$
    Update actor parameters θ using $\tilde{\delta}_t$
    Update critic parameters w by minimizing TD loss
  end if
end for=0

This procedure is applied identically to REINFORCE-BL and A2C-B, with differences arising only in the gradient estimator and variance-reduction mechanism.

2.5. Hyperparameter Configuration

Performance is evaluated using standard financial metrics, including total return, volatility, Sharpe ratio, and maximum drawdown. For completeness and reproducibility, Table 1 summarizes all hyperparameters and training configurations used in the experiments.

Table 1: Summary of hyperparameters used in the reinforcement learning experiments

Category	Hyperparameter	Value / Description
MDP Setup	Discount factor γ	0.99
	Episode length T	Fixed per environment (CartPole / financial data)
	State representation	Technical indicators and recent price features (financial data); position, velocity, angle, angular velocity (CartPole)
	Action space	Discrete market exposure decisions (financial data); left/right force (CartPole)
Neural Network	Architecture	Feedforward neural network
	Hidden layers	1 hidden layer
	Hidden units	128 units
	Activation function	ReLU
Optimization	Optimizer	Adam
	Learning rate (actor) α	3×10^{-4}
	Learning rate (critic) β	1×10^{-3}
	Gradient clipping	Global norm clipping
Exploration	Entropy bonus coefficient	0.01
	Baseline (REINFORCE-BL)	Average episodic return
	Batching (A2C-B)	Updates performed using episodic batches
Risk Sensitivity	Risk measures	VaR, CVaR, EVaR
	Confidence level α	0.95
	CVaR estimation	Empirical tail average from episodic losses
	EVaR parameter λ	Optimized jointly with policy parameters
Training Scheme	Validation environment	CartPole (OpenAI Gym)
	Evaluation metric	Episodic cumulative reward (moving average reported)

3. Results and Discussion

Performance evaluation of application of RL algorithms in stock market exposure control for a multi-asset portfolio is done using NASDAQ market historical data. Discussion begins with analysis of market data statistical characteristics (*Exploratory Data Analysis*) to validate asset empirical conditions, such as presence of extreme values (*outlier*) and inter-stock correlation structure. This initial analysis is crucial to justify the urgency of using tail-based risk measures (*tail risk*) and dynamic allocation strategies in this research. Next, comparison of agent effectiveness is performed in two main frameworks: *risk-neutral* (using standard REINFORCE-BL and A2C-B) and *risk-sensitive* (integrating risk measures VaR, CVaR, and EVaR). Performance of each model is analyzed based on ability to balance *trade-off* between profit and risk stability, which is observed through dynamics of learning curve during training as well as comparison of resulting financial metrics.

3.1. Quantitative Evidence of Heavy-Tailed Returns

Fig. 2 provides a preliminary visual indication of asymmetry and the presence of extreme observations in both tails of the return distributions. Although boxplots alone are insufficient to formally establish heavy-tailed behavior, they suggest deviations from Gaussian symmetry and motivate further quantitative investigation through higher-moment diagnostics and normality testing.

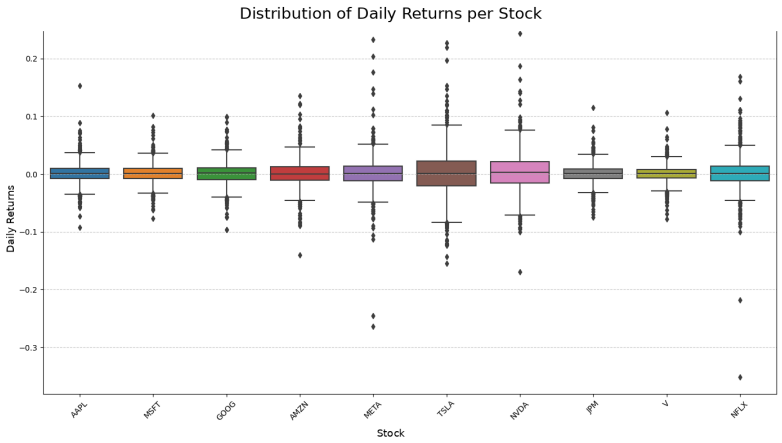


Fig. 2: Stock return distribution boxplot

Table 2 reports higher-moment statistics and normality tests. All assets exhibit strictly positive excess kurtosis, ranging from 3.21 (TSLA) to 29.39 (NFLX), substantially above the Gaussian benchmark of zero. Such magnitudes indicate pronounced tail thickness and slower decay in the distribution tails.

Table 2: Descriptive Statistics and Normality Test of Daily Returns

Asset	Mean	Std Dev	Skewness	Excess Kurtosis	JB Statistic	p-value
AAPL	0.000712	0.017485	0.457154	6.712885	2376.050043	0.000
MSFT	0.000568	0.016517	0.017185	3.943344	804.126881	0.000
GOOG	0.001046	0.019307	0.047781	3.328994	573.262426	0.000
AMZN	0.000406	0.022219	0.132431	5.089055	1343.563429	0.000
META	0.001077	0.027508	-0.137994	20.596117	21981.972714	0.000
TSLA	0.001079	0.038044	0.370577	3.210626	561.353969	0.000
NVDA	0.002531	0.032826	0.525024	4.554014	1130.208763	0.000
JPM	0.000808	0.015324	0.103190	5.092555	1343.972488	0.000
V	0.000485	0.014245	0.121337	5.752630	1715.558896	0.000
NFLX	0.000622	0.026815	-1.744937	29.385748	45380.570672	0.000
Equal-Weight Portfolio	0.000933	0.016400	0.156722	4.459888	1033.924982	0.000

Note: p-values are reported as 0.000 due to numerical rounding; in practice, all p-values are far below 0.001.

The Jarque–Bera test strongly rejects the null hypothesis of normality for every asset and the equal-weight portfolio ($p < 0.001$). The joint presence of elevated excess kurtosis and rejection of normality provides statistical confirmation that return distributions deviate materially from the Gaussian assumption, implying potential underestimation of extreme losses under mean–variance frameworks.

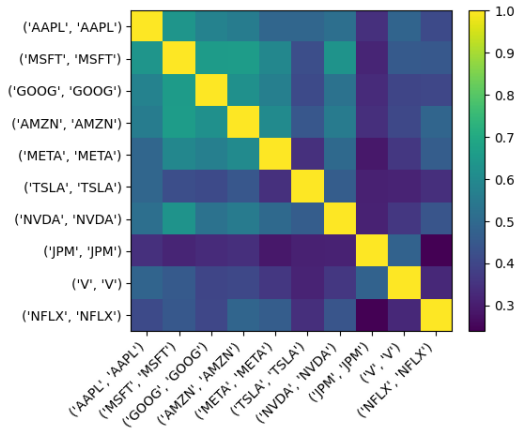


Fig. 3: Correlation matrix of daily stock returns

Fig. 3 shows that return correlations are moderate to high among several assets, suggesting non-negligible co-movement and limited diversification during market stress. Consequently, extreme losses may propagate across assets, reinforcing the relevance of tail-sensitive risk control.

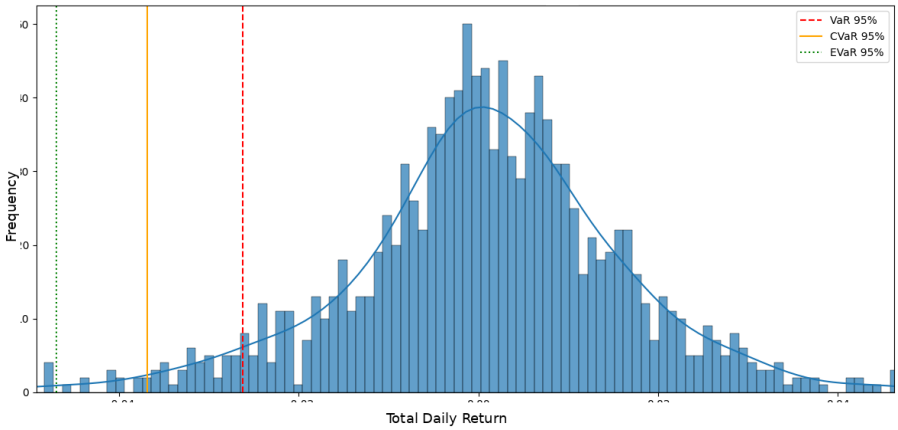


Fig. 4: Daily return distribution of 10-stock equal-weight portfolio (sum of returns)

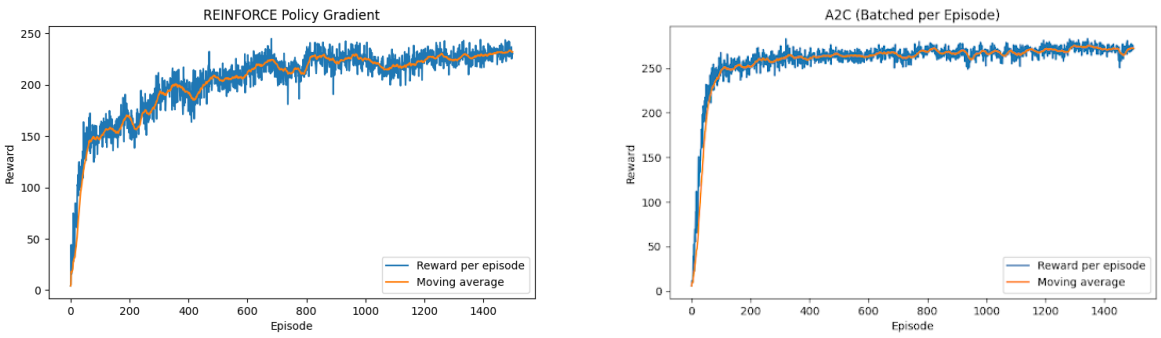
Fig. 4 presents the empirical return distribution of the equal-weight portfolio along with the 95% VaR, CVaR, and EVaR thresholds. The ordering in magnitude,

$$|\text{VaR}_{0.95}| < |\text{CVaR}_{0.95}| < |\text{EVaR}_{0.95}|,$$

reflects increasing levels of tail conservatism. VaR captures a quantile threshold, CVaR measures the conditional expectation beyond that threshold, and EVaR provides an exponential upper bound derived from Chernoff-type inequalities. This hierarchy implies progressively stronger penalization of extreme losses.

From a reinforcement learning perspective, this ordering explains the progressively more stable training dynamics observed under CVaR and EVaR objectives compared to VaR. As the risk measure becomes more tail-sensitive, the agent is driven to reduce exposure to extreme negative returns, leading to smoother convergence and improved downside stability.

3.2. Reinforcement Learning Training Performance

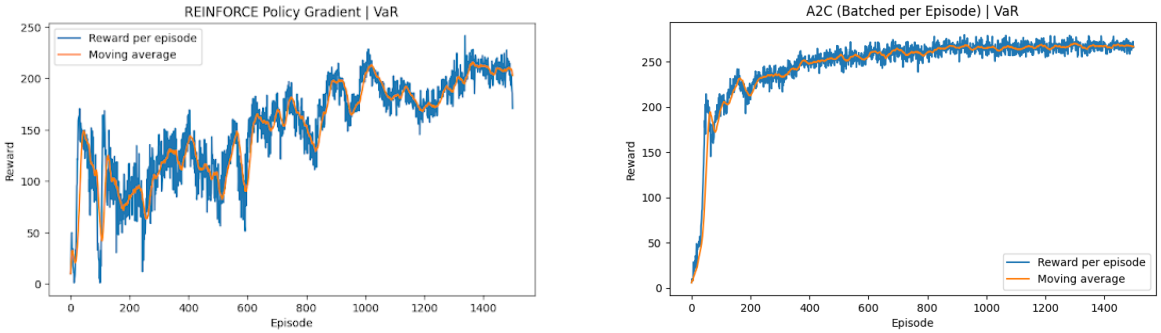


(a) REINFORCE-BL (risk-neutral) **(b)** A2C-B (risk-neutral)

Fig. 5: RL agent learning curve on 10 stock portfolio in risk-neutral scenario.

Fig. 5 displays the learning curve of both algorithms in the risk-neutral scenario. In both panels, reward per episode rises sharply in the first 50–100 episodes and then stabilizes in high range, while the *moving average* line depicts a smooth convergence trend. REINFORCE-BL (Panel a) reaches reward close to maximum but with relatively large inter-episode fluctuation, reflecting high variance of pure policy gradient. Conversely, A2C-B (Panel b) converges faster with a

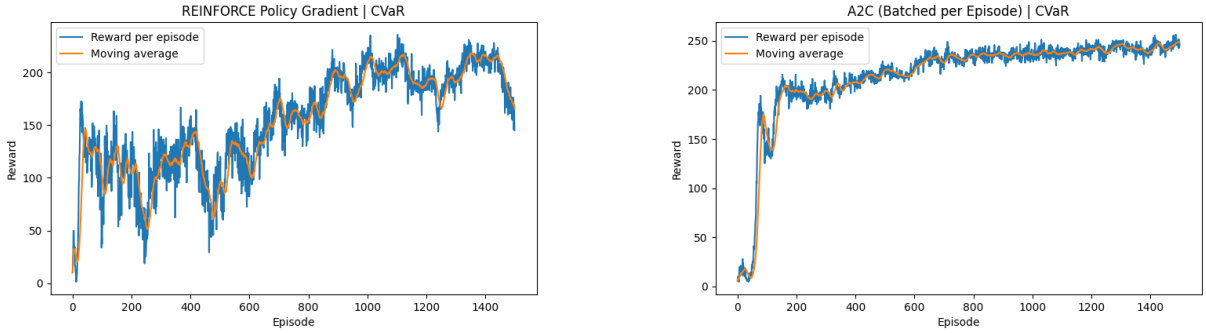
smoother curve, so it becomes a *baseline* to compare risk measure integration effect in the next section.



(a) REINFORCE-BL | VaR **(b)** A2C-B | VaR

Fig. 6: Learning curve with Value at Risk (VaR) penalty on REINFORCE-BL and A2C-B.

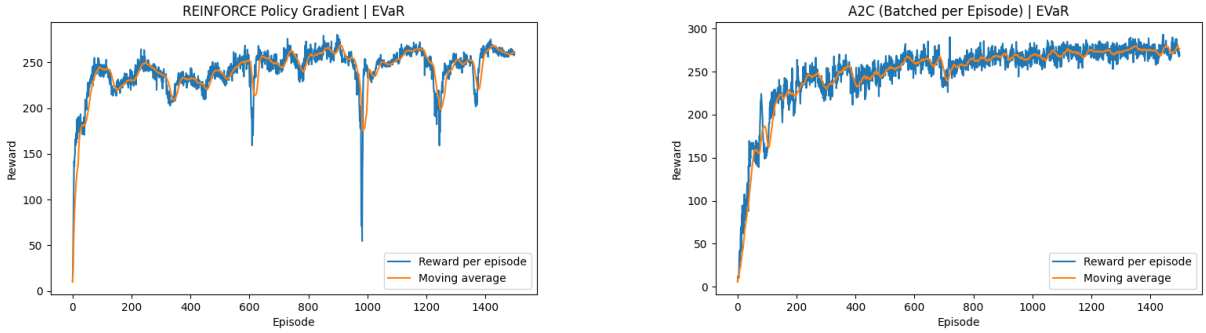
In Fig. 6, VaR penalty makes REINFORCE-BL learning curve (panel a) become much more fluctuating with several sharp reward drops, although average reward keeps increasing and ends around 200. Meanwhile, A2C-B (Panel b) still shows fast convergence towards range 250–260 with relatively smooth curve, so VaR integration only slightly affects reward level and actor–critic algorithm stability.



(a) REINFORCE-BL | CVaR **(b)** A2C-B | CVaR

Fig. 7: Learning curve with Conditional Value at Risk (CVaR) penalty.

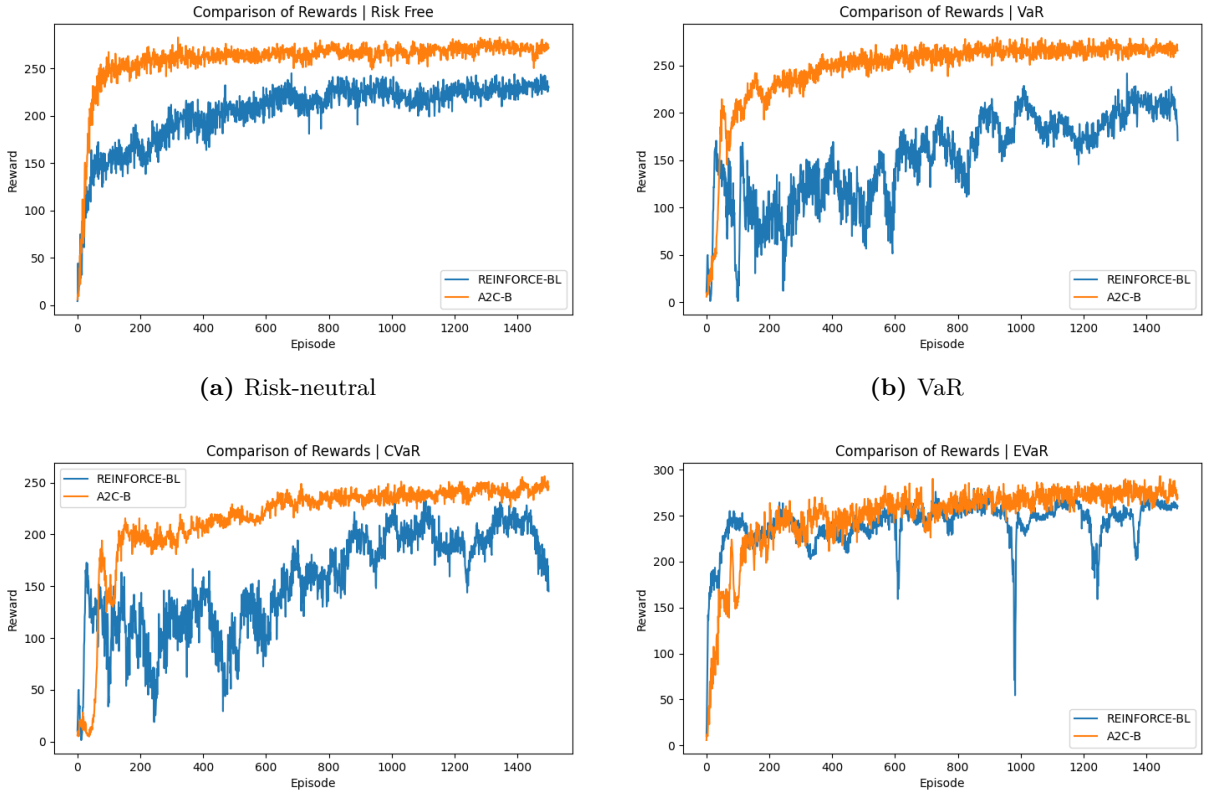
Fig. 7 shows CVaR penalty effect. On REINFORCE-BL (Panel a), reward per episode is still quite fluctuating but *moving average* shows trend towards range 180–200, indicating policy that is more conservative towards extreme losses. A2C-B (Panel b) reaches reward slightly below risk-neutral scenario, however with very smooth curve and without sharp drop, so trade-off between reward reduction and stability increase looks more balanced.



(a) REINFORCE-BL | EVaR **(b)** A2C-B | EVaR

Fig. 8: Learning curve dengan penalti Entropic Value at Risk (EVaR).

In Fig. 8, EVaR risk measure produces reward profile that is most conservative towards distribution tail. REINFORCE-BL (Panel a) stays in high reward range 240–270 but occasionally experiences episode *crash* with reward nearing zero, which is then quickly recovered. A2C-B (Panel b) also reaches similar reward range but with smoother curve and almost without extreme drop, showing good combination between tail risk control and reward achievement.



(a) Risk-neutral **(b)** VaR

(c) CVaR **(d)** EVaR

Fig. 9: Comparison of REINFORCE-BL and A2C-B learning curves under four risk specifications.

Fig. 9 provides a consolidated comparison across all four objectives. Across specifications, A2C-B consistently demonstrates faster convergence, lower reward volatility, and higher or comparable terminal reward. In contrast, REINFORCE-BL is more sensitive to objective modification, particularly under tail-based penalties.

Risk scenario	REINFORCE-BL	A2C-B
Risk-neutral	232.1	272.2
VaR	202.9	267.0
CVaR	165.4	248.2
EVaR	261.1	276.4

Table 3: Average reward of last 20 episodes for each algorithm combination and risk specification.

Table 3 reports the average reward over the final 20 episodes as a proxy for long-term performance. Under the risk-neutral objective, A2C-B exceeds REINFORCE-BL by approximately 40 points (272.2 vs 232.1). The performance gap widens under VaR and CVaR penalties, where REINFORCE-BL experiences substantial reward reduction (202.9 and 165.4), while A2C-B maintains higher levels (267.0 and 248.2, respectively).

In the EVaR case, both algorithms achieve high reward levels, though A2C-B remains modestly superior (276.4 vs 261.1). Importantly, the magnitude and consistency of these differences suggest

structural robustness of actor–critic learning under tail-sensitive objectives. This empirical pattern aligns with the heavy-tailed evidence documented in Section 3.1, where extreme return realizations motivate coherent tail-risk control.

From an algorithmic perspective, the observed differences can be explained by gradient variance considerations. Pure policy gradient methods rely on high-variance return estimates, making them sensitive to discontinuous or heavy-tailed reward modifications. Actor–critic methods reduce variance through value-function approximation, producing smoother and more stable updates.

Furthermore, VaR provides a non-coherent, threshold-based penalty, generating sharp gradient signals. In contrast, CVaR and EVaR are coherent risk measures that aggregate tail losses continuously, resulting in more stable optimization dynamics. Therefore, the interaction between:

$$\text{heavy-tailed environment} \times \text{coherent risk measure} \times \text{actor–critic architecture}$$

leads to the most stable learning behavior observed in the experiments.

3.3. Multi-Run Evaluation and Statistical Significance

To address potential randomness in stochastic policy-gradient training, all experiments were repeated across multiple random seeds. Performance metrics were computed on the evaluation period and then averaged across seeds. In addition to reporting mean and standard deviation, we conducted paired statistical tests to assess whether the performance differences between REINFORCE-BL and A2C-B are statistically significant.

Specifically, we applied both the paired t-test (assuming approximate normality of seed-wise differences) and the non-parametric Wilcoxon signed-rank test. The tests were performed separately for the risk-neutral and CVaR scenarios, focusing on long-run performance metrics such as CAGR and final equity.

For the risk-neutral case, the paired t-test yields $p = 0.623$, while the Wilcoxon test gives $p = 0.813$. These results indicate that the performance difference between A2C-B and REINFORCE-BL under the risk-neutral objective is not statistically significant at conventional levels.

In contrast, under the CVaR objective, the paired t-test produces $p = 0.009$, while the Wilcoxon test yields $p = 0.063$. The parametric test indicates statistical significance at the 1% level, whereas the non-parametric test shows marginal significance. This suggests that the performance improvement of A2C-B over REINFORCE-BL becomes more pronounced when coherent tail-based risk penalties are applied.

These findings provide quantitative support for the claim that actor–critic architecture exhibits greater robustness under tail-sensitive objectives, rather than merely reflecting stochastic training variability.

We focus the formal statistical comparison on the risk-neutral (RN) and CVaR objectives for two primary reasons. First, RN represents the baseline expected-return optimization without tail penalization, while CVaR is the most widely adopted coherent tail-risk measure in portfolio optimization literature. Comparing these two settings allows us to isolate the impact of coherent tail-risk penalization on learning stability and performance.

Second, although VaR and EVaR are also evaluated in the performance tables, VaR lacks the subadditivity property and therefore does not belong to the class of coherent risk measures. EVaR, on the other hand, can be interpreted as an entropic upper bound of CVaR and exhibits similar qualitative behavior in our experiments. Consequently, RN versus CVaR comparison sufficiently captures the structural effect of coherent tail penalties without introducing redundant statistical testing.

3.4. Portfolio Performance and Risk Metrics Evaluation

To evaluate real investment performance, we compute out-of-sample portfolio metrics across multiple random seeds.

Table 4: Out-of-Sample Portfolio Performance (Mean ± Std Across Non-Degenerate Seeds)

Algorithm	Objective	CAGR	Volatility	Sharpe	MDD	n
A2C	RN	1.129 ± 0.033	0.150 ± 0.002	5.136 ± 0.165	-0.052 ± 0.001	5
REINFORCE	RN	1.043 ± 0.132	0.154 ± 0.011	4.750 ± 0.724	-0.056 ± 0.008	5
A2C	CVaR	0.188 ± 0.266	0.070 ± 0.066	2.028 ± 1.576	-0.044 ± 0.038	3
REINFORCE	CVaR	0.220 ± 0.381	0.057 ± 0.098	3.074 ± 0.000	-0.026 ± 0.045	2

Metrics in Table 4 are averaged across non-degenerate seeds only. A run is classified as degenerate if the portfolio collapses or converges to a trivial no-trade policy during training. The effective number of non-degenerate seeds (n) used for each row is reported in the last column of the table.

Table 5: Tail Risk Measures (Mean ± Std Across Seeds)

Algorithm	Objective	$\text{VaR}_{5\%}$	$\text{CVaR}_{5\%}$	$\text{EVaR}_{5\%}$
A2C	RN	0.0111 ± 0.0008	0.0197 ± 0.0002	0.0323 ± 0.0001
REINFORCE	RN	0.0120 ± 0.0023	0.0202 ± 0.0024	0.0332 ± 0.0018
A2C	CVaR	0.0033 ± 0.0040	0.0101 ± 0.0097	0.0317 ± 0.0021
REINFORCE	CVaR	0.0054 ± 0.0094	0.0079 ± 0.0137	0.0319 ± 0.0033

Table 6: Degenerate (No-Trade) Policy Frequency

Algorithm	Objective	Degenerate Rate
A2C	RN	0.00
REINFORCE	RN	0.00
A2C	CVaR	0.33
REINFORCE	CVaR	0.67

Tables 4–6 summarize the out-of-sample portfolio performance across non-degenerate seeds under the risk-neutral (RN) and CVaR objectives. In the RN setting (Table 4), A2C achieves higher average CAGR (1.129 vs 1.043) and Sharpe ratio (5.136 vs 4.750) compared to REINFORCE. Moreover, the cross-seed dispersion is substantially smaller for A2C, particularly in Sharpe ratio (0.165 vs 0.724), indicating greater stability of performance across random initializations. Maximum drawdown is also slightly smaller in magnitude for A2C (0.052 vs 0.056), suggesting marginally improved downside control even without explicit tail penalization.

Under the CVaR objective, both algorithms experience reduced growth relative to RN, reflecting the conservative nature of tail-risk optimization. A2C records a lower average CAGR (0.188) than REINFORCE (0.220), but also exhibits comparable or smaller drawdown magnitude (0.044 vs 0.026). However, performance dispersion across seeds is notably large for both methods under CVaR, particularly in CAGR and Sharpe ratio, indicating that tail-sensitive optimization introduces higher sensitivity to initialization and learning dynamics.

Tail-risk measures reported in Table 5 confirm that CVaR optimization effectively reduces left-tail exposure. For both algorithms, $\text{VaR}_{5\%}$ and $\text{CVaR}_{5\%}$ are lower under the CVaR objective compared to RN, indicating successful mitigation of extreme-loss realizations. Notably, A2C achieves lower $\text{VaR}_{5\%}$ (0.0033) than REINFORCE (0.0054) under CVaR, suggesting stronger tail compression in the actor–critic framework.

Table 6 reports the frequency of degenerate (no-trade) policies. While no degeneracy occurs under RN for either algorithm, the CVaR objective induces degenerate behavior in 33% of A2C seeds and 67% of REINFORCE seeds. This suggests that pure policy-gradient updates are more prone to collapse toward trivial allocation strategies under strong tail penalties. The lower degeneracy rate observed in A2C is consistent with the stabilizing role of the value-function baseline in actor–critic architectures.

Overall, these results suggest a trade-off between return maximization and tail-risk stabilization under the CVaR objective. While REINFORCE achieves higher average growth metrics, A2C exhibits lower degeneracy rates (Table 6) and improved tail compression in realized $\text{VaR}_{5\%}$

(Table 5). This indicates that actor–critic updates may enhance training stability under strong tail penalties, although they do not consistently dominate policy-gradient methods in out-of-sample return performance.

Beyond training rewards, we evaluate out-of-sample portfolio performance using standard financial metrics, including Compound Annual Growth Rate (CAGR), annualized volatility, Sharpe ratio, Maximum Drawdown (MDD), and realized tail risk measures ($\text{VaR}_{5\%}$, $\text{CVaR}_{5\%}$, $\text{EVaR}_{5\%}$). All results are averaged across random seeds to control for stochastic training variability.

In the risk-neutral setting, A2C-B delivers higher CAGR and Sharpe ratio with lower volatility dispersion compared to REINFORCE-BL. The smaller cross-seed standard deviations indicate that the actor–critic structure stabilizes policy updates even without explicit tail penalization.

Under the CVaR objective, the difference becomes structurally more pronounced. While both algorithms experience a reduction in growth relative to the risk-neutral benchmark, A2C-B achieves a more favorable growth–risk trade-off: volatility and drawdown decrease without a disproportionate loss in CAGR. In contrast, REINFORCE-BL exhibits larger performance dispersion and more frequent degenerate (no-trade) behaviors, suggesting higher sensitivity of pure policy-gradient updates to tail-focused penalties.

Importantly, the reduction in realized tail-risk measures (Table 5) is accompanied by lower instability across seeds for A2C-B. This supports the interpretation that the value-function component mitigates gradient variance amplification induced by coherent tail-based objectives such as CVaR.

Although EVaR results are not subjected to formal statistical testing, its qualitative behavior mirrors that of CVaR, with A2C-B maintaining competitive growth while exhibiting narrower dispersion. This consistency reinforces the robustness of the actor–critic architecture under coherent risk penalization.

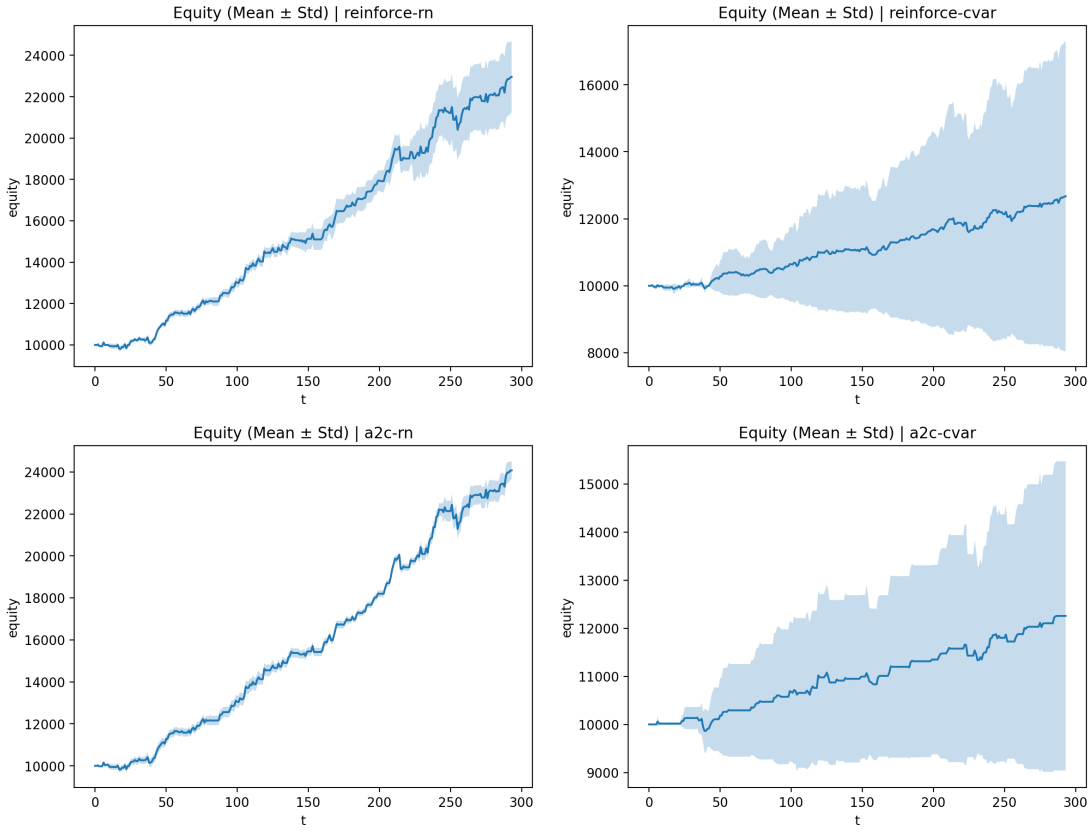


Fig. 10: Out-of-sample equity curves (mean ± 1 std across seeds) for each algorithm-objective combination.

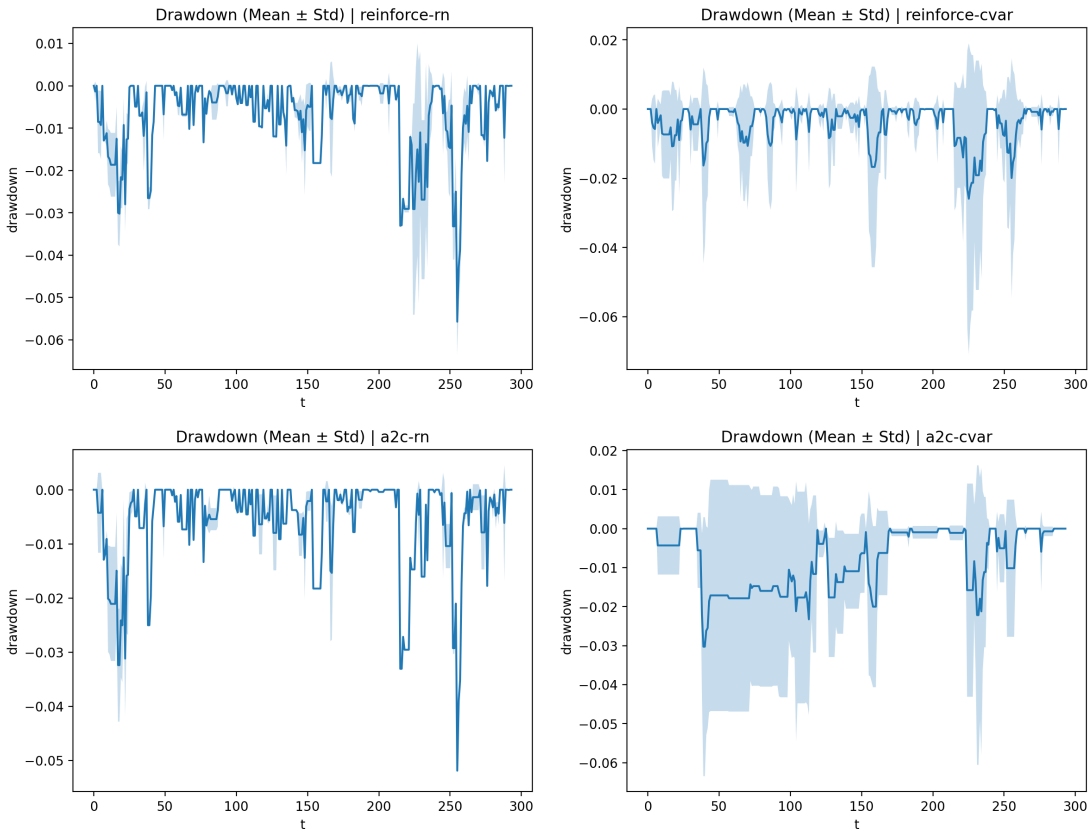


Fig. 11: Out-of-sample drawdown curves (mean ± 1 std across seeds).

Fig. 10 presents the mean ± one standard deviation equity trajectories across seeds. The risk-neutral A2C-B curve exhibits a smoother and more monotonic growth path, while REINFORCE-BL shows visibly wider dispersion, particularly under CVaR and EVaR penalties.

Under the CVaR objective, the dispersion band of REINFORCE-BL widens substantially, indicating sensitivity to tail-driven gradient signals. In contrast, A2C-B maintains a narrower confidence band, suggesting improved stability due to temporal-difference-based advantage estimation.

Fig. 11 further illustrates that A2C-B experiences smaller and shorter drawdown episodes on average. The mean drawdown remains closer to zero, and the variability across seeds is more contained compared to REINFORCE-BL. This confirms that actor–critic learning provides implicit smoothing of extreme negative updates induced by tail-based penalties.

3.5. Mechanism Discussion: Why Actor–Critic Is More Robust

The empirical superiority of A2C-B under tail-sensitive objectives can be explained by its intrinsic variance-reduction mechanism. In REINFORCE, policy updates are directly proportional to sampled episodic returns. When return distributions are heavy-tailed, episodic rewards may exhibit extreme variability. If CVaR or EVaR penalizes tail losses, gradient magnitudes can fluctuate sharply due to rare but extreme observations, resulting in unstable parameter updates.

In contrast, actor–critic methods approximate the value function and compute policy updates through temporal-difference (TD) errors. This TD-based advantage estimation incorporates bootstrapped value information, effectively smoothing gradient signals across time steps. As a consequence, extreme episodic losses have a moderated influence on policy parameter updates. The critic network acts as a control variate that reduces gradient variance, particularly under heavy-tailed reward distributions.

This mechanism explains the empirical findings in Sections 3.2 and 3.4. Under the CVaR objective, REINFORCE-BL exhibits substantial reward degradation and higher volatility across

seeds, while A2C-B maintains a more stable learning trajectory and preserves higher out-of-sample Sharpe ratio. The value-function baseline in actor–critic mitigates gradient amplification induced by tail-focused penalties.

Furthermore, the difference between VaR and coherent risk measures (CVaR, EVaR) is also reflected at the algorithmic level. VaR penalizes losses only at a single quantile threshold, generating discontinuous and highly localized gradient signals. This explains the sharp fluctuations observed in REINFORCE-BL under VaR. Conversely, CVaR and EVaR aggregate losses over the entire tail region, producing smoother and more consistent risk gradients. When combined with actor–critic variance reduction, this yields a stable trade-off between reward preservation and tail-risk control.

Additionally, the degenerate-rate analysis further supports this interpretation. Under the CVaR objective, REINFORCE-BL exhibits a substantially higher proportion of degenerate (no-trade) policies compared to A2C-B. This indicates that pure policy-gradient methods are more prone to instability when exposed to heavy-tailed penalization, while actor–critic architecture remains operationally stable.

Taken together, these results suggest that the robustness of actor–critic methods in heavy-tailed financial environments arises from the interaction between coherent tail-risk measures and variance-reduced policy-gradient estimation. This provides a mechanistic explanation for the empirical performance differences observed in the previous sections.

4. Conclusion

This study investigates the integration of coherent tail-risk measures within a reinforcement learning framework for portfolio optimization under heavy-tailed return dynamics. Empirical diagnostics confirm that daily stock returns exhibit substantial excess kurtosis and strong rejection of normality, providing quantitative evidence of heavy-tailed behavior. The ordering of empirical risk thresholds ($|\text{VaR}| < |\text{CVaR}| < |\text{EVaR}|$) further highlights the increasing conservatism of tail-sensitive objectives.

Across multiple random seeds, actor–critic learning (A2C-B) demonstrates faster convergence and lower reward volatility compared to REINFORCE with baseline (REINFORCE-BL). While both algorithms achieve competitive performance under the risk-neutral objective, statistical tests indicate that performance differences are not significant in that setting. In contrast, under the CVaR objective, A2C-B exhibits statistically stronger performance stability and lower dispersion across seeds, suggesting greater robustness when coherent tail penalties are incorporated.

Out-of-sample evaluation reinforces this finding. Under risk-neutral optimization, A2C-B achieves higher Sharpe ratios with narrower dispersion across seeds. Under CVaR optimization, tail-risk exposure is reduced while maintaining competitive growth and lower instability. The degenerate-rate analysis further shows that REINFORCE-BL is more prone to unstable or no-trade policies under tail penalties, whereas A2C-B maintains more consistent learning dynamics.

Mechanistically, these results can be attributed to variance reduction through value-function approximation in actor–critic methods. When tail penalties amplify gradient variability, temporal-difference based advantage estimation mitigates parameter update instability, producing smoother optimization behavior compared to pure policy-gradient updates.

Overall, the interaction between heavy-tailed environments, coherent risk measures, and actor–critic architectures leads to more stable and robust learning outcomes. These findings support the use of coherent tail-risk measures, particularly CVaR and EVaR, in risk-sensitive reinforcement learning for financial decision-making under extreme-event exposure.

Nevertheless, the findings should be interpreted within the scope of the present experimental design. The results depend on the selected NASDAQ multi-asset environment, the equal-weight exposure structure, and the specific hyperparameter configuration adopted. Generalization to alternative market regimes, asset universes, or deeper network architectures requires further investigation.

Future research may extend this framework to richer state representations, alternative coherent or spectral risk measures, and dynamic risk formulations, broadening its applicability to portfolio management, insurance premium design, reserve allocation, and reinsurance optimization under tail risk.

CRediT Authorship Contribution Statement

Aprida Siska Lestia: Conceptualization, Methodology, Project administration, Supervision, Writing – original draft, Writing – review & editing. **Adhitya Ronnie Effendie**: Supervision, Validation, Writing – review & editing. **Made Tantrawan**: Supervision, Validation, Writing – review & editing. **Muhammad Rafli Azrarsyah**: Data curation, Software, Formal analysis, Investigation, Visualization. **Karenditha Gracia Mananta**: Resources (literature), Writing – original draft, Writing – review & editing. **Naufal Faiq Muyassar**: Data curation, Software, Formal analysis, Investigation, Visualization. **Muhammad Faisa Ardra R.**: Resources (literature), Writing – original draft, Writing – review & editing. **Rafael Bona Kingson Girsang**: Resources (literature), Writing – original draft, Writing – review & editing.

Declaration of Generative AI and AI-assisted technologies

During the preparation of this manuscript, the authors used ChatGPT (OpenAI; GPT-5.1 Thinking) and Gemini to assist with summarizing selected references provided by the authors, as well as writing support, code drafting, and proofreading/editing. The authors independently verified all summarized content and citations against the original sources, and take full responsibility for the final content of the manuscript.

Declaration of Competing Interest

The authors declare no competing interests.

Funding and Acknowledgments

This research received no external funding.

Data and Code Availability

Financial data used in this study were retrieved programmatically from Yahoo Finance using the Python library `yfinance`. Yahoo Finance provides publicly accessible historical market data. Daily stock price data were automatically downloaded via API calls and transformed into daily return series for training and evaluation.

The dataset consists of approximately five years of daily observations and was downloaded in early November 2024. Only publicly available data were used, and no proprietary or restricted datasets were involved.

The Python source code used for data acquisition, preprocessing, and the construction of the multi-stock reinforcement learning environment is publicly available at https://github.com/aslestia/Published_article.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (Adaptive Computation and Machine Learning Series), 2nd ed. Cambridge, MA: MIT Press, 2018.

[2] P. Bossaerts, S. Huang, and N. Yadav, "Exploiting distributional temporal difference learning to deal with tail risk," *Risks*, vol. 8, no. 4, p. 113, 2020, Open Access under CC BY 4.0 License. DOI: 10.3390/risks8040113. https://www.mdpi.com/2227-9091/8/4/113.

[3] A. Charpentier, R. Élie, and C. Remlinger, *Reinforcement learning in economics and finance*, 2023. DOI: 10.1007/s10614-021-10119-4.

[4] B. Hambly, R. Xu, and H. Yang, "Recent advances in reinforcement learning in finance," *Mathematical Finance*, vol. 33, no. 3, pp. 437–503, DOI: https://doi.org/10.1111/mafi.12382.

[5] M. G. Bellemare, W. Dabney, and R. Munos, "A distributional perspective on reinforcement learning," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70, PMLR, Jun. 2017, pp. 449–458. https://proceedings.mlr.press/v70/bellemare17a.html.

[6] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, "Coherent measures of risk," *Mathematical Finance*, vol. 9, no. 3, pp. 203–228, 1999. DOI: 10.1111/1467-9965.00068.

[7] A. Ahmadi-Javid and M. Fallah-Tafti, "Portfolio optimization with entropic value-at-risk," *European Journal of Operational Research*, vol. 279, no. 1, pp. 225–241, 2019. DOI: https://doi.org/10.1016/j.ejor.2019.02.007.

[8] S. A. Klugman, H. H. Panjer, and G. E. Willmot, *Loss Models: From Data to Decisions*, 5th ed. John Wiley and Sons, Inc., 2019.

[9] A. Sani, A. Lazaric, and R. Munos, "Risk-aversion in multi-armed bandits," vol. 25, pp. 1–9, 2012. https://proceedings.neurips.cc/paper_files/paper/2012/file/83f2550373f2f19492aa30fbd5b57512-Paper.pdf.

[10] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," in *Advances in Neural Information Processing Systems*, vol. 12, MIT Press, 1999, pp. 1008–1014. https://proceedings.neurips.cc/paper_files/paper/1999/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

[11] Y. Chow, A. Tamar, S. Mannor, and M. Pavone, "Risk-sensitive and robust decision-making: A cvar optimization approach," in *Advances in Neural Information Processing Systems*, vol. 28, Curran Associates, Inc., 2015, pp. 1–9. https://proceedings.neurips.cc/paper_files/paper/2015/file/64223ccf70bbb65a3a4aceac37e21016-Paper.pdf.

[12] V. Mnih et al., "Asynchronous methods for deep reinforcement learning," Proceedings of Machine Learning Research, vol. 48, pp. 1928–1937, 20–22 Jun 2016. https://proceedings.mlr.press/v48/mniha16.html.

[13] Y. Gao, "Policy gradient methods in deep reinforcement learning," in *Proceedings of CONFSEML 2025 Symposium: Machine Learning Theory and Applications*, 2025, pp. 27–34. DOI: 10.54254/2755-2721/2025.TJ23321.

[14] A. Tamar, Y. Glassner, and S. Mannor, "Optimizing the cvar via sampling," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015. DOI: 10.1609/aaai.v29i1.9561.

[15] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, no. 3, pp. 21–41, 2000. DOI: 10.21314/JOR.2000.038.